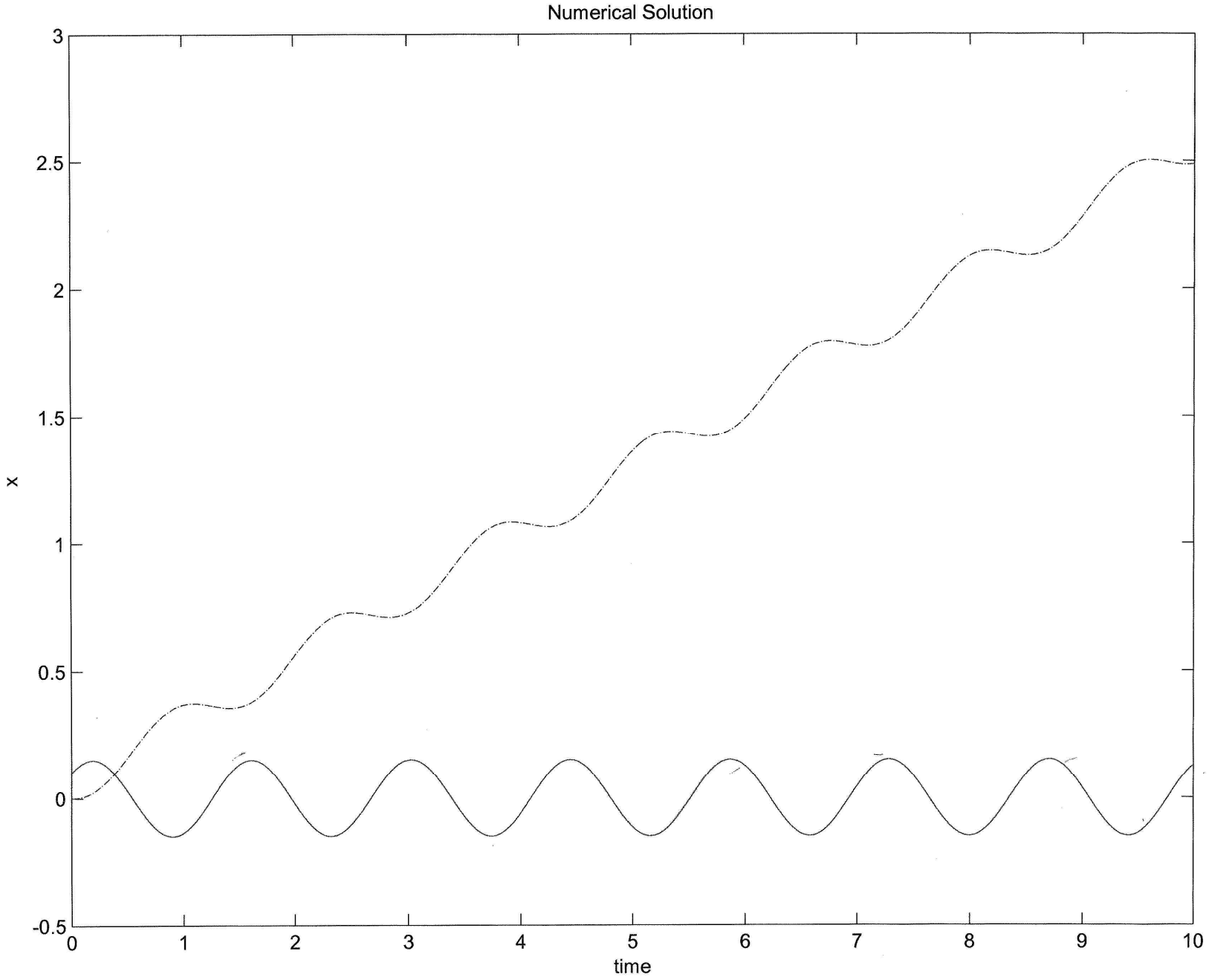
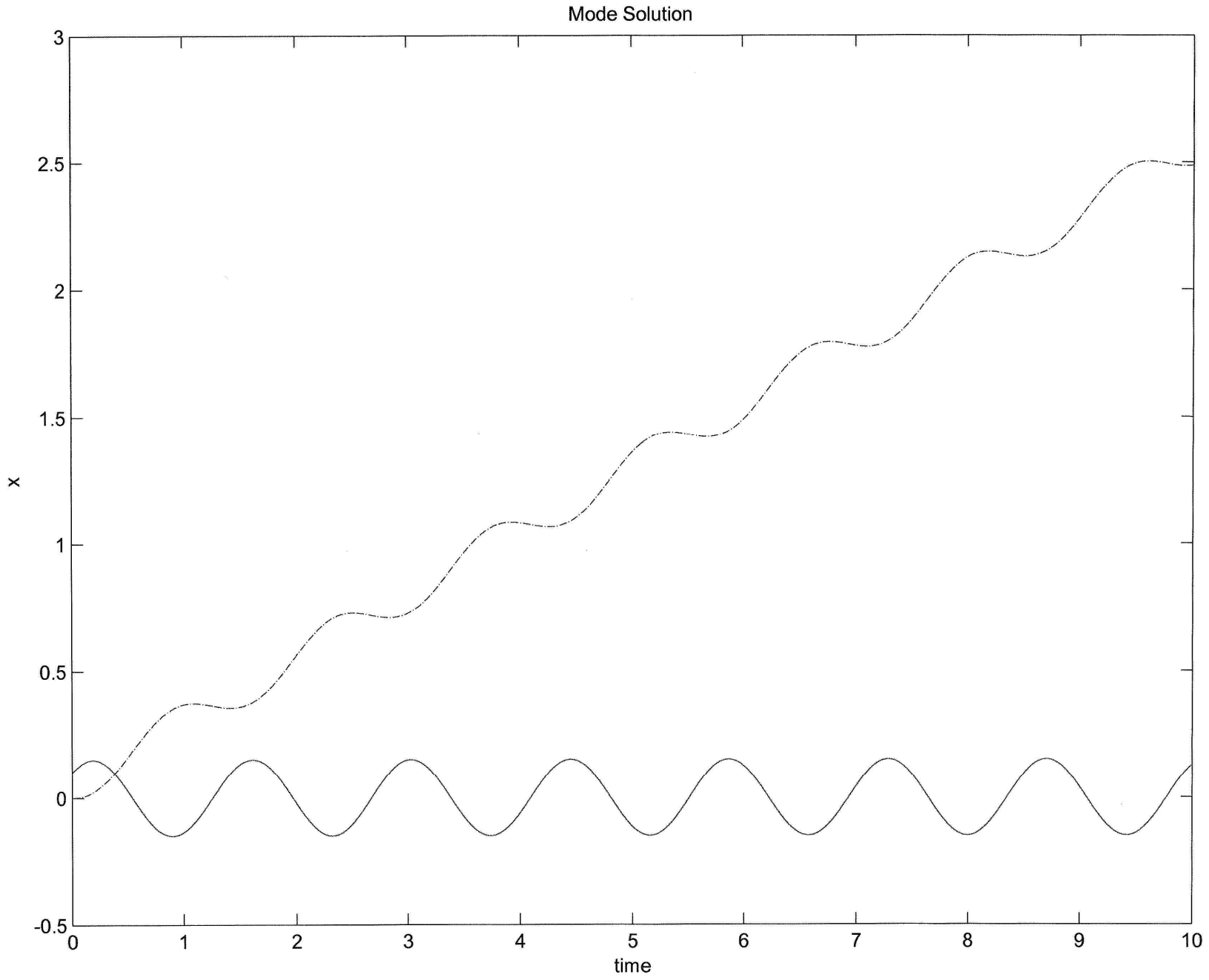
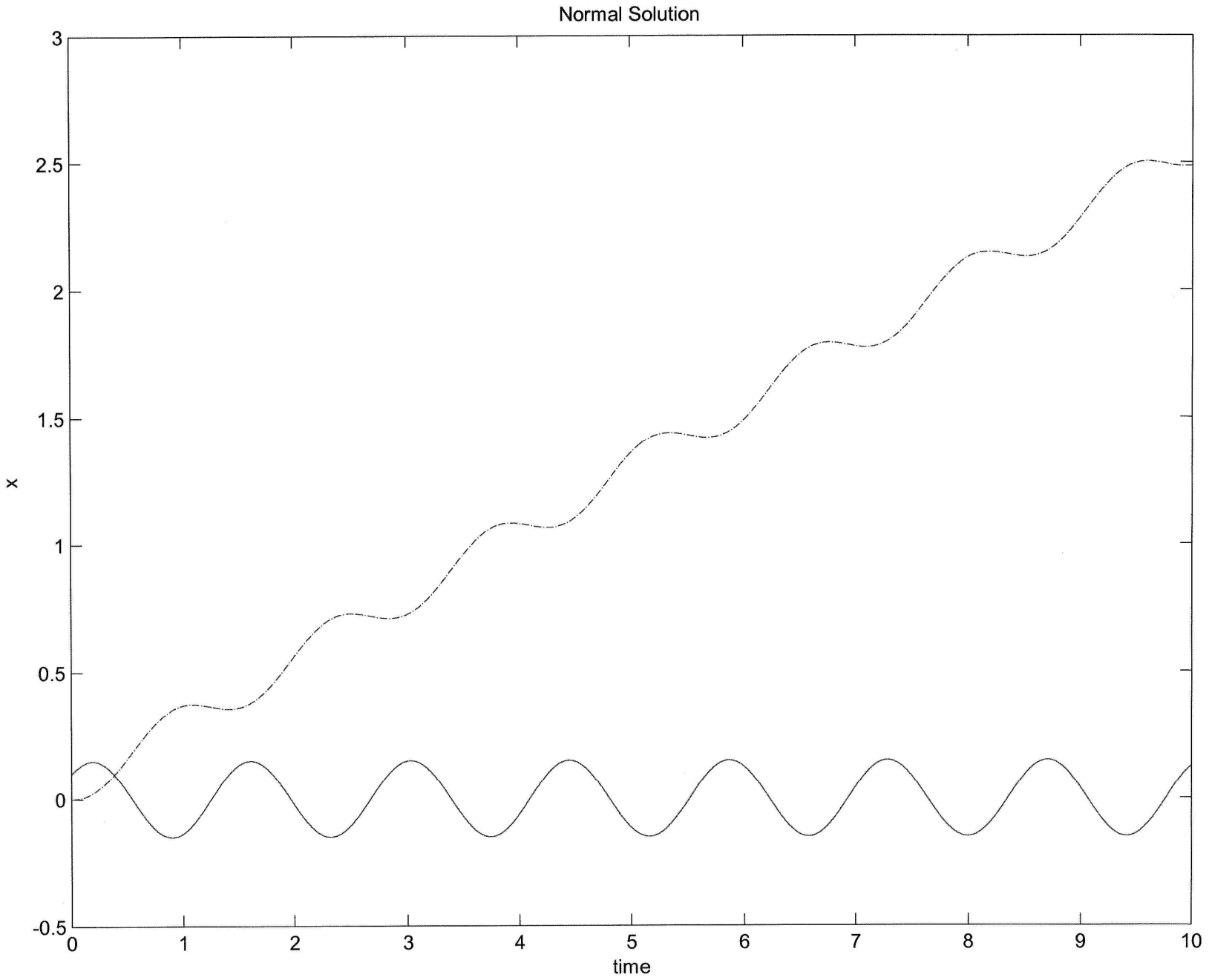


Student Version of MATLAB





```
function problem28Runner
clc
%Simulation duration
t0 = 0;
ts = 10;
n = 3;
tspan = linspace(t0,ts,10^n);

%Problem parameters
% m = 1;%prob 25
% k = 1;%prob 25
mc = 1;
mp = 1;
L= 1;
g = 9.8;

%M and K matrices
% M = [m 0 0;0 m 0;0 0 m];%prob 25
% K = [2*k -k 0;-k 2*k -k;0 -k 2*k];%prob 25
M = [L*mc 0; 0 mc];
K = [g*(mp+mc) 0; -g*mp 0];

%Eigenvectors and Eigenvalues
[V D] = eig(M^(-1)*K);

%Initial Conditions
% x0 = [1 2 3];%prob 25
% v0 = [4 5 6];%prob 25
x0 = [.1 0];
v0 = [.5 0];

%Results of one solution
xmatrix = problem28SolveNum(M,K,x0,v0,tspan);
xmatrix2 = problem28SolveMode(M,K,x0,v0,tspan)';
xmatrix3 = problem28SolveNorm(M,K,x0,v0,tspan)';

%Unpack Solution
x1 = xmatrix(:,1);
x2 = xmatrix(:,2);
% x3 = xmatrix(:,3);%prob 25

x21 = xmatrix2(:,1);
x22 = xmatrix2(:,2);
% x23 = xmatrix2(:,3);%prob 25

x31 = xmatrix3(:,1);
x32 = xmatrix3(:,2);
% x33 = xmatrix3(:,3);%prob 25

%Plot Solution
figure(1)
plot(tspan,x1);
```

```
hold on
plot(tspan,x2,'-.r');
% plot(tspan,x3,'--g');%prob 25
hold off
title('Numerical Solution');
xlabel('time')
ylabel('x')

figure(2)
plot(tspan,x21);
hold on
plot(tspan,x22,'-.r');
% plot(tspan,x23,'--g');%prob 25
hold off
title('Mode Solution');
xlabel('time')
ylabel('x')

figure(3)
plot(tspan,x31);
hold on
plot(tspan,x32,'-.r');
% plot(tspan,x33,'--g');%prob 25
hold off
title('Normal Solution');
xlabel('time')
ylabel('x')
end
```

```
function xmatrix = problem28SolveNorm(M,K,x0,v0,tspan)

%Eigenvectors and Eigenvalues
[V D] = eig(M^(-1/2)*K*M^(-1/2));
w = sqrt(diag((D)));
Ai = linsolve(V, x0');
Bi = linsolve(V, v0');

%Linear combinations of eigenvector solutions
qmatrix = zeros(length(w),length(tspan));
for i = 1:length(w)
    for j = 1:length(w)
        if(w(i)~=0)
            qmatrix(j,:) = qmatrix(j,:) + (Ai(i).*cos(w(i).*tspan) + (Bi(i)/w(i)).*(sin(w(i).*tspan))).*V(j,i);
        else
            qmatrix(j,:) = qmatrix(j,:) + (Ai(i) + Bi(i).*tspan).*V(j,i);
        end
    end
end

xmatrix = M^(-1/2)*qmatrix;
end
```

$x_0 = M^{1/2}x_0$; $v_0 = M^{1/2}v_0$; ← This line needs to be here, it didn't matter for my M, K, but would in general

odd

↳ $M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ in this case.

```
function xmatrix = problem28SolveMode(M,K,x0,v0,tspan)

%Eigenvectors and Eigenvalues
[U D] = eig(M^(-1)*K);

w = sqrt(diag((D)));

Ai = linsolve(U, x0');
Bi = linsolve(U, v0');


%Linear combinations of eigenvector solutions
xmatrix = zeros(length(w),length(tspan));
for i = 1:length(w)
    for j = 1:length(w)
        if(w(i)~=0)
            xmatrix(j,:) = xmatrix(j,:) + (Ai(i).*cos(w(i).*tspan) + (Bi(i)/w(i)).*(sin(w↙
(i).*tspan))).*U(j,i);
        else
            xmatrix(j,:) = xmatrix(j,:) + (Ai(i) + Bi(i).*tspan).*U(j,i);
        end
    end
end
end
```

```
function zdot = problem28RHS(t,z,p)
M = p.M;
K = p.K;

L = length(M);

x = z(1:L);

xdot = z(L+1:end);

dot = xdot';

ddot = -((M)^(-1)*(K*x))';

zdot = [dot ddot]';
end
```