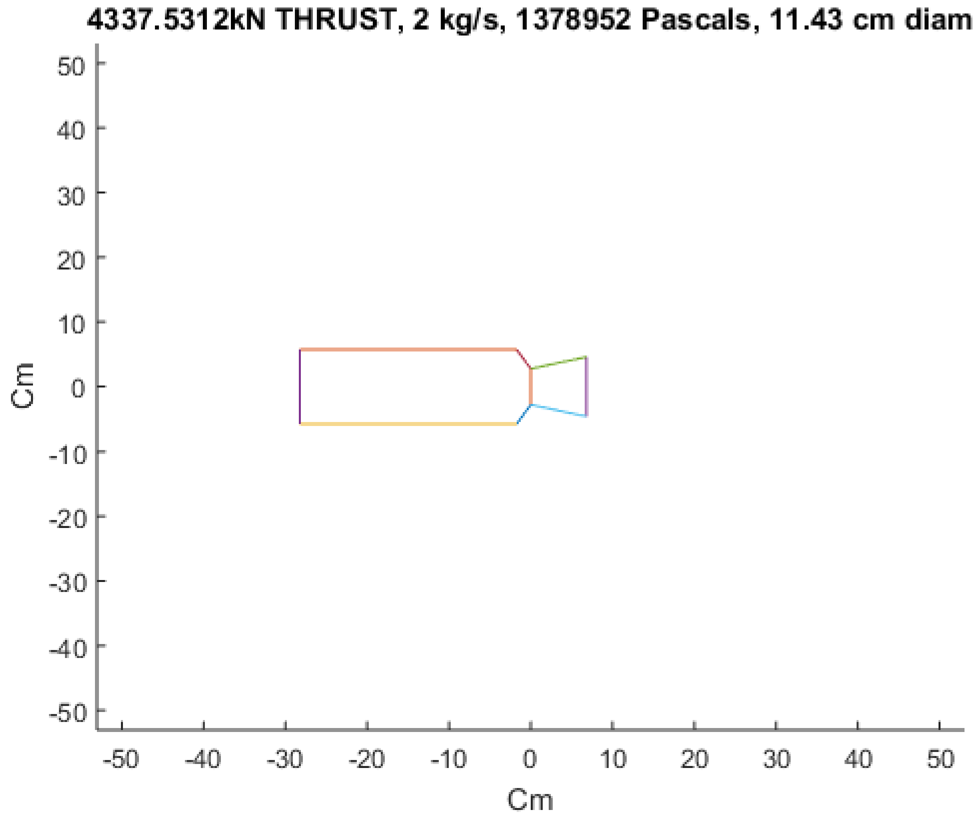
```
function [] = ethalox_engine(m_dot, Pc, Rc)
%ethalox_engine this will design an engine for you
%   it will display the thrust and design...
% m_dot = 2;
% Pc = 1.724e+6;
% Rc = 0.0760;

close all;

theta = 0.261799;
mix_ratio = 1.25;
gam = 1.2;      % specific heat ratio
Tc = 3060;      % flame temperature, Kelvin
M = 22.9/1000;  % molecular weight km/mol
Pa = 101325;    % atmospheric pressure
R_un = 8.314;   % universal gas constant
Pe = Pa;        % assuming nozzle is designed properly for the
 atmosphere
L_star = 50 * 0.0254; % combustion length, standard value 50 inches
beta = 1.0472;  % beta angle

% pressure and temperature at the throat
Pt = Pc * ((1+((gam-1)/2))^((-gam)/(gam-1)));
Tt = Tc * ((1+((gam-1)/2))^-1);
% throat area
At = (m_dot/Pt)*sqrt((R_un*Tt)/(M*gam));
At_radius = sqrt(At/pi)*100;
% mach value
Me = sqrt((2/(gam-1)) * (((Pc/Pa)^((gam-1)/gam))-1));
% exit area
Ae = (At/Me)*(((1+(((gam-1)/2)*Me*Me))/((gam+1)/2))^((gam+1)/
(2*(gam-1))));
Ae_radius = sqrt(Ae/pi)*100;
% exit velocity
Ve = sqrt( (2*(gam/(gam-1))) * (R_un * (Tc/M)) * (1-((Pe/Pc)^((gam-1)/
gam))));
% thrust
F = (m_dot*Ve) + ((Pe-Pa)*Ae);

% nozzle length
Ln = ((Ae_radius - At_radius)/sin(theta))*cos(theta);

% find the proper chamber length using the characteristic chamber
 length
% method. Combustion cross-sectional area >= 3*At
Lc = (L_star * At) / (pi*(Rc^2)*1.1);
conv_len = ((Rc-(At_radius/100))/tan(beta)) / cos(beta);


%print out the engine calcs
figure;
hold on;
```

```
ax.XAxisLocation = 'origin';
ax.YAxisLocation = 'origin';  % setting y axis location to origin
title([num2str(F) 'kN THRUST, ' num2str(m_dot) ' kg/s, ' num2str(Pc) '
 Pascals, ' num2str(Rc*200) ' cm diam'])
ylabel('Cm')
xlabel('Cm')
%axis([-0.5*Ln 1.5*Ln -2*Ae_radius 2*Ae_radius ])
% draw the nozzle!
plot(zeros(100),linspace(-At_radius, At_radius))
plot(Ln+ zeros(100), linspace(-Ae_radius, Ae_radius))
plot(linspace(0,Ln) ,linspace(At_radius, Ae_radius))
plot(linspace(0,Ln) ,-linspace(At_radius, Ae_radius))

plot(100*linspace(-conv_len*cos(beta),0), linspace(Rc*100, At_radius))
plot(100*linspace(-conv_len*cos(beta),0), -linspace(Rc*100,
 At_radius))
plot(100*linspace(-conv_len*cos(beta)-Lc, -conv_len*cos(beta)),
 linspace(Rc*100, Rc*100))
plot(100*linspace(-conv_len*cos(beta)-Lc, -conv_len*cos(beta)), -
linspace(Rc*100, Rc*100))
plot(100*linspace(-conv_len*cos(beta)-Lc, -conv_len*cos(beta)-Lc),
 100*linspace(-Rc,Rc))

xlim([-200*Lc 200*Lc])
ylim([-200*Lc 200*Lc])

hold off;
savefig([num2str(m_dot,1) 'mdot_' num2str(Pc) 'Pc_'
 num2str(Rc*1000,1) 'Rc'])

% print off key attributes
disp(sprintf('\n %f lbs Thrust ', (F/1000)*224.8));
disp(sprintf('\n %f Pascals Chamber Pressure', Pc));
disp(sprintf('\n %f CC Length mm ', (Lc*1000)));
disp(sprintf('\n %f CC radius mm ', (Rc*1000)));
disp(sprintf('\n %f Convergent Section Length mm ', (conv_len*1000)));
disp(sprintf('\n %f Nozzle Length mm ', (Ln*10)));
disp(sprintf('\n %f Exit Nozzle Radius mm ', (Ae_radius*10)));
disp(sprintf('\n %f Throat Nozzle Radius mm ', (At_radius*10)));

end
```

```
 975.077009 lbs Thrust

 1378952.000000 Pascals Chamber Pressure

 265.232457 CC Length mm

 57.150000 CC radius mm

 34.361508 Convergent Section Length mm

 68.099261 Nozzle Length mm
```

```
45.639133 Exit Nozzle Radius mm

27.392019 Throat Nozzle Radius mm
```



Published with MATLAB® R2015a