```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ECE341_Junior Design

% Sensor Calibration with Matlab

% Wurscher, Westley ; Gonzalez, Yenifer R. ; Ma, Naixuan

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
%Connect to Arduino board
a = arduino('COM6','Uno');
%Setup photograph
figure
xlabel('Time(sec)');
yyaxis left
ylabel('Resistance(Ohms)');
yyaxis right
ylabel('Temperature(C)&Differnce(%)');
title('Resistance & Approximated Temperature vs Time');
ax = gca;
ax.FontSize = 12;
ax.YLim = [0 100];
yyaxis left
ax.YLim = [0 20000];
% Creating animated lines to add points to
% Resistance values
line1 = animatedline('Color', 'red', 'Linewidth', 2, 'DisplayName',
'PhotoResistance');
line2 = animatedline('Color', 'black', 'Linewidth', 2, 'DisplayName',
'ThermistorResistance');
yyaxis right
% Steinhart-Hart Model
line3 = animatedline('Color', 'blue', 'Linewidth', 2, 'DisplayName',
'SteinHartTemp','LineStyle','--');
% Linear Approximation
line4 = animatedline('Color', 'cyan', 'Linewidth', 2, 'DisplayName',
'LinearApprox');
% Running Average
line5 = animatedline('Color', 'magenta', 'Linewidth', 2, 'DisplayName',
'Average','LineStyle',':');
% Difference
line6 = animatedline('Color', 'yellow', 'Linewidth', 2, 'DisplayName',
'Difference');
% Show the intersection points of SteinHart temperature and LinearApprox
line7 = animatedline('Color', 'green', 'Linewidth', 2, 'DisplayName',
'Intersection','Marker','o','MarkerSize',10,'MarkerFaceColor','green','Ma
ximumNumPoints',1);
%legend('LDR Resistance','Thermistor Resistance','Thermistor
Temperature','Linear Approximation of Temp','Average
Temperature','Difference Between Steinhart & Linear','Location','east')
% Setup an array to get some data points to calculate average temperature
i = 0;
array = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
```

```
% Gets current time
startTime = datetime('now');
% Announce the variables
photocellVoltage = 0;
thermistorVoltage = 0;
thermistorResistance = 0;
% The coefficients we will use for the Steinhart equation to calculate
temperature
A = 0.00128663;
B = 0.000217271;
C = 8.47847e-8;
while 1
      %Read the thermistor voltage from Arduino
      thermistorVoltage = readVoltage(a,'A0');
      %The resistance of the thermistor
      thermistorResistance = 10000 * ((5 / thermistorVoltage) - 1.0);
      %The temperature of thermistor
      thermistorTemp = (1.0 / (A + (B*log(thermistorResistance)) +
(C*log(thermistorResistance)*log(thermistorResistance)*log(thermistorResi
stance)))) ;
      %This converts the temperature to celcius
      thermistorTemp = thermistorTemp - 273.15;
      %Change the temperature to integer value
      thermistorTemp1 = int16(thermistorTemp)
      %Read the photocell voltage from Arduino
      photocellVoltage = readVoltage(a,'A1');
      %The resistance of the photocell
      photoResistance = 1000* (5.0 / photocellVoltage - 1.0);
      %Get the data into array
      i= i+1;
      if (i > 20)
      i = 1;
      end
      array(i) = thermistorTemp;
      average
=(array(1)+array(2)+array(3)+array(4)+array(5)+array(6)+array(7)+array(8)
+array(9)+array(10)+array(11)+array(12)+array(13)+array(14)+array(15)+arr
ay(16)+array(17)+array(18)+array(19)+array(20))/20;
      %Linearapprox of thermistor temperature
      linearApprox = (23415 - thermistorResistance)/536.6;
      %Change the temperature to integer value
      linearApprox1 = int16(linearApprox)
      %The percent difference of SteinHart measured temperature and
linearapprox temperature
      percentDifference = abs( (linearApprox - thermistorTemp) /
thermistorTemp ) * 100;
      %Add real time data to lines
      t = datetime('now') - startTime;
      addpoints(line1, datenum(t), photoResistance);
      addpoints(line2, datenum(t), thermistorResistance);
      addpoints(line3, datenum(t), thermistorTemp);
      addpoints(line4, datenum(t), linearApprox);
      addpoints(line5, datenum(t), average);
      addpoints(line6, datenum(t), percentDifference);
```

```
        ax = gca;
        ax.XLim = datenum([t-seconds(10) t]);
        datetick('x','keeplimits');
        %Compare SteinHart temperature and linearapprox temperature
        if(thermistorTemp1 == linearApprox1)
        addpoints(line7, datenum(t), thermistorTemp);
        end
        grid on;
        drawnow;
end
```