

---

```

%Shishir Khanal
%ME 3350
%Date: 10/28/2021
%Purpose: To develop the plots for the follower profile and the cam
profile
%-----
clc;clear; close all;
theta1 = 0:0.01:(pi/2);
dwell1 = 1 + 0*theta1;
plot(theta1,dwell1)
hold on
%plot of dwell from 0 to 90 degrees
theta2 = (pi/2):0.01:(5*pi/6);
followerrise = evaluateRise(theta2);
plot(theta2,followerrise)
hold on
%plot of rise from 90 to 150 degrees
theta3 = (5*pi/6):0.01:(11*pi/9);
dwell2 = 4 + 0*theta3;
plot(theta3,dwell2);
hold on
%plot of dwell from 150 to 220 degrees
theta4 = (11*pi/9):0.01:(5*pi/3);
followerreturn = evaluateReturn(theta4);
plot(theta4,followerreturn)
hold on
%plot of return from 220 to 300 degrees
theta5 = (5*pi/3):0.01:(2*pi);
dwell3 = 1 + 0*theta5;
plot(theta5,dwell3)
createplot(theta5,dwell3,'theta[0,2*pi]->','y[theta]->','Plot of the
Follower Profile')
%plot of dwell from 300 to 360 degrees
%plot of the follower profile
%-----
r = [dwell1 followerrise dwell2 followerreturn dwell3];
theta = [theta1 theta2 theta3 theta4 theta5];
figure(2)
polarplot(theta,r)
%Plot of Cam Profile
%-----
velocity = diff(r);
%Numerical differentiation function to evaluate first derivative of
the
%follower profile
figure(3)
subplot(2,1,1)
createplot(theta(1:630),velocity,'theta[0,2*pi]','velocity[theta]','Numerical
Plot of Velocity Profile of the Cam')
subplot(2,1,2)
velocity2 = [0*theta1 velocityRise(theta2) 0*theta3
velocityReturn(theta4) 0*theta5];

```

---

---

```

createplot(theta,velocity2,'theta[0,2*pi)','velocity[theta]','Plot of
Velocity Profile of the Cam using Equations')
%plot of velocity of the Cam
%-----
acceleration = diff(r,2);
%Numerical differentiation function to evaluate second derivative of
the
%follower profile
figure(4)
subplot(2,1,1)
createplot(theta(1:629),acceleration,'theta[0,2*pi)','acceleration[theta]','Numeri
Plot of Acceleration Profile of the Cam')
subplot(2,1,2)
acceleration2 = [0*theta1 accelerationRise(theta2) 0*theta3
accelerationReturn(theta4) 0*theta5];
createplot(theta,acceleration2,'theta[0,2*pi)','acceleration[theta]','Plot
of Acceleration Profile of the Cam using Equations')
%-----
jerk = diff(r,3);
%Numerical differentiation function to evaluate third derivative of
the
%follower profile
figure(5)
subplot(2,1,1)
createplot(theta(1:628),jerk,'theta[0,2*pi)','Jerk[theta]','Numerical
Plot of Jerk Profile of the Cam')
subplot(2,1,2)
jerk2 = [0*theta1 jerkRise(theta2) 0*theta3 jerkReturn(theta4)
0*theta5];
createplot(theta,jerk2,'theta[0,2*pi)','Jerk[theta]','Plot of Jerk
Profile of the Cam using Equations')
%-----
x= r.*cos(theta);
y = r.*sin(theta);
xlswrite('Camcoordinates.xlsx',[x(:),y(:)])
%Convert the polar coordinate of Cam profile to cartesian coordinates
and
%save it in excel file

%-----
%Functions:
%to evaluate Displacements
function y_rise = evaluateRise(theta)
%Function to evaluate the rise of the follower
y_rise = 1 + (405/4)*((2*theta - pi)/pi).^3 - (3645/16)*((2*theta -
pi)/pi).^4 + (2187/16)*((2*theta - pi)/pi).^5;
end

function y_return = evaluateReturn(theta)
%Function to evaluate the return of the follower
y_return = 4 - (30*((9*theta - (11*pi))/(4*pi)).^3) + (45*((9*theta -
(11*pi))/(4*pi)).^4) - (18*((9*theta - (11*pi))/(4*pi)).^5);
end
%-----

```

---

---

```

%to evaluate velocity
function velocityrise = velocityRise(theta)
%Function to evaluate the rise velocity of the follower
angularvel = 31.43;
velocityrise = ((1215/(2*(pi^3)))*((2*theta) - pi).^2 - (3645/
(2*(pi^4)))*((2*theta) - pi).^3 + (10935/(8*(pi^5)))*((2*theta) -
pi).^4)*angularvel;
end

function velocityreturn = velocityReturn(theta)
%Function to evaluate the return velocity of the follower
angularvel = 31.43;
velocityreturn = ((-405/(32*(pi^3)))*((9*theta) - 11*pi).^2 + (405/
(64*(pi^4)))*((9*theta) - 11*pi).^3 - (405/(512*(pi^5)))*((9*theta) -
11*pi).^4)*angularvel;
end

%-----
% to evaluate acceleration
function acelrise = accelerationRise(theta)
%Function to evaluate the rise acceleration of the follower
angularvel = 31.43;
acelrise = ((2430/(pi^3))*((2*theta) - pi) - (10935/(pi)^4)*((2*theta)
- pi).^2 + (10935/(pi)^5)*((2*theta) - pi).^3)*(angularvel)^2;
end

function acelreturn = accelerationReturn(theta)
%Function to evaluate the return acceleration of the follower
angularvel = 31.43;
acelreturn = ((-3645/(16*pi^3))*((9*theta - 11*pi) + (10935/
(64*pi^4))*((9*theta - 11*pi).^2 - (3645/(128*pi^5))*((9*theta -
11*pi).^3)*(angularvel)^2;
end

%-----
% to evaluate jerk
function jrise = jerkRise(theta)
%Function to evaluate the rise jerk of the follower
angularvel = 31.43;
jrise = ((4860/(pi^3)) - (43740/(pi^4))*((2*theta - pi) + (65610/
(pi^5))*((2*theta - pi).^2)*(angularvel)^3;
end

function jreturn = jerkReturn(theta)
%Function to evaluate the return jerk of the follower
angularvel = 31.43;
jreturn = ((-32805/(16*(pi)^3)) + (196830/(64*(pi)^4))*((9*theta -
11*pi) - (98415/(128*(pi)^5))*((9*theta - 11*pi).^2)*(angularvel)^3;
end

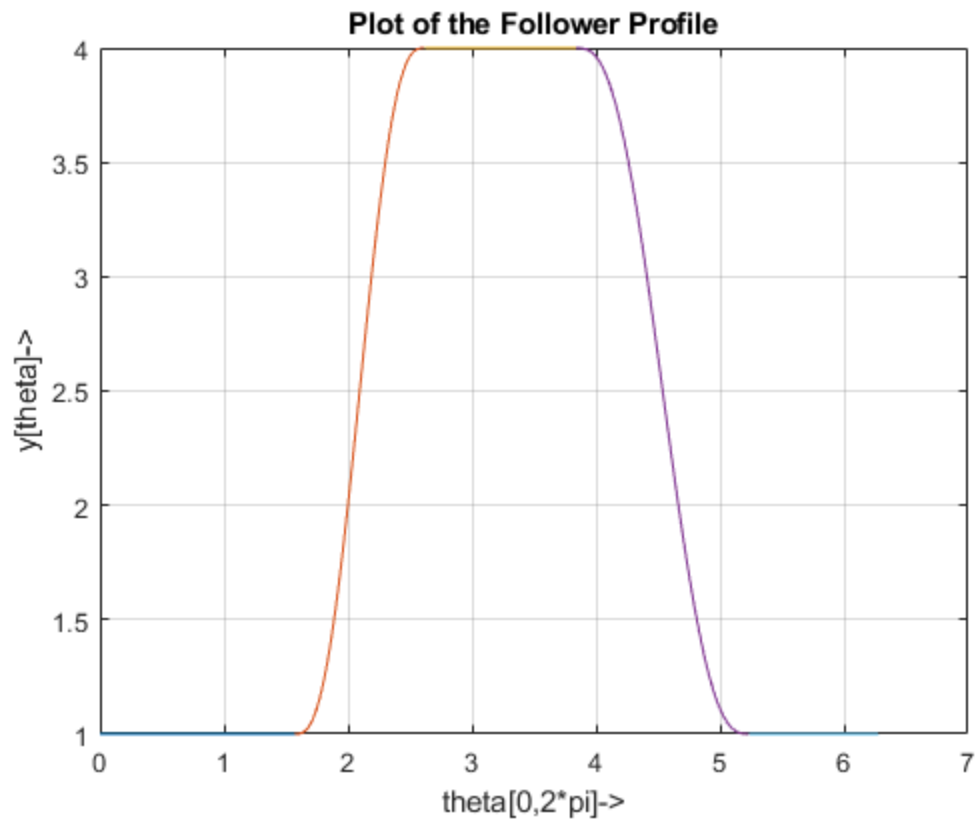
%-----
function createplot(xparam, yparam,xlbl, ylbl,plottitle)
%createplot() doesn't output anything but creates a formatted plot of
%parameters based on the given parameters
% Detailed explanation goes here
plot(xparam,yparam)
xlabel(xlbl)

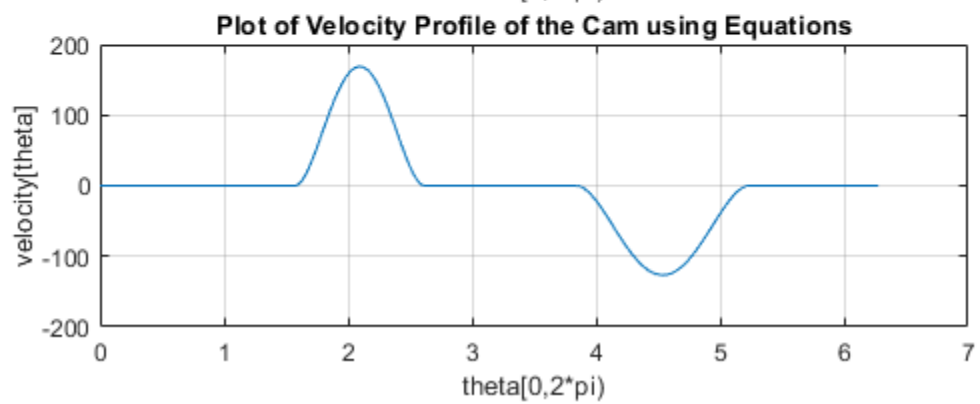
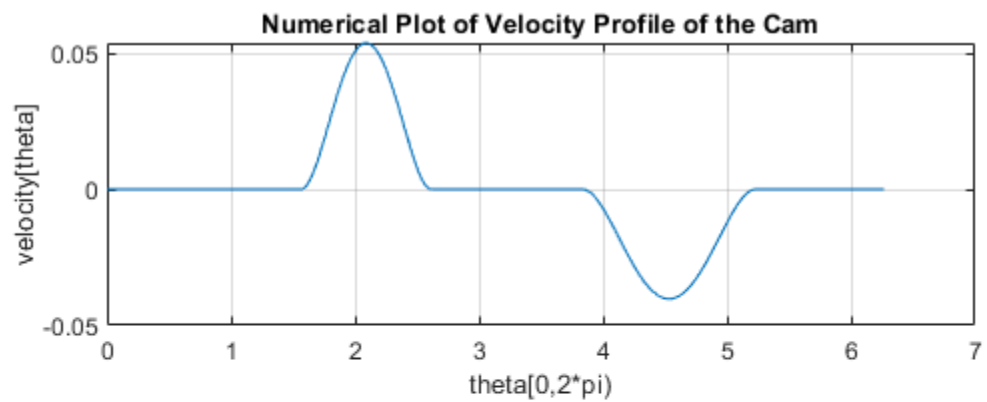
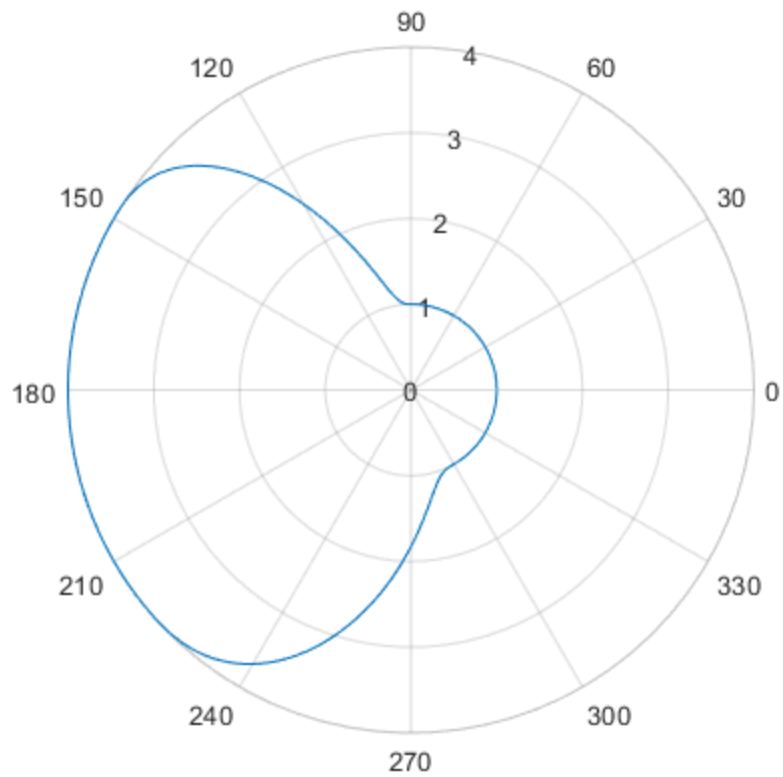
```

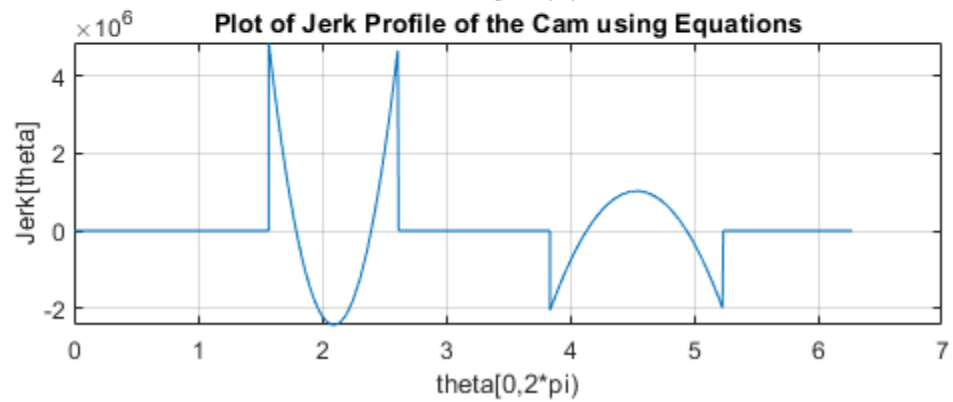
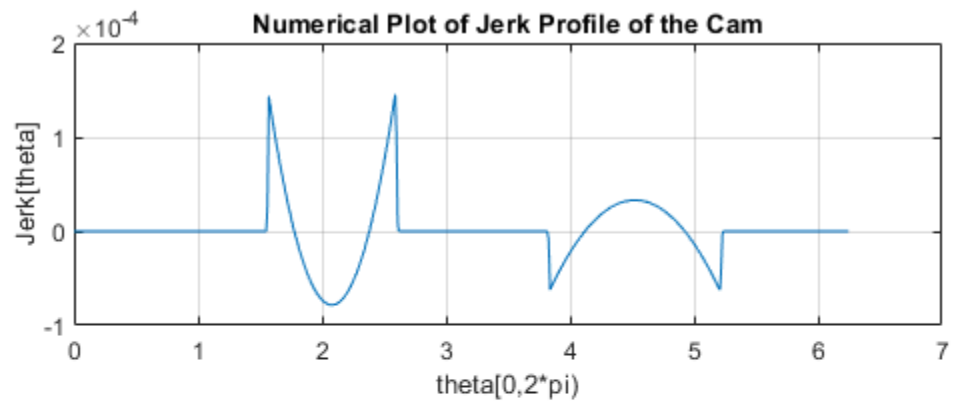
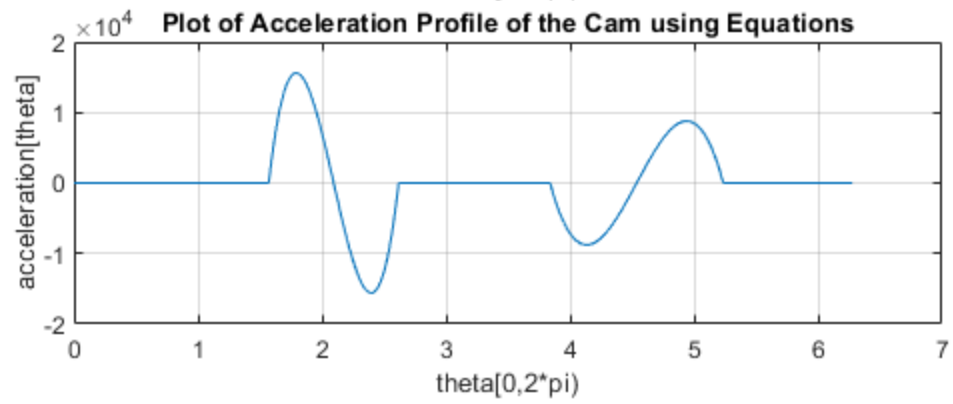
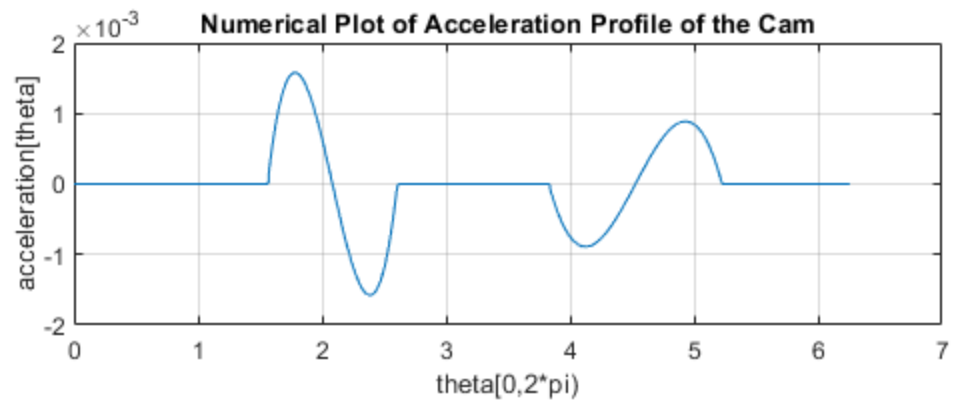
---

---

```
ylabel(ylbl)
title(plottitle)
grid on;
end
%-----
%-----
```







---

*Published with MATLAB® R2019b*