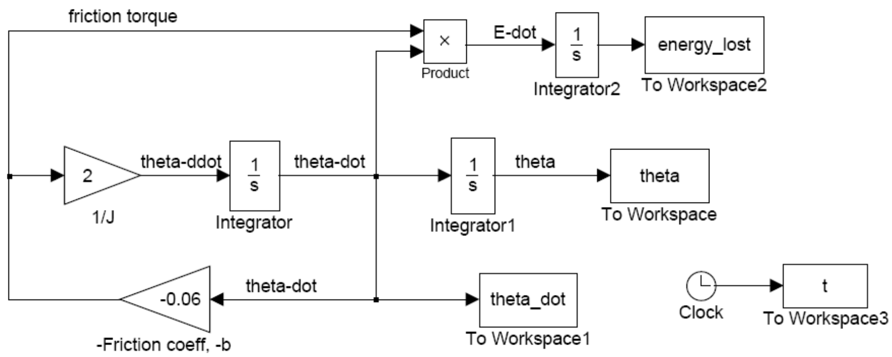
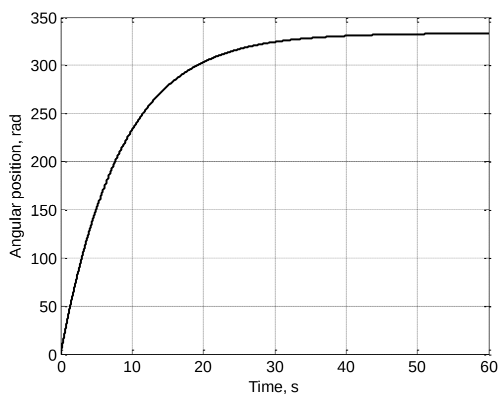
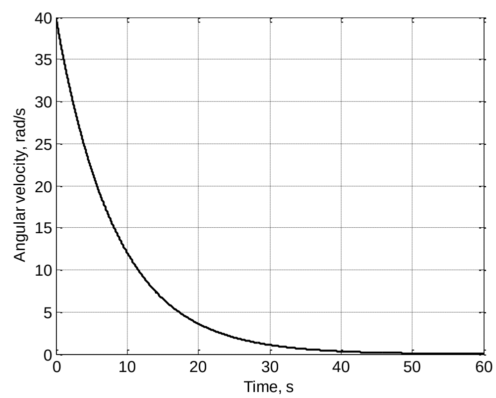
Chapter 6: Numerical Simulation of Dynamic Systems

6.1 The mathematical model of the rotational mechanical system is $J\ddot{\theta} + b\dot{\theta} = 0$. Because we have non-zero initial conditions we cannot use a transfer function. The Simulink model using the integrator-block method is shown below:

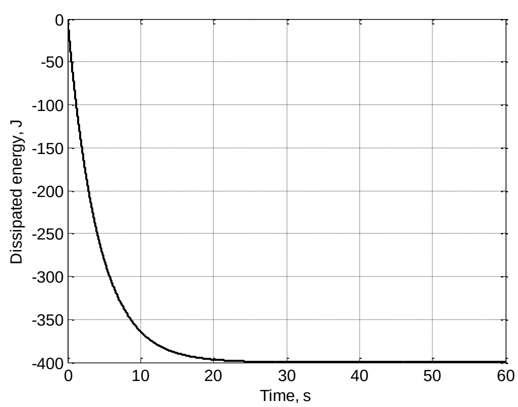


Note that $\ddot{\theta} = -b\dot{\theta}/J$ is integrated twice to yield $\dot{\theta}$ and θ. The initial angular velocity $\dot{\theta}_0$ is included in the first integrator. Energy-dissipation rate $\dot{\xi} = -b\dot{\theta}^2$ is integrated to obtain lost energy (see plots below).

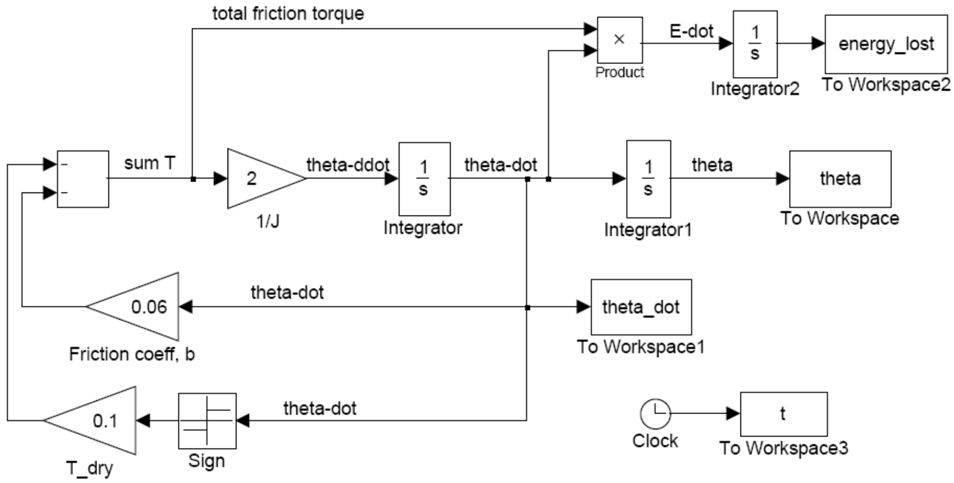




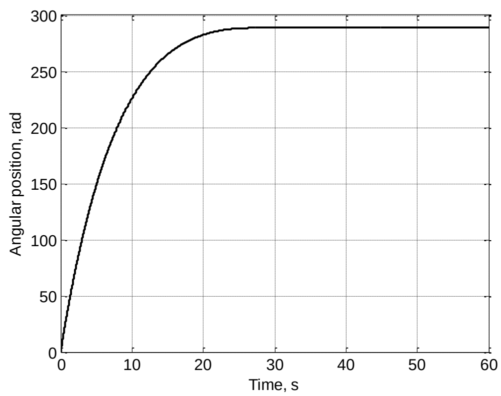
Note that the total energy lost (below) is equal to the total initial energy: $\frac{1}{2}J\dot{\theta}_0^2$ = 400 J.

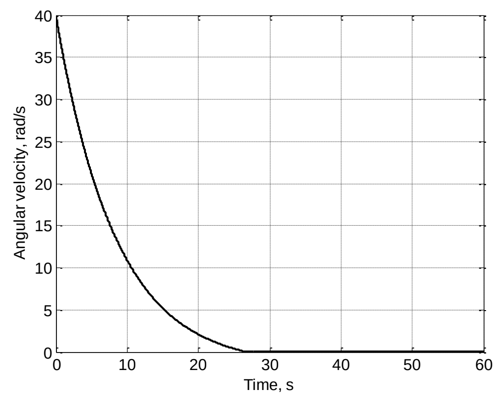


6.2 The mathematical model of the rotational mechanical system is $J\ddot{\theta}+b\dot{\theta}+T_{\text{dry}}\,\text{sgn}(\dot{\theta})=0$. The Simulink model from Problem 6.1 (with dry-friction torque added) is shown below:

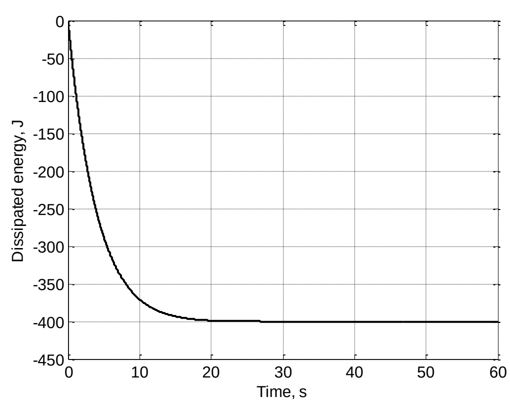


Angular position and angular velocity are below. Note that angular velocity decays “faster” than the result in Problem 6.1 due to the addition of the nonlinear dry-friction torque.





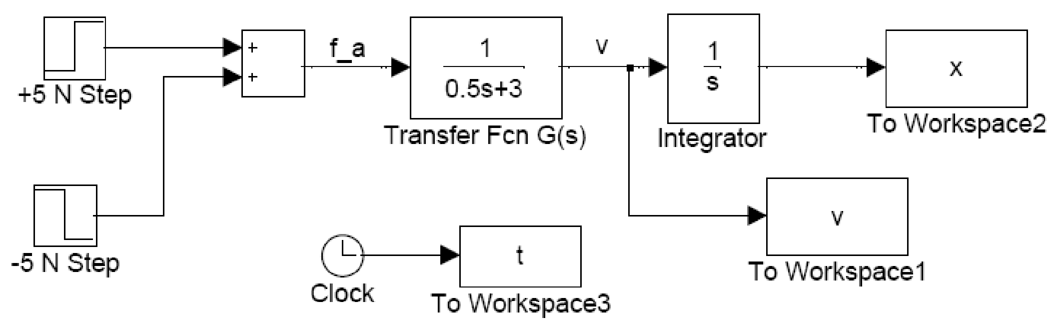
Note that the total energy lost (below) is equal to the total initial energy: $\dfrac{1}{2}J\dot{\theta}_0^2$ = 400 J.



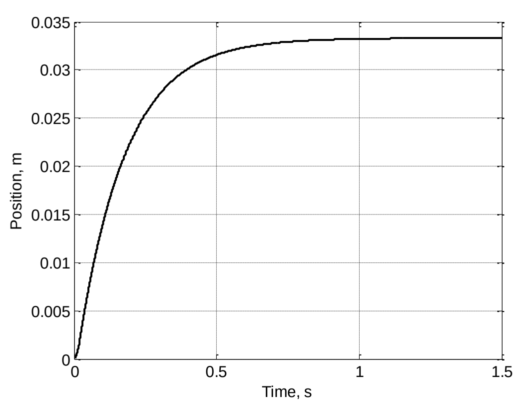
6.3 The mathematical model of the mechanical system is $m\dot{v} + bv = f_a(t)$. Because the system has zero initial conditions we may use a transfer function to obtain velocity $v(t) = \dot{x}(t)$.

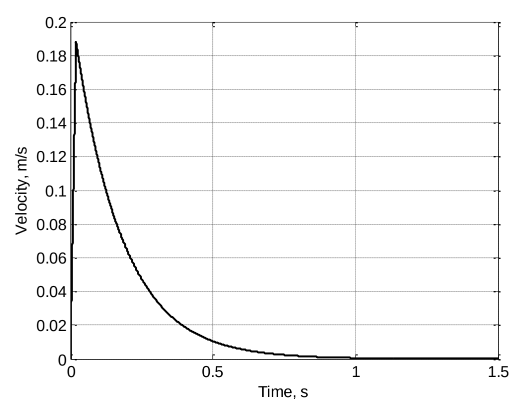
$$G(s) = \frac{V(s)}{F_a(s)} = \frac{1}{ms + b} = \frac{1}{0.5s + 3}$$

We can integrate velocity to obtain position *x*(*t*). The Simulink model is below:



The 5-N pulse force is constructed by adding a +5 N step and a –5 N step (delayed by 0.02 s). The Simulink model uses `ode4` and a fixed step size of 10^{-4} s. Position and velocity are below.





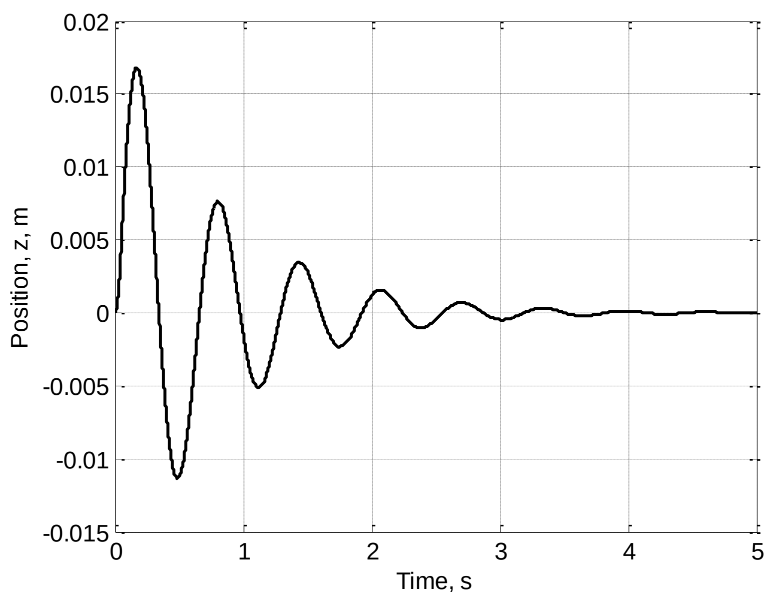
6.4 The mathematical model of the mechanical system is $m\ddot{z} + b\dot{z} + kz = F_a(t)$. Because the system has zero initial conditions we may use a transfer function for the system dynamics:

$$G(s) = \frac{Z(s)}{F_a(s)} = \frac{1}{ms^2 + bs + k} = \frac{1}{2s^2 + 5s + 200}$$

The following MATLAB commands will obtain the system's response to a pulse input:

```
>> sysG = tf(1,[ 2 5 200 ]);          % create sysG transfer function G(s)
>> t = 0:0.001:5;                     % time vector
>> Fa = zeros(size(t));               % zero vector for input force Fa(t)
>> Fa(1:51) = 8;                      % Input pulse Fa(t) = 8 N for 0 < t ≤ 0.05 s
>> [z,t] = lsim(sysG,Fa,t);           % Simulate system response using lsim.m
```

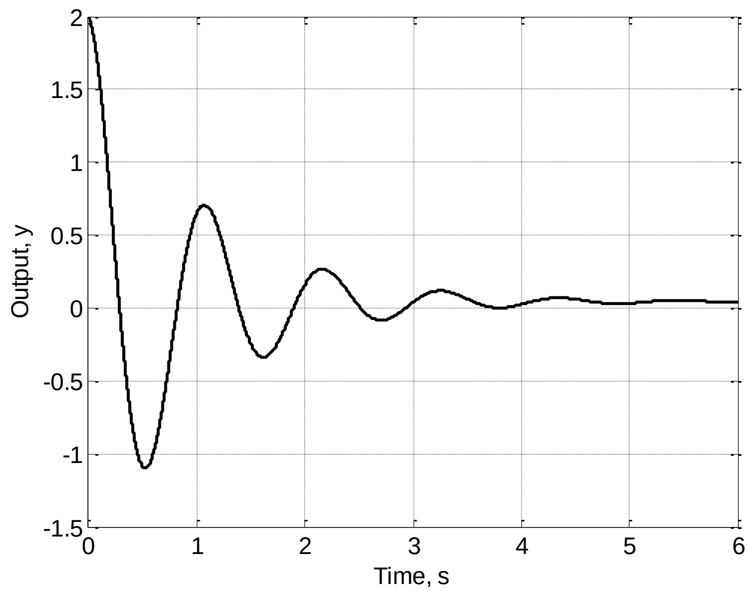
A plot of position $z(t)$ is created using the MATLAB command `plot(t,z)`.



6.5 The MATLAB command `lsim` allows us to represent the system as a SSR and include the non-zero initial conditions. The following commands compute the system's response.

```
>> A = [ 0 1 ; -34 -2 ];            % SSR A matrix
>> B = [ 0 ; 2.5 ];                 % SSR B matrix
>> C = [ 1 0 ];                     % SSR C matrix
>> D = 0;                           % SSR D matrix (null)
>> sys = ss(A,B,C,D);               % create sys as SSR
>> t = 0:0.001:6;                   % time vector
>> u = 0.6*ones(size(t));           % input vector, step u(t) = 0.6U(t)
>> x0 = [ 2 ; -1.5 ];               % initial state vector, x(0)
>> [y,t] = lsim(sys,u,t,x0);        % Simulate system response using lsim.m
```

A plot of position $y(t)$ is created using the MATLAB command `plot(t,y)`.

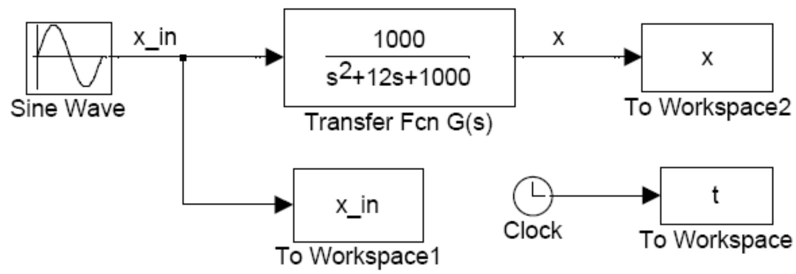


6.6 The mathematical model of the mechanical system is $m\ddot{x} + b\dot{x} + kx = kx_{\text{in}}(t)$

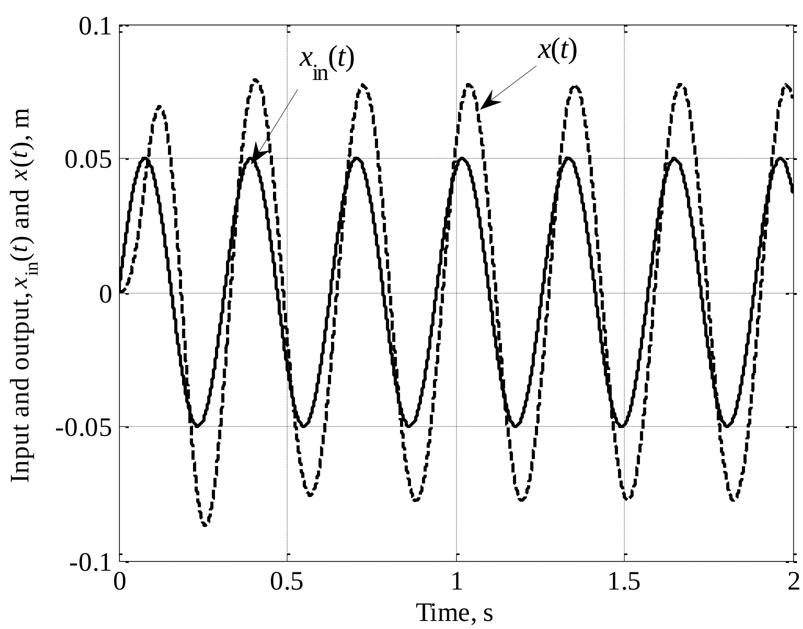
Therefore the transfer function is

$$G(s) = \frac{X(s)}{X_{\text{in}}(s)} = \frac{k}{ms^2 + bs + k} \qquad \text{or,} \qquad \frac{X(s)}{X_{\text{in}}(s)} = \frac{500}{0.5s^2 + 6s + 500} = \frac{1000}{s^2 + 12s + 1000}$$

The Simulink diagram is below



Setting the Sine Wave input to $x_{\text{in}}(t) = 0.05\sin 20t$ yields the simulation output:



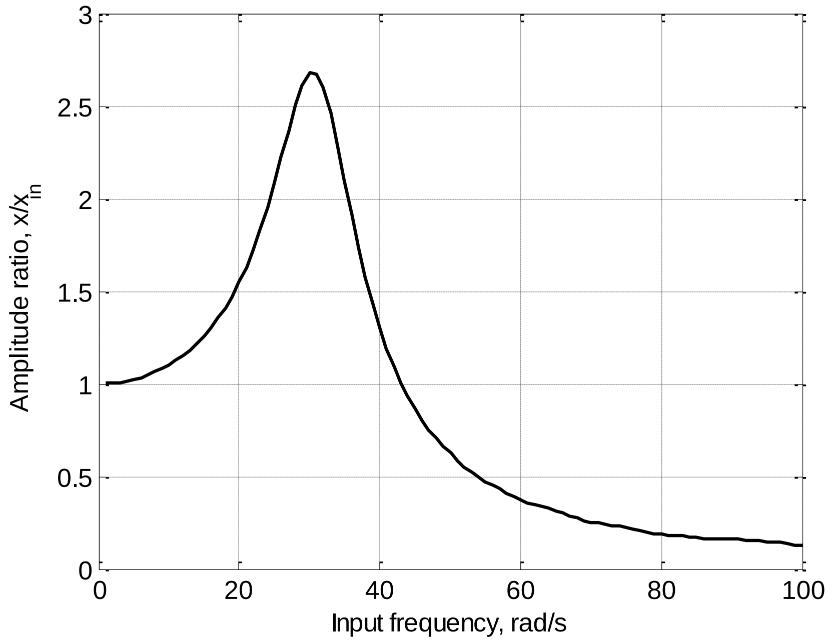
6.7 The Mfile for executing the Simulink model `Prob6_6.mdl` is below. A few key features are summarized:

1. Vector `w` is the input frequency, 1 to 100 rad/s in increments of 1 rad/s
2. The input frequency for a particular simulation run is `w_in`
3. The simulation stop time (`tstop`) is set to 10 cycles and the fixed step time (`dt`) is computed so that 1000 steps are required for each simulation.
4. `ode4` (RK-4 integration) is the numerical solver with fixed step time `dt`
5. The input frequency (`w_in`), simulation stop time (`tstop`), and step size (`dt`) are set as variables in the Simulink model Prob6_6.mdl (see `Configuration Parameters`)
6. The last 1/3 of the simulation data is `x_end = x(670:1001)`

```
%   run_Prob6.m
%   Mfile for Problem 6.7
%
w = 1:1:100;                        % input frequency vector, rad/s
for i=1:100
      clear t x x_in                  %  clear old values
      w_in = w(i);                    %  frequency, rad/s
      Tperiod = 2*pi/w_in;            %  period, s
      tstop = 10*Tperiod;             %  sim end time, s (10 cycles)
      dt = tstop/1000;                %  RK-4 integration step size, s
      sim Prob6_6                     %  execute Simulink model
      x_end = x(667:1001);            %  use last 1/3 of simulation data
      AR(i) = max(x_end)/max(x_in);   %  amplitude ratio
end

figure(1)
plot(w,AR)
grid
xlabel('Input frequency, rad/s')
ylabel('Amplitude ratio, x/x_i_n ')
```

Executing `run_Prob6` plots the amplitude ratio (next page):



This plot shows that the amplitude of $x(t)$ essentially matches the amplitude of $x_{in}(t)$ at low frequencies (ω ~ 1 rad/s) and again at ω ~ 43 rad/s since the amplitude ratio ~ 1. The output amplitude is nearly 2.7 times greater than the input amplitude at frequency ω ~ 30 rad/s. For input frequencies higher than 43 rad/s the output amplitude is *smaller* than the input amplitude. Finally, note that when ω = 20 rad/s the amplitude ratio = 1.5 which matches the simulation results plotted in Problem 6.5.

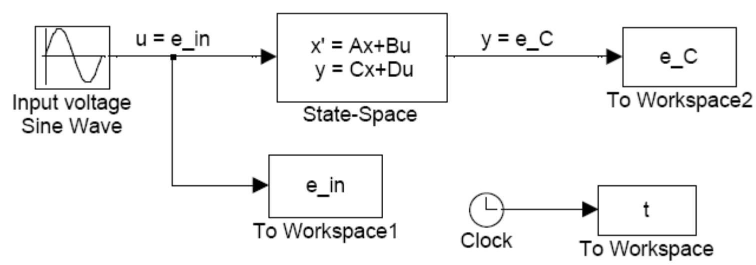
6.8 The state-space representation (SSR) for the electrical system was obtained in Problem 5.11

$$\dot{\mathbf{x}} = \begin{bmatrix} -R_2/L & -1/L \\ 1/C & -1/R_1C \end{bmatrix}\mathbf{x} + \begin{bmatrix} 1/L \\ 1/R_1C \end{bmatrix}u \qquad \text{and} \qquad y = \begin{bmatrix} 0 & 1 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \end{bmatrix}u$$

where $x_1 = I_L$, $x_2 = e_C$, and $u = e_{\text{in}}(t)$. Using the numerical values given in the problem the SSR is

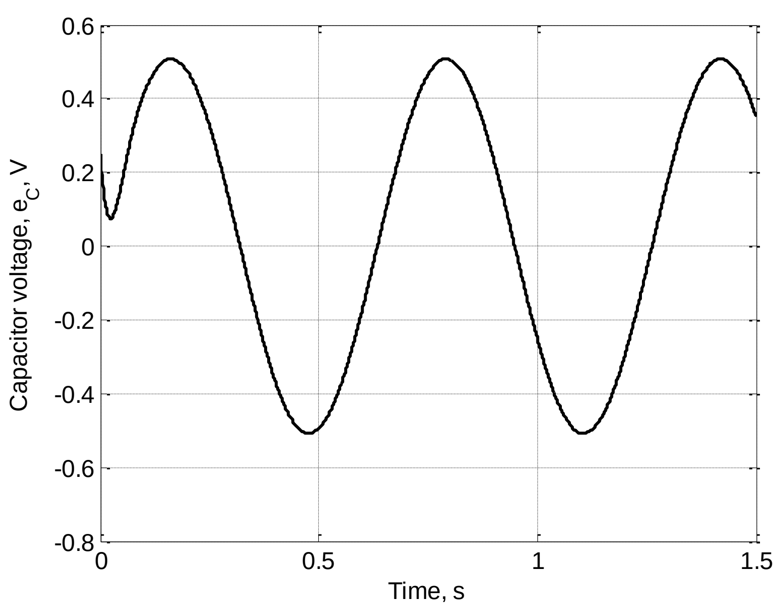
$$\dot{\mathbf{x}} = \begin{bmatrix} -20 & -100 \\ 25 & -62.5 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 100 \\ 62.5 \end{bmatrix}u$$

The following Simulink model determines the system response to a sinusoidal voltage input:



Note that the numerical values for the SSR matrices A, B, C, and D are entered in the State-Space block. Furthermore, the initial state vector is $x_1 = 0$ (no current) and $x_2 = 0.25$ V = 0.01 C/0.04 F and is entered in the State-Space block. The input Sine Wave is $e_{\text{in}}(t) = 0.5\sin 10t$ V.

The capacitor voltage response is shown below:



6.9 The SSR for the electrical system (see Problem 5.11 or Problem 6.8) is

$$\dot{\mathbf{x}} = \begin{bmatrix} -20 & -100 \\ 25 & -62.5 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 100 \\ 62.5 \end{bmatrix}u \quad \text{and} \quad y = \begin{bmatrix} 0 & 1 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \end{bmatrix}u$$

where $x_1 = I_L$, $x_2 = e_C$, and $u = e_{\mathrm{in}}(t)$.

The following MATLAB commands determine the system's response to a voltage input:

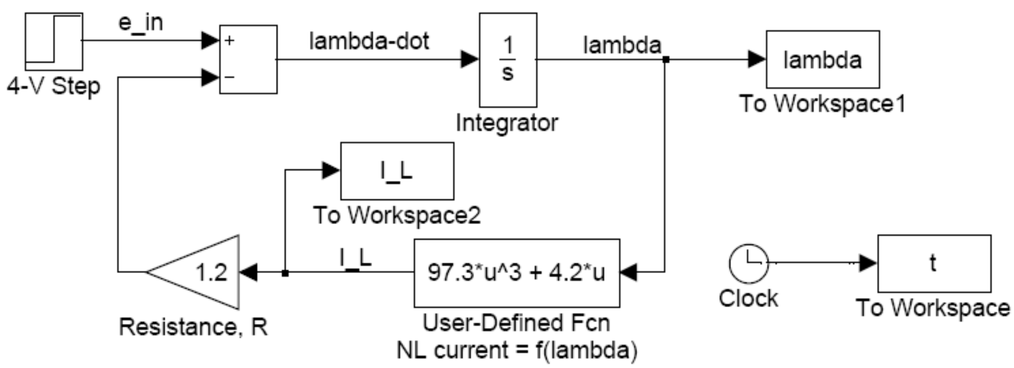
```
>> A = [ -20 -100 ; 25  -62.5 ];        % SSR A matrix
>> B = [ 100  ;  62.5 ];                % SSR B matrix
>> C = [ 0 1 ];                         % SSR C matrix
>> D = 0;                               % SSR D matrix (null)
>> sys = ss(A,B,C,D) ;                  % create system (SSR)
>> t = 0 : 0.001 : 1.5 ;                % time vector
>> u = 0.5*sin(10.*t);                  % input voltage vector
>> x0 = [ 0 ; 0.25 ] ;                  % initial states
>> [y,t] = lsim(sys,u,t,x0) ;           % simulate system using lsim.m
```

Using `plot(t,y)` yields a plot exactly like the plot shown in Problem 6.8.

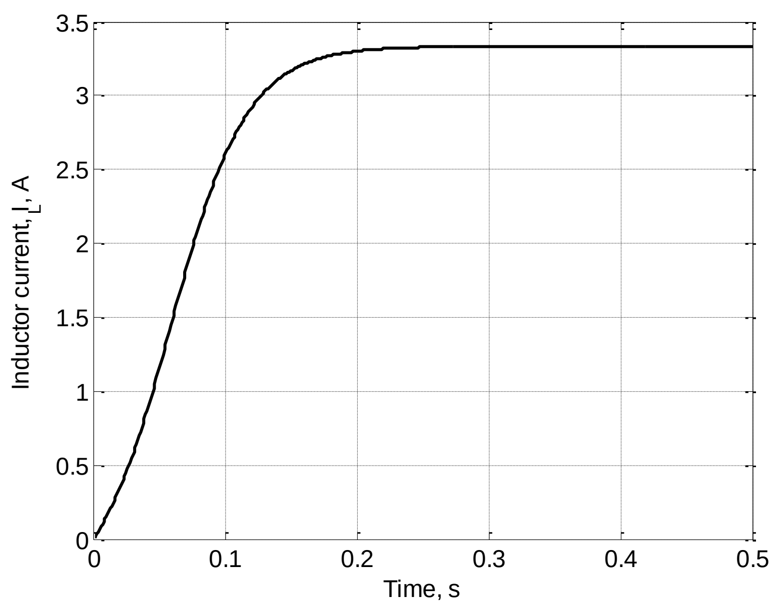
6.10 The mathematical model of the nonlinear RL electrical system (see Problem 3.11) is

$$\dot{\lambda} + R(97.3\lambda^3 + 4.2\lambda) = e_{\text{in}}(t) \quad \text{or} \quad \dot{\lambda} = -R(97.3\lambda^3 + 4.2\lambda) + e_{\text{in}}(t)$$

The Simulink model shown below determines the system's response. Note that the time-derivative of flux linkage must be integrated, and that the User-Defined function `Fcn` is used for the nonlinear current equation.



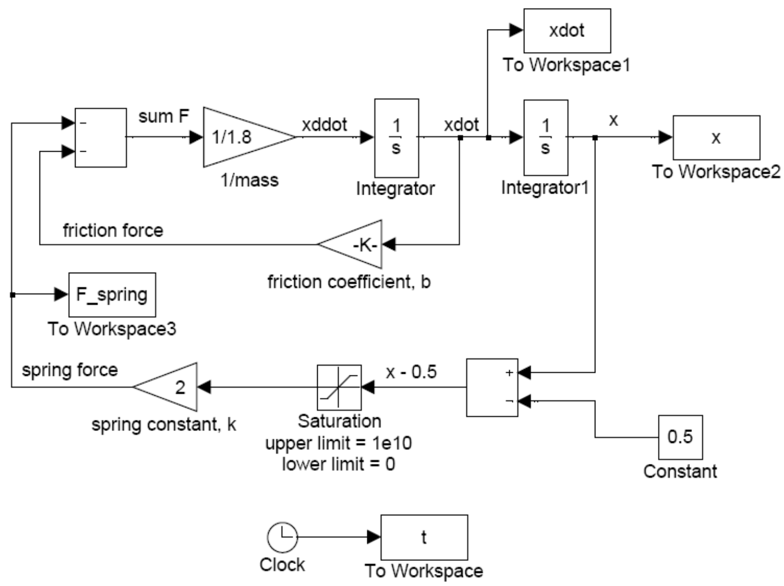
The inductor current I_L response is plotted below.



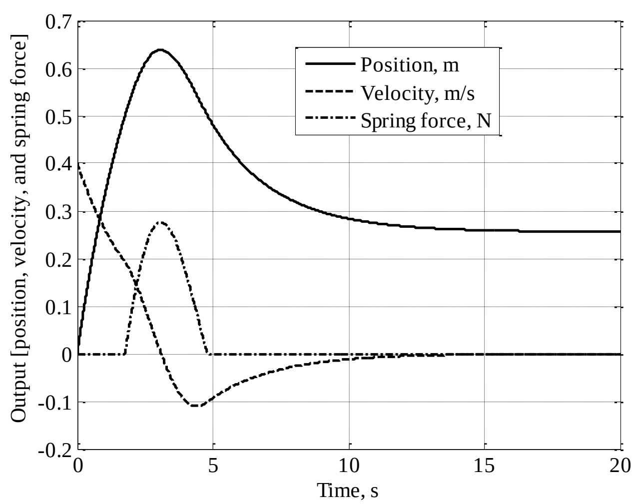
6.11 The mathematical model of the simple mechanical system is

$$m\ddot{x} + b\dot{x} = 0 \qquad \text{for } x < 0.5 \text{ m}$$
$$m\ddot{x} + b\dot{x} + k(x - 0.5) = 0 \qquad \text{for } x \geq 0.5 \text{ m}$$

Because the mass has initial conditions we must use the integrator-block method. The Simulink diagram is below:



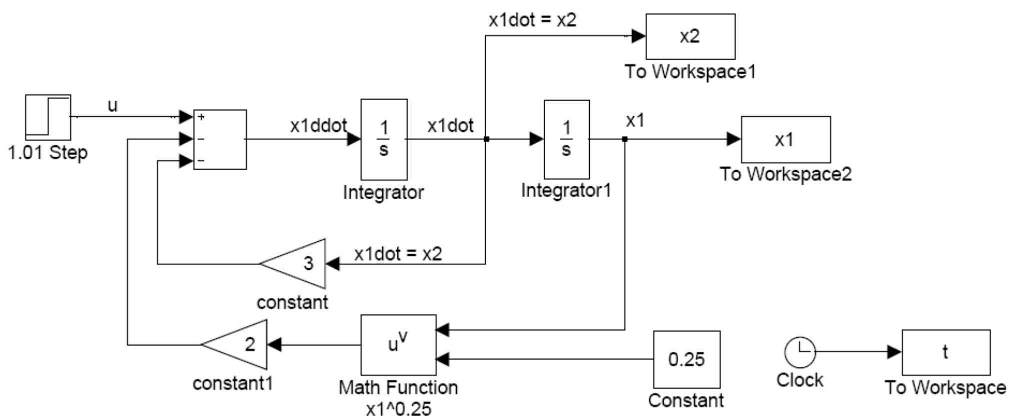
Note that the discontinuous spring force is computed by passing the difference $x - 0.5$ (m) through a saturation block with an upper limit of +Infinity and lower limit of zero. Therefore the difference $x - 0.5$ can never be negative and the spring cannot "pull" on the mass. The first integrator block has its initial condition set to $\dot{x}(0) = 0.4$ m/s (initial speed). Position, velocity, and spring force are combined on a single plot (below) in order to save space. Note that the mass does **NOT** return to $x = 0$.



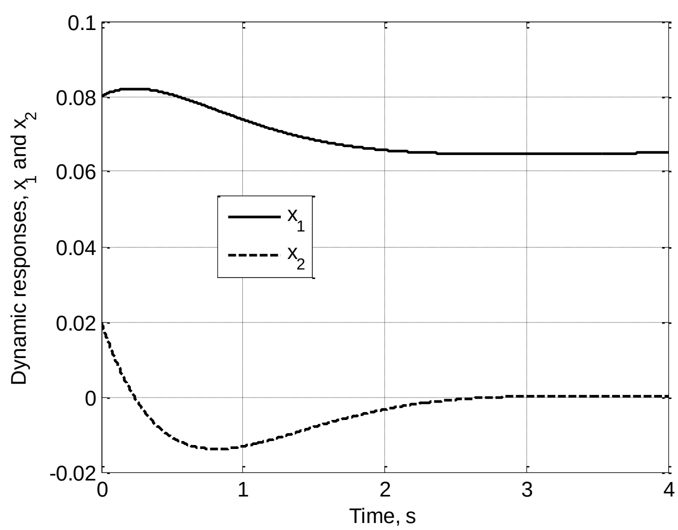
6.12 The mathematical model is

$$\dot{x}_1 - x_2 = 0 \qquad\qquad \dot{x}_1 = x_2$$
$$\text{or}$$
$$\dot{x}_2 + 2x_1^{1/4} + 3x_2 = u \qquad\qquad \ddot{x}_1 = -2x_1^{1/4} - 3\dot{x}_1 + u$$

a) Because the system is nonlinear and has non-zero initial conditions we must use the integrator-block method. The Simulink diagram is below. Note that the Math Function is used for $x_1^{1/4}$.



The responses for $x_1(t)$ and $x_2(t)$ are plotted below.



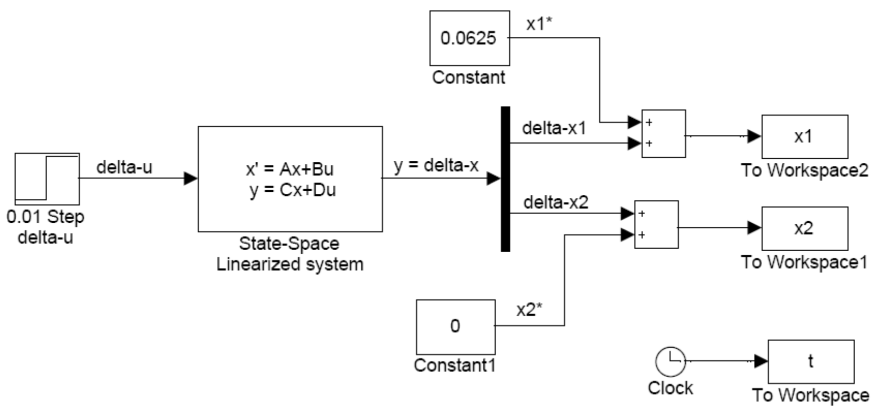
b) The linearized system (see Problem 5.5) is in state-space representation (SSR)

$$\delta\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ -4 & -3 \end{bmatrix}\delta\mathbf{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}\delta u \qquad \text{and} \qquad \mathbf{y} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\delta\mathbf{x} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}\delta u$$

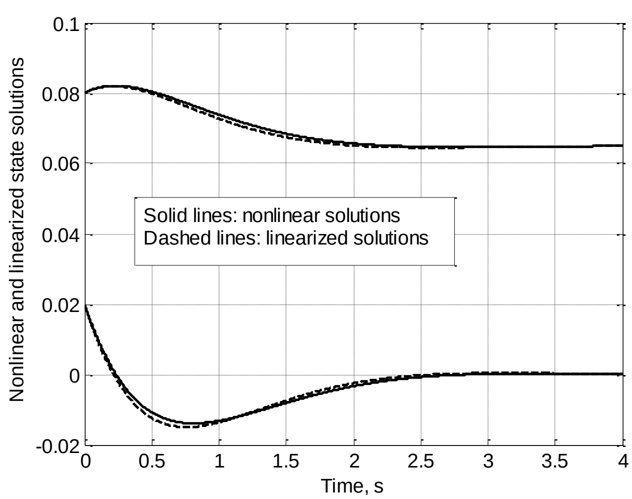
The linearized system is in terms of the perturbation states and perturbation input: $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}^*$ and $\delta u = u - u^*$. The output (or C) matrix is the identity matrix since we want both perturbation states $\delta x_1(t)$ and $\delta x_2(t)$ available for plotting. The nominal input is $u^* = 1$ and the nominal (reference) states are $x_1^* = 0.0625$ and $x_2^* = 0$. The SSR is obtained by linearization about $\mathbf{x}^*$ and u^*.

The following Simulink model uses the SSR for the linearized dynamics. Note that the reference state $\mathbf{x}^*$ must be added to the perturbation state vector $\delta\mathbf{x}(t)$ in order to estimate $\mathbf{x}(t)$. Note also that the perturbation input is $\delta u = u - u^* = 0.01$ since $u = 1.01$ in part (a) and $u^* = 1$.

Finally, note that the initial perturbation states are $\delta x_1(0) = x_1(0) - x_1^* = 0.08 - 0.0625 = 0.0175$, and $\delta x_2(0) = x_2(0) - x_2^* = 0.02 - 0 = 0.02$. The SSR block uses these initial perturbation states.

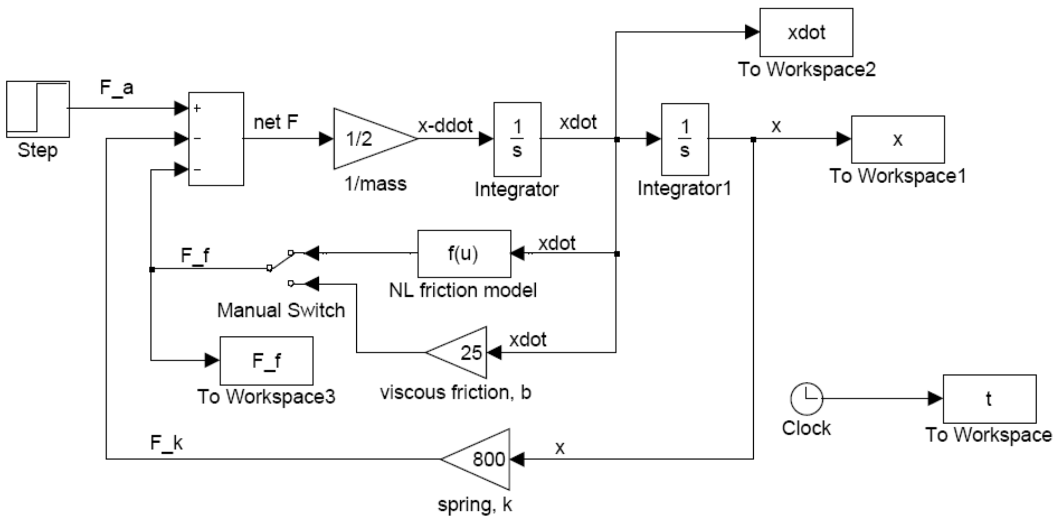


The linearized solutions $\mathbf{x}(t) = \delta\mathbf{x} + \mathbf{x}^*$ from the SSR simulation are plotted along with the nonlinear states obtained in part (a). The linearized solutions are shown as the dashed curves and the nonlinear solutions are the solid curves. Note that the linearized solutions are very close to the (true) nonlinear solutions.

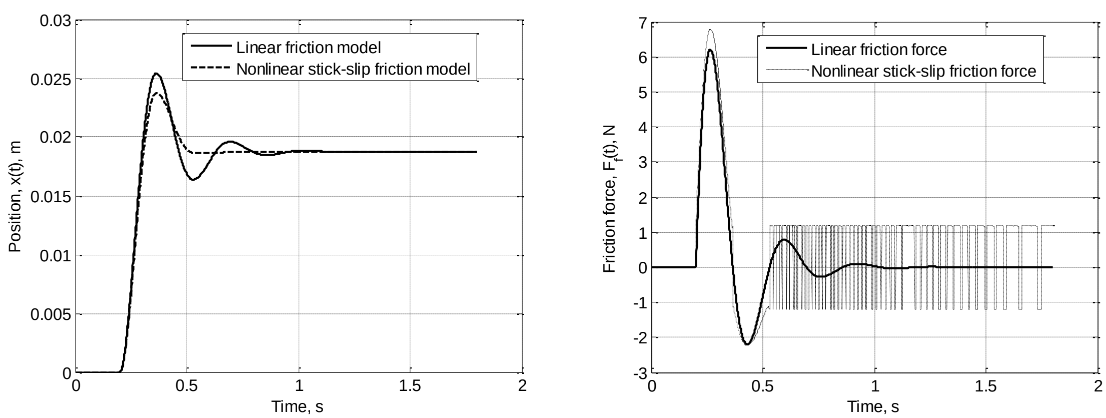


6.13 The mathematical model of the mechanical system is $m\ddot{x} + F_f + kx = F_a(t)$

The Simulink diagram (next page) uses a manual switch to select either the linear viscous friction or the nonlinear stick-slip friction model. The integrator-block method is required for nonlinear systems. The nonlinear friction force is computed using the User-Defined `Fcn` block.



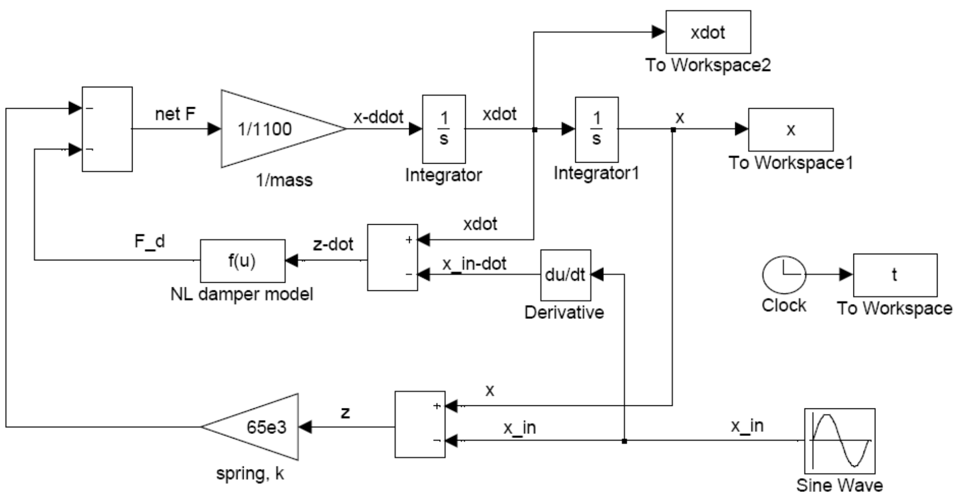
The responses for $x(t)$ and friction force $F_f(t)$ are plotted below. Note that when the stick-slip nonlinear friction model is used the response $x(t)$ exhibits a smaller peak value, fewer oscillations, and quicker response time. The nonlinear friction force is a bit larger (at the first peak) and exhibits "chatter" between +/- F_{st} (the "stiction" force) when velocity is approximately zero.



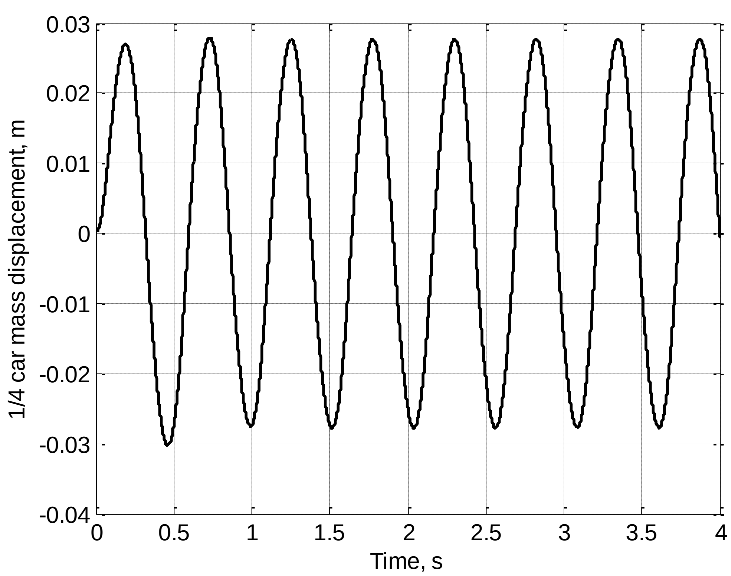
6.14 The mathematical model of the mechanical suspension system is $m\ddot{x} + F_d + kx = kx_{\text{in}}(t)$

where the nonlinear damper force is $F_d = \dfrac{4500\dot{z}}{\sqrt{\dot{z}^2 + v^2}}$ (N)

The Simulink diagram (next page) uses the integrator-block method since the dynamics are nonlinear. The nonlinear damper force is computed using the User-Defined `Fcn` block. The input sinusoidal function is created using the Sine Wave, and the Derivative block is used to compute $\dot{x}_{\text{in}}$. The relative position z and relative velocity $\dot{z} = \dot{x} - \dot{x}_{\text{in}}(t)$ are computed by differencing the mass position/velocity with input position/velocity.



The response for ¼-car mass position $x(t)$ is plotted below.

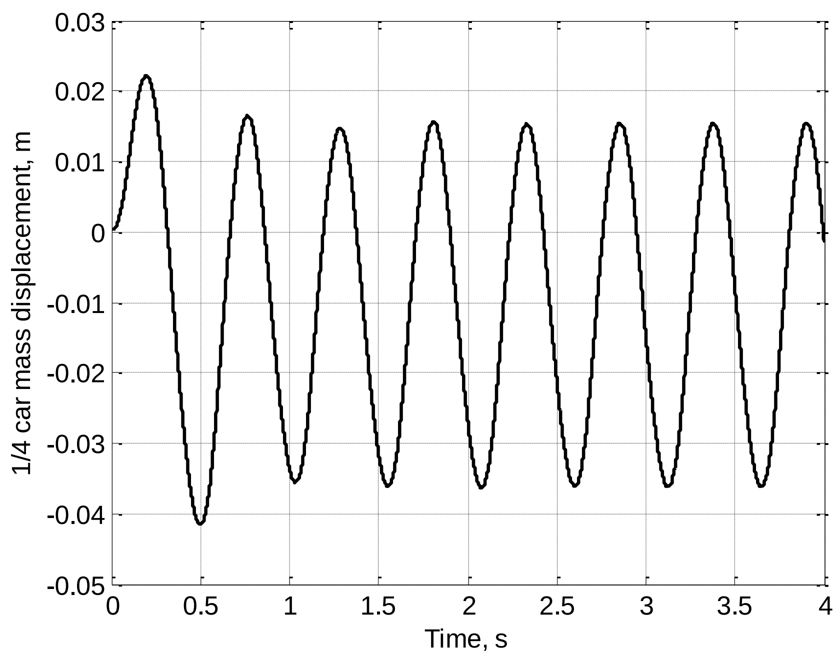


6.15 The mathematical model of the mechanical suspension system is $m\ddot{x} + F_d + kx = kx_{\text{in}}(t)$

where the nonlinear damper force is $F_d = \dfrac{3389(\dot{z} - v_1)}{\sqrt{(\dot{z} - v_1)^2 + v_2^2}} + 1020.84$ (N) (A)

The Simulink diagram is essentially the same Simulink diagram as used in Problem 6.14 except that the User-Defined `Fcn` block depicts the desired damper force F_d. Input to this User-Defined `Fcn` block is still relative velocity $\dot{z} = \dot{x} - \dot{x}_{\text{in}}(t)$.

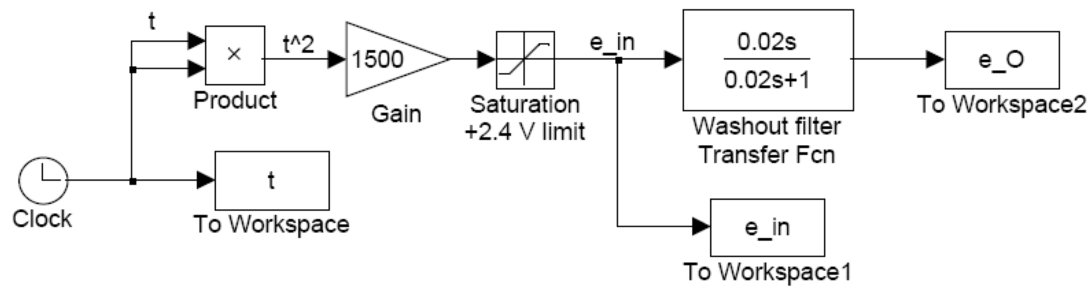
The response for ¼-car mass position $x(t)$ with the damper force model [Eq. (A)] is plotted on the next page. Note that the oscillatory response is not symmetric about zero; i.e., the amplitude of the "down" stroke (compression) is larger than the amplitude of the "up" stroke. The up/down amplitude of the position response in Problem 6.14 is symmetric about zero. The asymmetric behavior is because the damper force (A) is not symmetric – it is larger during extension (up).



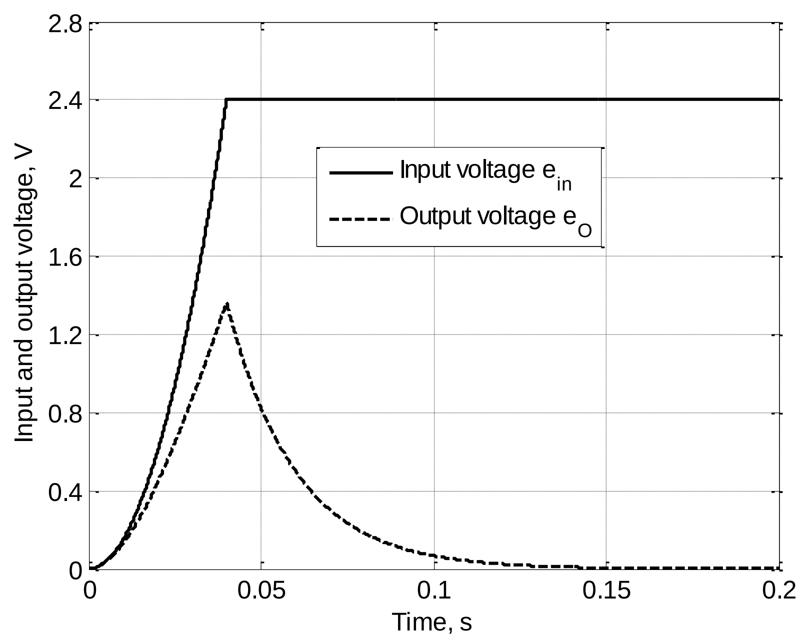
6.16 The mathematical model of the washout filter is $RC\dot{e}_O + e_O = RC\dot{e}_{\text{in}}(t)$

Hence, the transfer function of the filter is $G(s) = \dfrac{E_O(s)}{E_{\text{in}}(s)} = \dfrac{RCs}{RCs+1} = \dfrac{0.02s}{0.02s+1}$

Note that the quadratic input voltage $e_{\text{in}}(t)$ is created by squaring time t. The input voltage is limited to +2.4 V by passing the signal $1500t^2$ through a saturation block with limit +2.4.



Note that the output voltage (below) "washes out" to zero for a *constant* input voltage e_{in} = 2.4 V.



6.17 The transfer function of the filter is

$$G(s)=\frac{E_O(s)}{E_{\mathrm{in}}(s)}=\frac{RCs}{RCs+1}=\frac{0.02s}{0.02s+1}$$

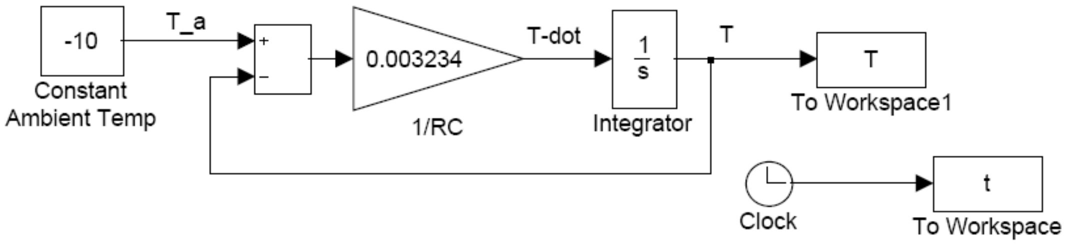
The following MATLAB commands determine the filter's response to a voltage input:

```
>> numG = [0.02 0 ];                  % numerator of transfer function G(s)
>> denG = [0.02 1 ];                  % denominator of transfer function G(s)
>> sysG = tf(numG,denG);              % create system (transfer function)
>> t = 0:1e-4:0.2;                    % time vector
>> u = 1500*t.^2;                     % input voltage vector = 1500t^2 V
>> u(401:2001) = 2.4;                 % input voltage vector = 2.4 V (for t ≥ 0.04 s)
>> [y,t] = lsim(sysG,u,t);            % simulate system using lsim.m
```

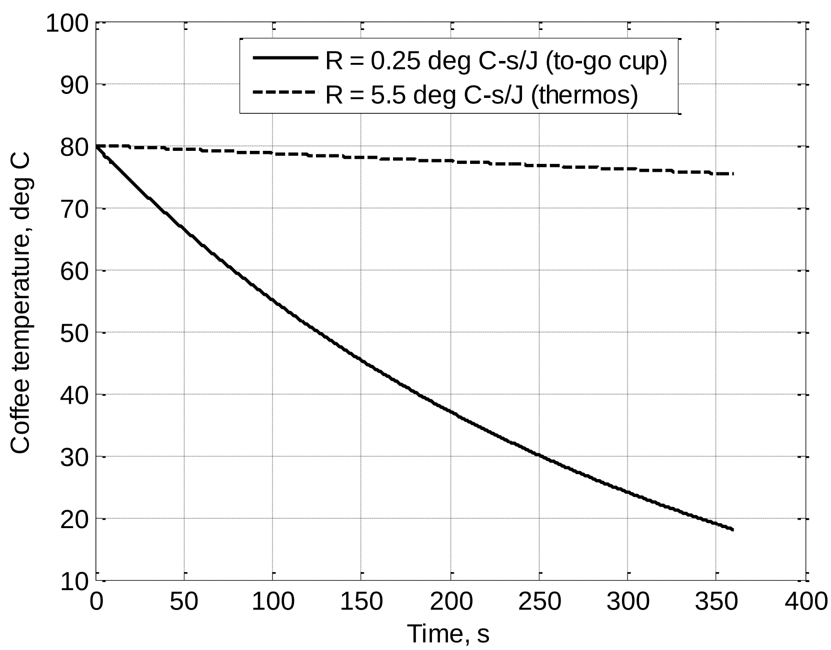
Using `plot(t,u,t,y)` yields a plot exactly like the plot shown in Problem 6.16.

6.18 The mathematical model of the thermal system is $RC\dot{T} + T = T_a$ (see Problem 4.23)

a) We must use the integrator-block method since the system has non-zero initial conditions. The Simulink diagram below uses the to-go cup insulation where $1/RC = 0.003234$ s^{-1}.



Plots of temperature $T(t)$ for a simulation time of 360 s (6 min) for both thermal resistance values are below. The final temperature with a to-go cup is **18.1 deg C** (part a) while the final temperature with a thermos ($1/RC = 0.000147$ s^{-1}) is **75.4 deg C** (part b).



6.19 The first-order model of the wind turbine generator system (see Example 2.8) is

$$J_{c1}\dot{\omega}_1 + b_{c1}\omega_1 = T_{\text{aero}} - \frac{1}{N}T_{\text{gen}}$$

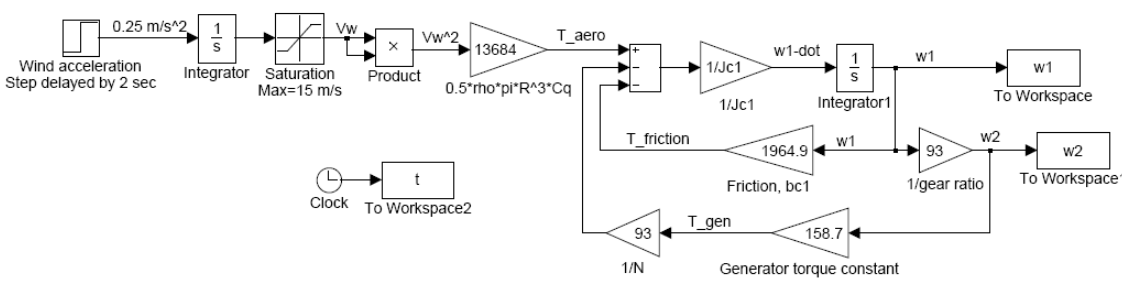
where the composite inertia is $J_{c1} = J_1 + \frac{1}{N^2}J_2$ and the composite friction is $b_{c1} = b_1 + \frac{1}{N^2}b_2$ Using the numerical values we have J_{c1} = 14,675,760 kg-m^2 and b_{c1} = 1964.9 N-m-s/rad.

a) At time t = 0 the turbine has constant angular velocity so $\dot{\omega}_1 = 0$; hence we can solve for angular velocity from the modeling equation (substitute $T_{\text{gen}} = 158.7\omega_2 = 158.7\omega_1/N$)

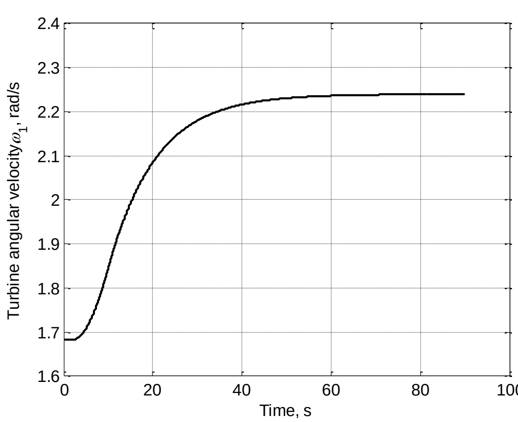
$$\omega_1(0) = \frac{T_{\text{aero}}}{b_{c1}} - \frac{158.7\omega_1(0)}{N^2 b_{c1}} \qquad \text{(A)}$$

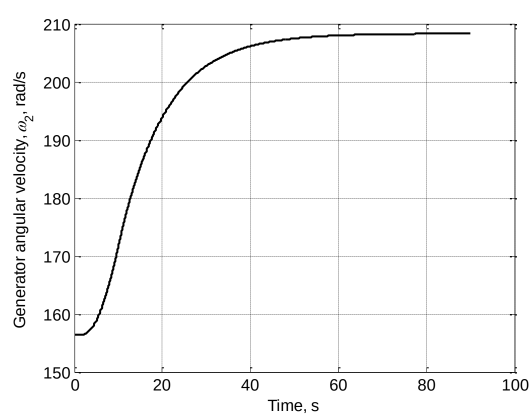
where $T_{\text{aero}} = \frac{1}{2}\rho\pi R^3 V_w^2 C_q$ = 2.3126(10^6) N-m. Solve Eq. (A) for the initial turbine shaft speed: **ω_1(0) = 1.6824 rad/s (16.1 rpm).** The initial generator-shaft speed is $\omega_2(0) = \omega_1(0)/N$ = **156.47 rad/s (1494.1 rpm).**

b) The Simulink diagram is below. The wind speed is computed by integrating the wind acceleration 0.25 m/s^2 (the integrator has initial condition $V_w(0)$ = 13 m/s) and passing the integrated wind acceleration through a Saturation block with an upper limit of 15 m/s. Note that the generator shaft speed ω_2 is computed by dividing the turbine speed ω_1 by the gear ratio N.



Plots of angular velocities $\omega_1(t)$ and $\omega_1(t)$ are below.





6.20 The state equation (SSR) of the pantograph system (see Problem 5.33) is

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -(k_1+k_2)/m_1 & -b_1/m_1 & k_2/m_1 & b_1/m_1 \\ 0 & 0 & 0 & 1 \\ k_2/m_2 & b_1/m_2 & -k_2/m_2 & -(b_1+b_2)/m_2 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 & 0 \\ k_1/m_1 & 0 \\ 0 & 0 \\ 0 & 1/m_2 \end{bmatrix}\mathbf{u}$$

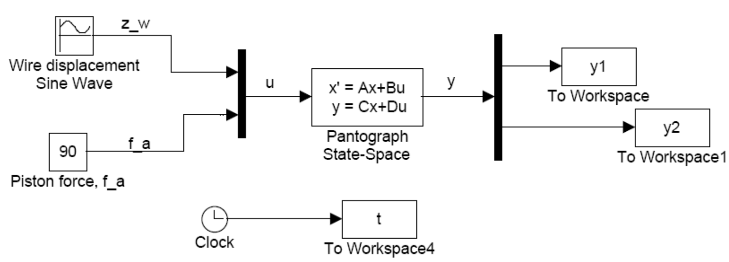
The state variables are $x_1 = z_1$, $x_2 = \dot{z}_1$, $x_3 = z_2$, and $x_4 = \dot{z}_2$; inputs are $u_1 = z_w(t)$ and $u_2 = f_a(t)$.

a) Using the numerical parameters the state equation becomes

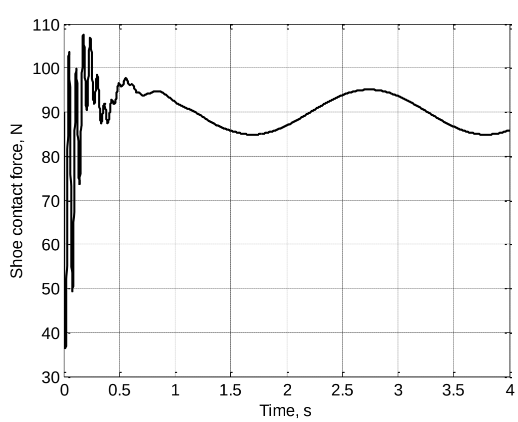
$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -9888.9 & -14.444 & 777.778 & 14.444 \\ 0 & 0 & 0 & 1 \\ 411.76 & 7.6471 & -411.76 & -9.4118 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 & 0 \\ 9111.1 & 0 \\ 0 & 0 \\ 0 & 0.058824 \end{bmatrix}\mathbf{u}$$

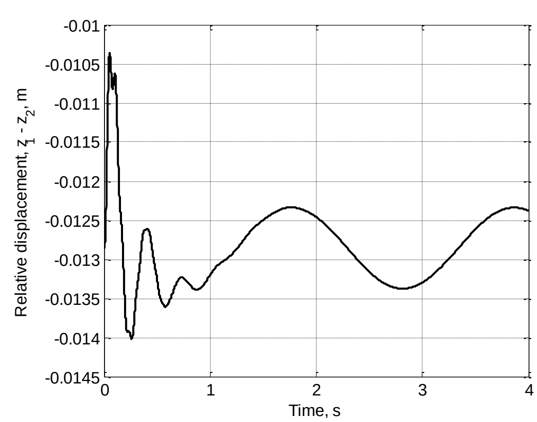
Multiplying the above A matrix by the initial state vector $\mathbf{x}(0) = [\ 0\ \ 0\ \ 0.0128571\ 0\ ]^T$ and adding the result to the B matrix times $\mathbf{u}(0) = [\ -0.0010976\ \ \ 90\ ]^T$ yields $\dot{\mathbf{x}}(0) \approx [0\ \ \ 0\ \ \ 0\ \ \ 0]^T$ and hence the system is in static equilibrium. The initial contact force is $\boldsymbol{k_1(z_1(0) - z_w(0)) = 90}$ **N =** $\boldsymbol{f_a}$**.**

b) The Simulink diagram is below. The wire input $z_w(t)$ is a sine wave with bias of $z_w(0)$. The SSR matrices A, B, C, and D are set in the State-Space block (see Problem 5.33 for C and D)



Plots of $y_1 = k_1(z_1 - z_w)$ = shoe contact force and $y_2 = z_1 - z_2$ are below. Since contact force remains > 0, the head remains in contact with the overhead wire (i.e., $z_1 - z_w > 0$).





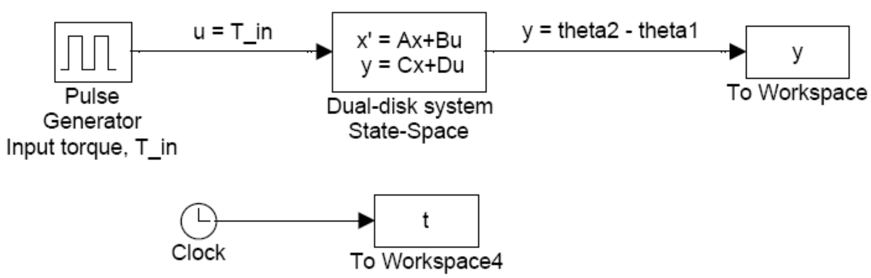
6.21 The SSR for the dual-disk system (see Problem 5.29) is

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -k/J_1 & -b/J_1 & k/J_1 & 0 \\ 0 & 0 & 0 & 1 \\ k/J_2 & 0 & -k/J_2 & -b/J_2 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \\ -1/J_1 \\ 0 \\ 1/J_2 \end{bmatrix}u \quad \text{and} \quad y = \begin{bmatrix} -1 & 0 & 1 & 0 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \end{bmatrix}u$$

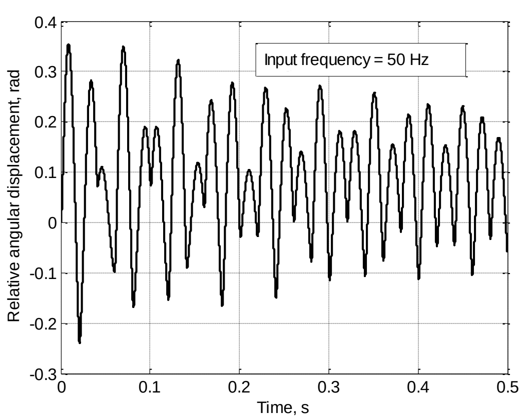
The state variables are $x_1 = \theta_1$, $x_2 = \dot{\theta}_1$, $x_3 = \theta_2$, and $x_4 = \dot{\theta}_2$. The input is $u = T_{in}(t)$. Using the numerical parameters the state equation becomes

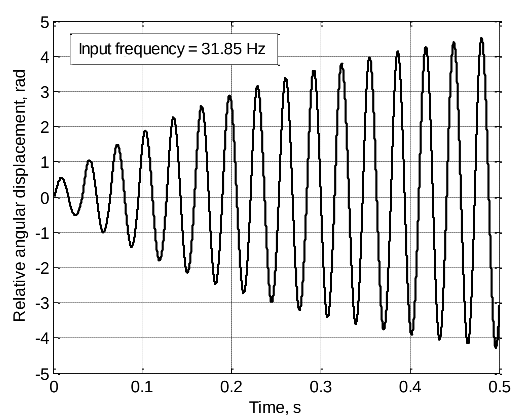
$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -20{,}000 & -6.6667 & 20{,}000 & 0 \\ 0 & 0 & 0 & 1 \\ 20{,}000 & 0 & -20{,}000 & -6.6667 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \\ -3.3333 \\ 0 \\ 3.3333 \end{bmatrix}u$$

The Simulink diagram is below. The Pulse Generator is used to build the periodic input torque pulse, $T_{in}(t)$. After opening the Pulse Generator block, the `Period` is set to either 0.02 s (50 Hz) or 0.0314 s (31.85 Hz). The `Amplitude` is set to 5,500 (N) and the `Pulse Width` is set to 10% (percentage of the period).



Plots of relative angular displacement $y = \theta_2 - \theta_1$ for the two input frequencies are below. The relative angular displacement appears to be "random" for the 50-Hz input pulse, while the amplitude of the angular twist steadily grows for the 31.85-Hz input (the input is near the resonant frequency of the mechanical system – to be discussed in Chapter 9).





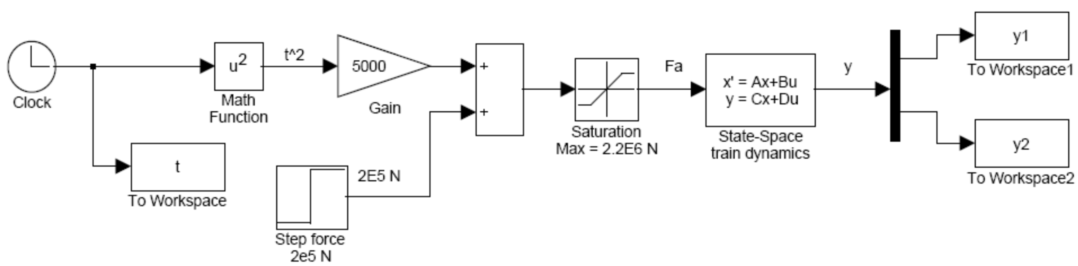
6.22 The SSR of the railroad-car system (see Problem 5.31) is

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -k/m_1 & -(b+b_r)/m_1 & k/m_1 & b/m_1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ k/m_2 & b/m_2 & -2k/m_2 & -(2b+b_r)/m_2 & k/m_2 & b/m_2 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & k/m_3 & b/m_3 & -k/m_3 & -(b+b_r)/m_3 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 1/m_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u$$

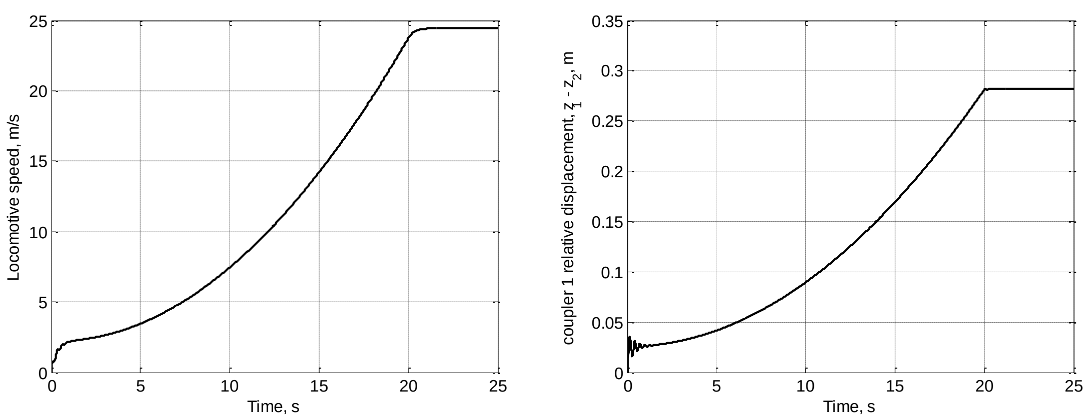
$$\mathbf{y} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u$$

The state variables are $x_1 = z_1$, $x_2 = \dot{z}_1$, $x_3 = z_2$, $x_4 = \dot{z}_2$, $x_5 = z_3$, and $x_6 = \dot{z}_3$. The input is $u = F_a$. The output variables are $y_1 = \dot{z}_1 = x_2$ (locomotive speed) and $y_2 = z_1 - z_2 = x_1 - x_3$.

The Simulink model of the locomotive system is below. Executing the Mfile `run_Prob22.m` sets the system parameters and the SSR matrices, executes the Simulink model, and plots the output variables.



The two outputs (locomotive speed and relative displacement $z_1 - z_2$) are plotted below. The final speed of the locomotive is $\dot{z}_1 =$ **24.44 m/s = 54.7 mph.**



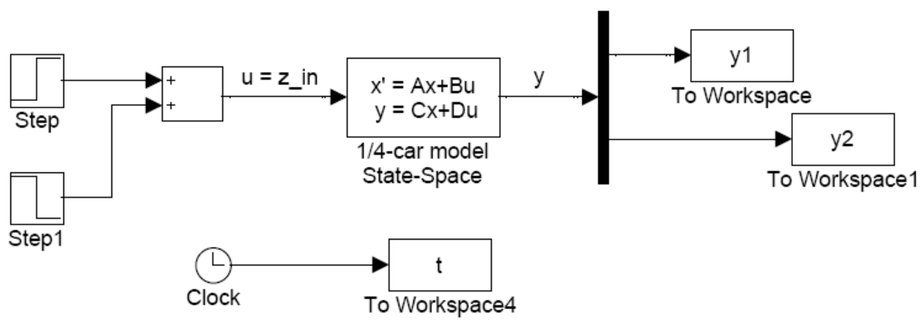
6.23 The SSR of the 1/4-car system (see Problem 2.30) is

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -k_1/m_1 & -b/m_1 & k_1/m_1 & b/m_1 \\ 0 & 0 & 0 & 1 \\ k_1/m_2 & b/m_2 & -(k_1+k_2)/m_2 & -b/m_2 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ k_2/m_2 \end{bmatrix}u$$

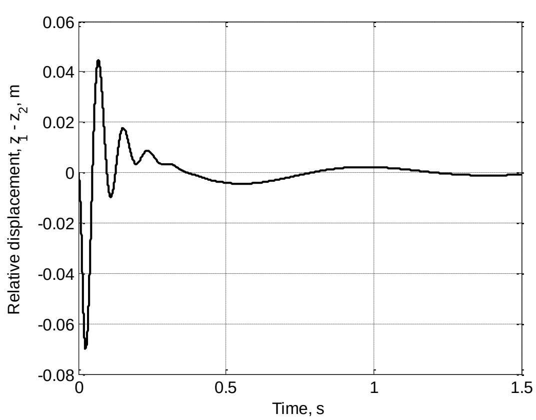
$$\mathbf{y} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}u$$

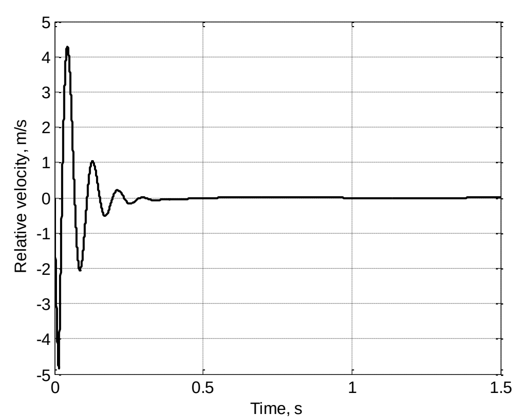
The state variables are $x_1 = z_1$, $x_2 = \dot{z}_1$, $x_3 = z_2$, and $x_4 = \dot{z}_2$. The input is $u = z_{in}(t)$. The output variables are suspension travel $y_1 = z_1 - z_2 = x_1 - x_3$ and relative velocity $y_2 = \dot{z}_1 - \dot{z}_2 = x_2 - x_4$

The Simulink model of the ¼-car system is below. The input bump $u = z_{in}$ is a pulse with amplitude of 0.1 m and duration of (0.4 m/26.82 m/s) = 0.0149 s. The pulse is created by adding two step functions with magnitudes +0.1 m and -0.1 m (delayed by 0.0149 s). Executing the Mfile `run_Prob23.m` sets the system parameters and the SSR matrices, executes the Simulink model, and plots the output variables.

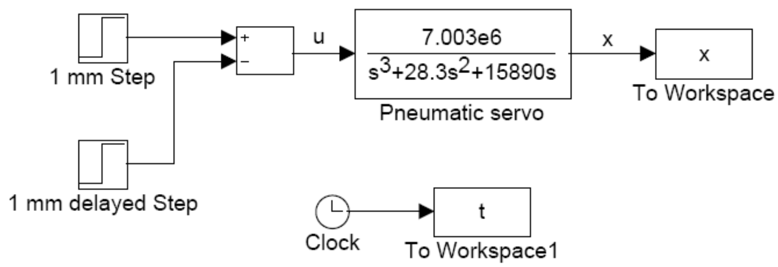


The two outputs (relative displacement and velocity are plotted below.



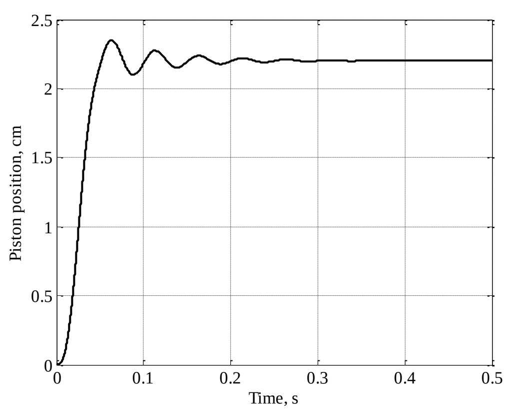


6.24 The Simulink model of the pneumatic servo is below

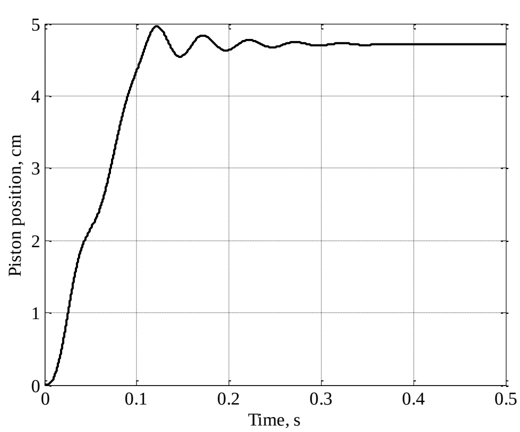


Note that the 1-mm pulse input is produced by adding together two step input functions: the first step has magnitude of 1E-3 m (1 mm) with step-up time at t = 0, the second step input also has a magnitude of 1E-3 m but with a step-up time of t = 0.05 sec.

a) The position response $x(t)$ to the 1-mm pulse input (pulse lasts 0.05 s) is below:



b) Next, the pulse time is increased from 0.05 s until the peak response of $x(t)$ equals 5 cm which is the maximum stroke of the piston (note the cylinder length is 10 cm). The maximum pulse duration is found to be **0.107 s**. The response $x(t)$ for the maximum pulse is below:

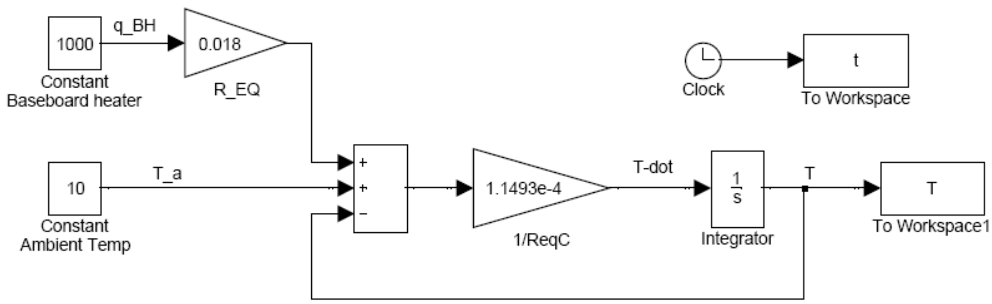


6.25 The mathematical model of the thermal system is (see Example 4.7)

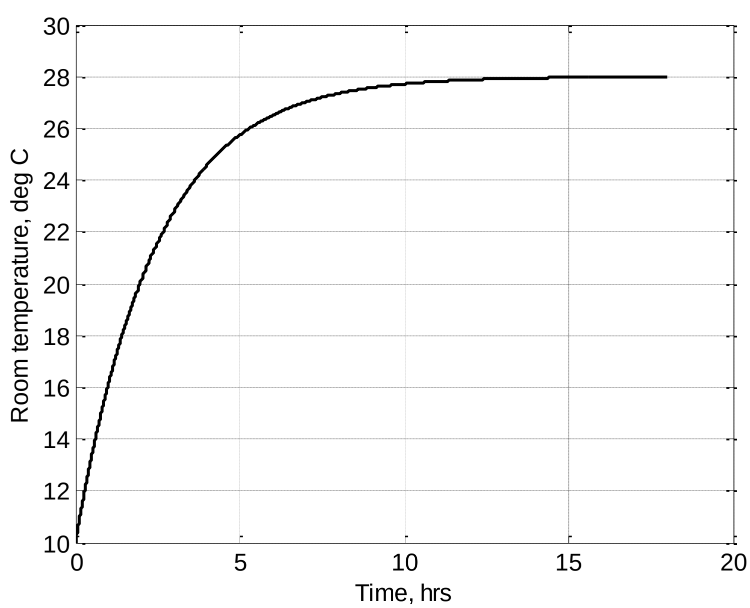
$$R_{EQ}C\dot{T} + T = R_{EQ}q_{BH} + T_a$$

The equivalent thermal resistance of all surfaces is $\frac{1}{R_{EQ}} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_6} = \sum_{i=1}^{6} \frac{1}{R_i}$

Or, R_{EQ} = 0.018014 deg C-s/J. The Simulink model of the thermal system is below. We must use the integrator-block method since the room has a non-zero initial temperature (10 deg C).



The room temperature response to a 1 kW heater input is plotted below. The final room temperature after 18 hrs is **27.99 deg C or 82.38 deg F.**



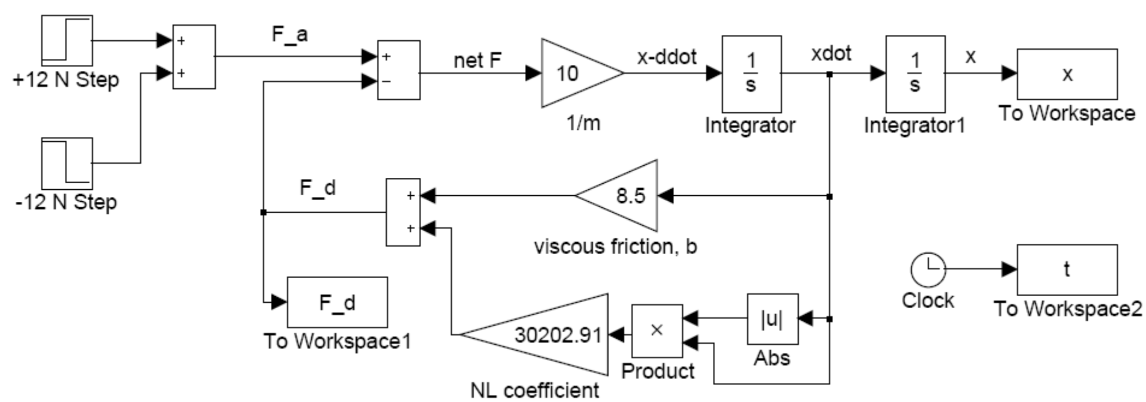
6.26 The mathematical model of the incompressible hydraulic damper is (see Problem 4.15a)

$$m\ddot{x} + b\dot{x} + \frac{\rho A^3}{2C_d^2 A_0^2}\dot{x}|\dot{x}| = F_a \quad \text{or} \quad \ddot{x} = \frac{-b}{m}\dot{x} - \frac{\rho A^3}{2mC_d^2 A_0^2}\dot{x}|\dot{x}| + \frac{1}{m}F_a$$

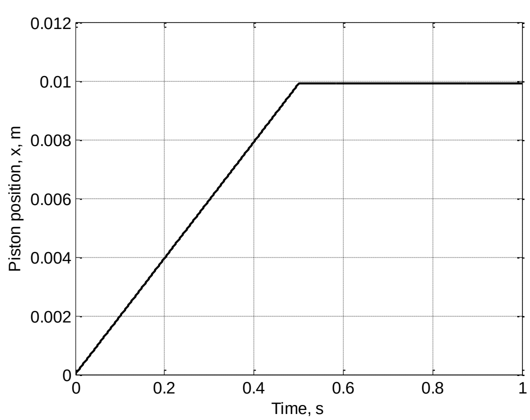
The Simulink model of the hydraulic system is on the next page. We must use the integrator-block method since the hydraulic damper is nonlinear. The nonlinear coefficient in the Simulink model is

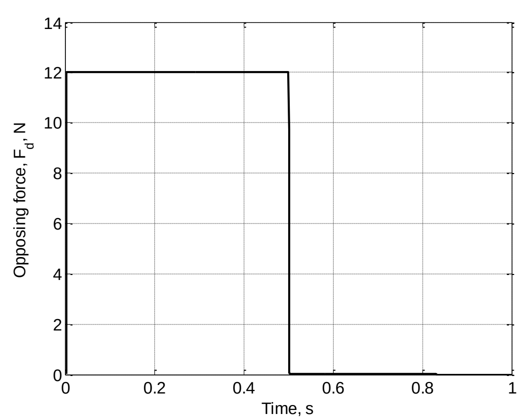
$$\frac{\rho A^3}{2C_d^2 A_0^2} = 30{,}202.91 \text{ kg/m}^2$$

The input pulse force F_a is created by adding two step functions (one delayed by 0.5 s). `ode4` is the fixed-step integration method with a step size of 10^{-4} s.



The piston position $x(t)$ and opposing force $F_d(t)$ responses to the pulse input are below. Note that the piston ramps up from zero to 0.01 m at a nearly constant velocity. The opposing force F_d steps up immediately to 12 N (to oppose F_a = 12 N) and then steps down to zero at t = 0.5 s when the input pulse force goes to zero.





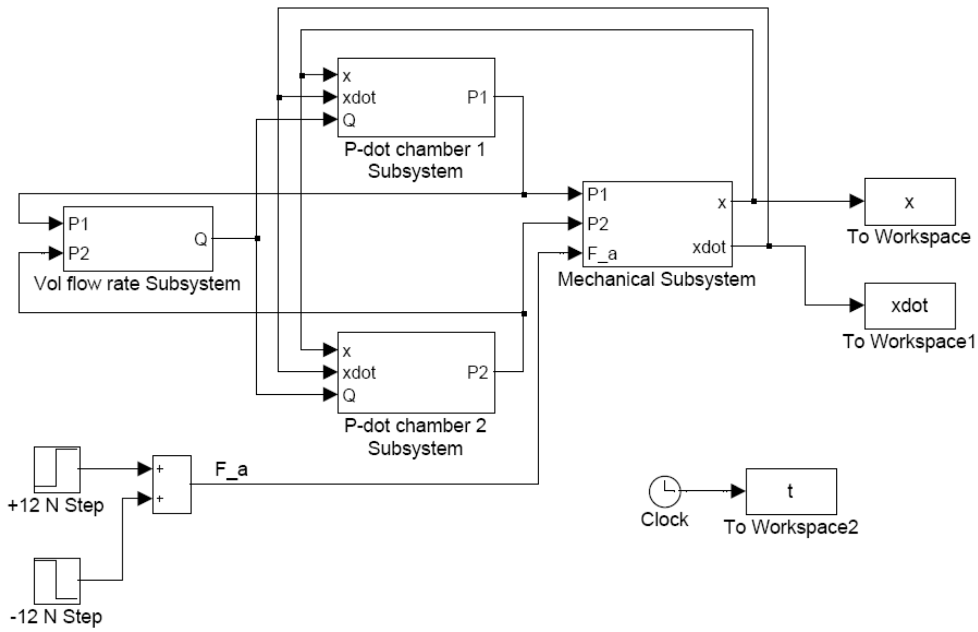
6.27 The mathematical model of the compressible hydraulic damper is (see Problem 4.15b)

$$\dot{P}_1 = \frac{\beta}{V_0 + Ax}\left(C_d A_0 \,\mathrm{sgn}(P_2 - P_1)\sqrt{\frac{2}{\rho}\left|P_2 - P_1\right|} - A\dot{x} \right) = \frac{\beta}{V_0 + Ax}(Q - A\dot{x})$$

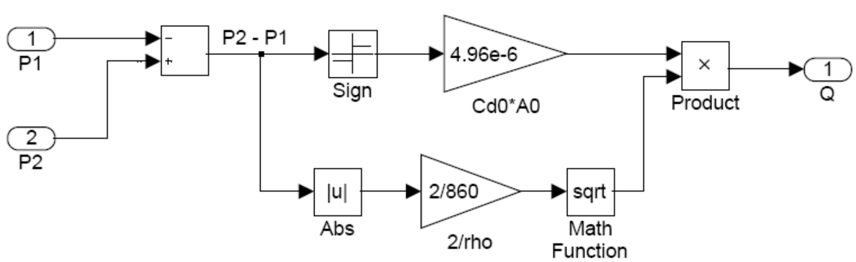
$$\dot{P}_2 = \frac{\beta}{V_0 - Ax}\left(-C_d A_0 \,\mathrm{sgn}(P_2 - P_1)\sqrt{\frac{2}{\rho}\left|P_2 - P_1\right|} + A\dot{x} \right) = \frac{\beta}{V_0 - Ax}(-Q + A\dot{x})$$

$$m\ddot{x} + b\dot{x} = (P_1 - P_2)A + F_a$$

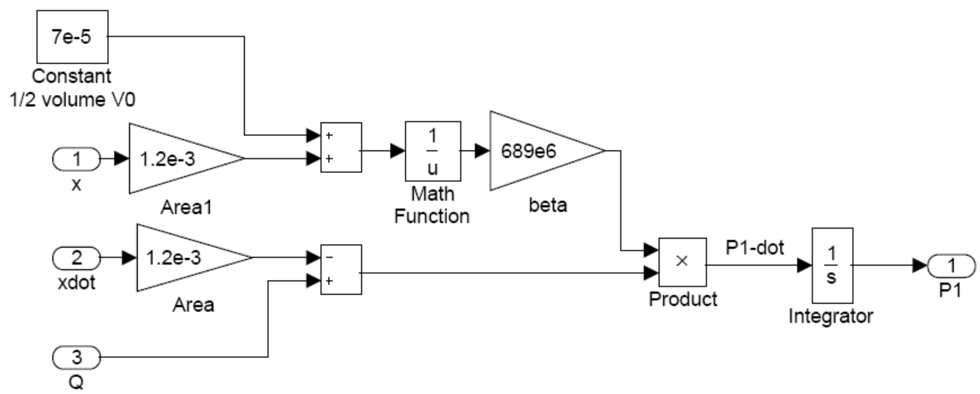
The Simulink model of the integrated hydraulic system is shown below.



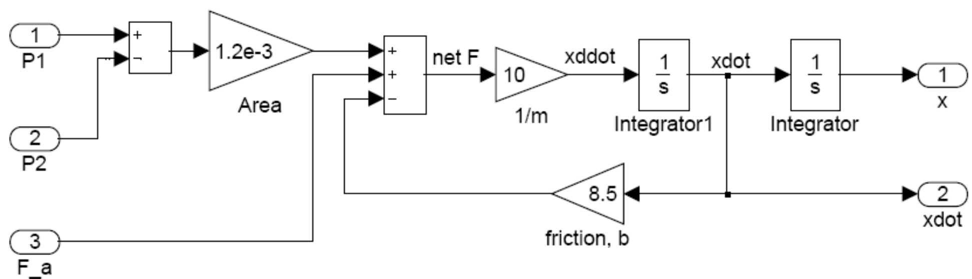
The volumetric flow-rate subsystem is shown below. Note that it matches the computation of flow rate Q in the mathematical model.



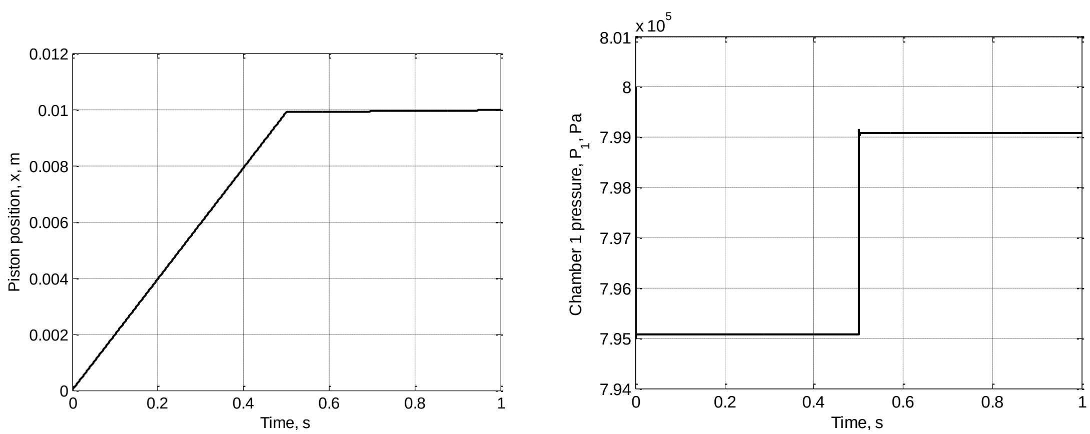
The subsystem for pressure-rate of chamber 1 is shown below (the subsystem for chamber 2 is nearly identical). Note that it matches the computation of $\dot{P}_1$ in the mathematical model.



Finally, the mechanical subsystem is shown below. It matches the mechanical modeling equation.



The piston position $x(t)$ and pressure $P_1(t)$ responses to the pulse force input are below. Note that the piston ramps up from zero to 0.01 m at a nearly constant velocity (very similar to the incompressible case from Problem 6.26). Pressure changes nearly instantly from 8E6 Pa to 7.95E6 Pa at time t = 0, and then to 7.991E6 Pa at time t = 0.5 s when the applied force goes to zero. The simplified (incompressible) model in Problem 6.26 is much easier to construct and it accurately depicts the hydraulic damper's dynamics.



6.28 The mathematical model of the electro-pneumatic system is (see Problem 4.20)

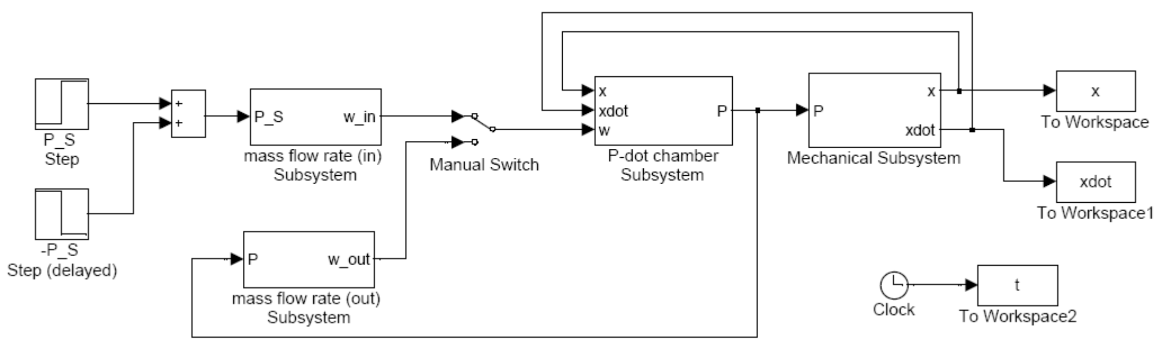
$$\dot{P} = \frac{nRT}{V_0 + A_1 x}\left(w_{\text{in}} - \frac{P}{RT} A_1 \dot{x} \right) \text{(supply)} \ \ \textbf{or} \ \ \dot{P} = \frac{nRT}{V_0 + A_1 x}\left(-w_{\text{out}} - \frac{P}{RT} A_1 \dot{x} \right) \text{(exhaust)}$$

$$m\ddot{x} + b\dot{x} = PA_1 - P_{\text{atm}} A_2 - F_L \quad \text{where} \ F_L = 4000(1 - e^{-500x}) - 20{,}000x \ \text{(N)}$$

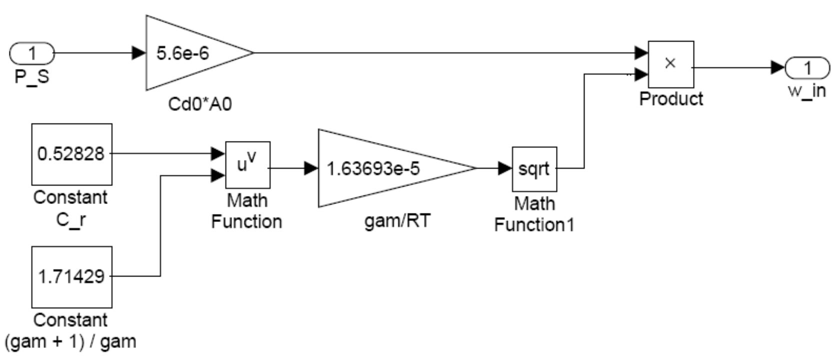
The choked mass-flow rate (in or out) is

$$w_{\text{in}} = C_d A_0 P_S \sqrt{\frac{\gamma}{RT_S} C_r^{\frac{\gamma+1}{\gamma}}} \ \text{(flow in)} \quad \textbf{or} \quad w_{\text{out}} = C_d A_0 P \sqrt{\frac{\gamma}{RT} C_r^{\frac{\gamma+1}{\gamma}}} \ \text{(flow out)}$$

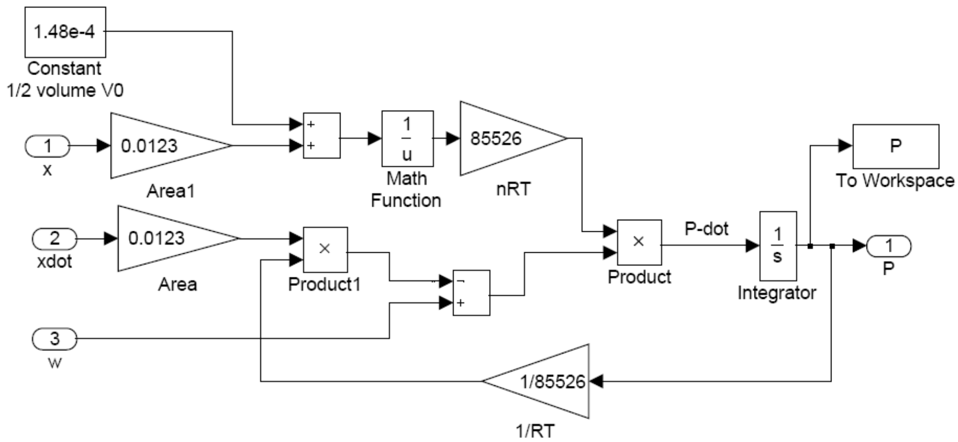
The Simulink model of the integrated hydraulic system is below. Note that there is a Manual Switch for setting the mass-flow rate to w_{in} (supply) or w_{out} (exhaust).



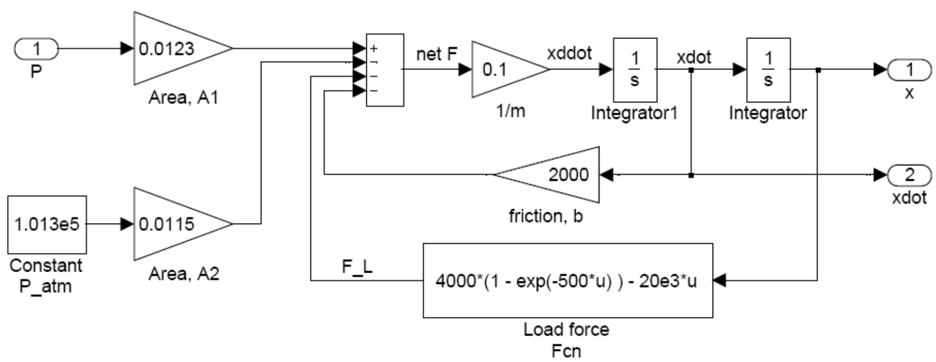
The mass-flow rate subsystem (supply) is shown below. Note that it matches the computation of flow rate w_{in} in the mathematical model (choked flow). The exhaust mass-flow rate is similar.



The subsystem for pressure-rate is on the next page. Note that it matches the computation of $\dot{P}$ in the mathematical model.



Finally, the mechanical subsystem is shown below. It matches the mechanical modeling equation. Note that the User Defined function `Fcn` is used to compute the load force.



The piston position $x(t)$ and pressure $P(t)$ responses to the supply pressure (pulse) input are below. The highest chamber pressure is $4.345(10^5)$ Pa, and therefore the low/high pressure ratio across the input valve is 0.457 which is less than the critical pressure ratio (0.528) for choked flow. Hence, the supply mass-flow rate is always choked.