```
clear
clc
close all

% User Defined Properties
a = arduino('COM5','Due','Libraries','rotaryEncoder')  % define the Arduino Communication↙
port
% a = arduino('/dev/ttyACM0','Due','Libraries','rotaryEncoder')    % for Linux

configurePin(a,'D2', 'DigitalOutput')   % ENA
configurePin(a,'D3', 'DigitalOutput')   % IN1
configurePin(a,'D4', 'DigitalOutput')   % IN2
configurePin(a,'D5', 'DigitalOutput')   % IN3
configurePin(a,'D6', 'DigitalOutput')   % IN4
configurePin(a,'D7', 'DigitalOutput')   % ENB
writeDigitalPin(a,'D2', 1);     % enable motor A
writeDigitalPin(a,'D3', 0);     % choose sense of rotation motor A
writeDigitalPin(a,'D4', 1);     % choose sense of rotation motor A
writeDigitalPin(a,'D5', 0);     % choose sense of rotation motor B
writeDigitalPin(a,'D6', 1);     % choose sense of rotation motor B
writeDigitalPin(a,'D7', 1);     % enable motor B

encoder = rotaryEncoder(a,'D8','D10',11)    % create encoder object 4x

time = 0;
data = 0;
count = 0;
rpm = 0;
v_rpm = 0;

ax1 = subplot(2,1,1); % top subplot
ax2 = subplot(2,1,2);

tic
while(1>0)
    pot = readVoltage(a,'A0');  % read analog voltage from potentiometer
    c = readVoltage(a,'A1') / 2; % current measurement(OP-Amp Gain = 2)
    dat = c *0.729; % torque computation (K_T = 0.729)
    writePWMVoltage(a,'D7',pot);    % change load motor speed with PWM
    count = count + 1;
    time(count) = toc;
    data(count) = dat;
    plot(ax1,time,data,'-r' );  % plot torque

    rpm = readSpeed(encoder);      % current angular velocity [rpm]
    v_rpm(count) = rpm / 40;       % velocity vector [reduction ratio 40:1]
    plot(ax2,time,v_rpm,'-r');     % plot angular velocity

    %Update the graph
    pause(.1);
end
```