Drexel University

Memo #5

To: ECE 303

From: David Schmidt

CC: Other recipients

Arduino Mega 2650 Lab 5 GUI for photo resistant diode

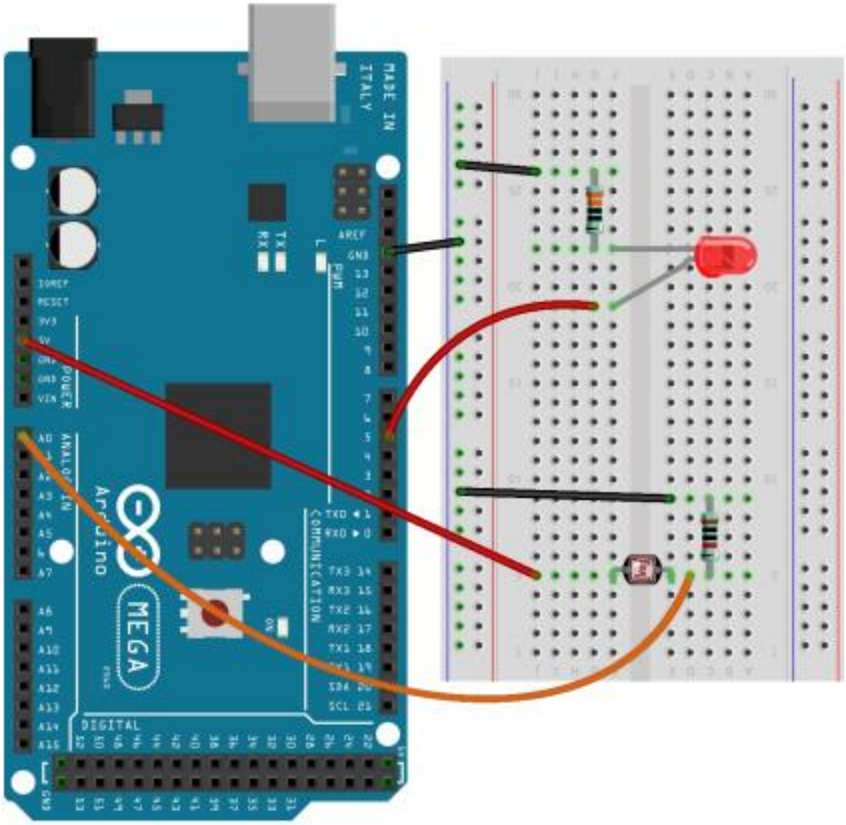
Task: To create a Photo resistant LED program that communicates serial data between MATLAB and Arduino software using a 9600 Baud rate. Application needed to have start button and 3 different continuously updating graphs for Photocell resistance, Current and Voltage over a displayed Duty Cycle percentage which increased .01% per cycle

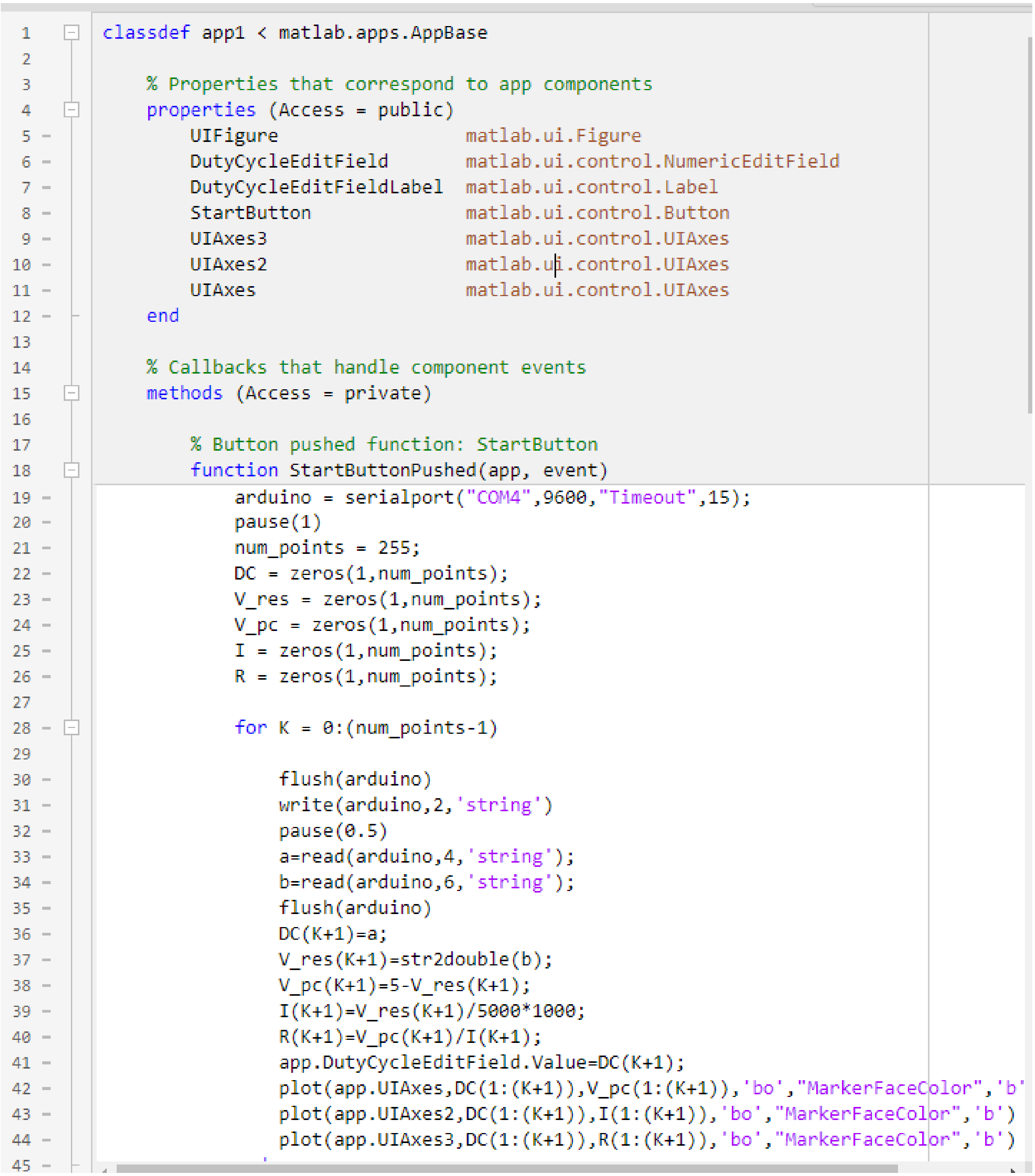
Equipment: Mega Arduino 2650 starter kit

1-1k and 1-10k ohm resistor
Jumper wires
1 Photo cell
1 5v LED
Bread Board

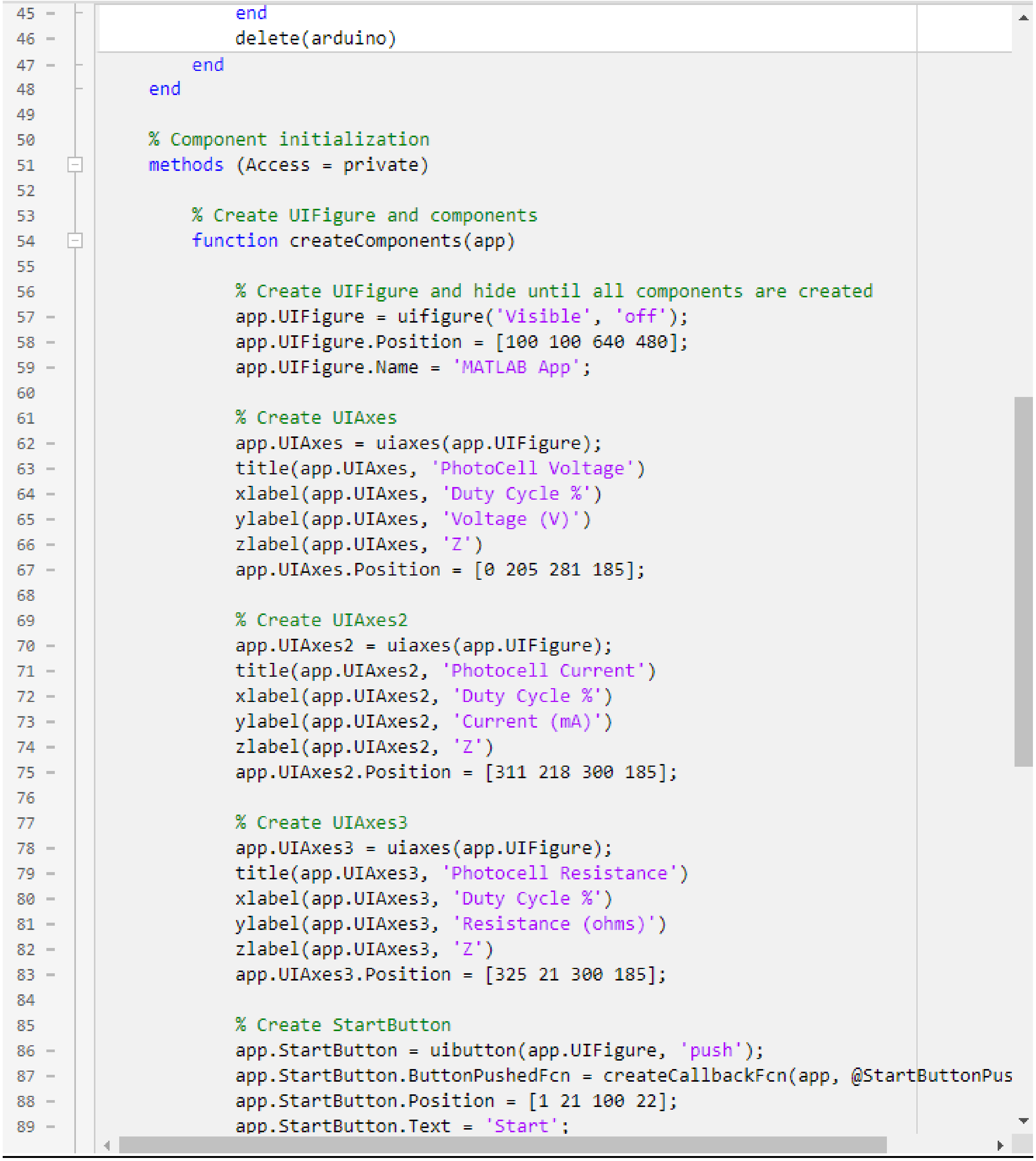
CODE: Code is on the following pages and the Arduino sketch will be attached with the submission

Saitama Japan **9/20/2021**

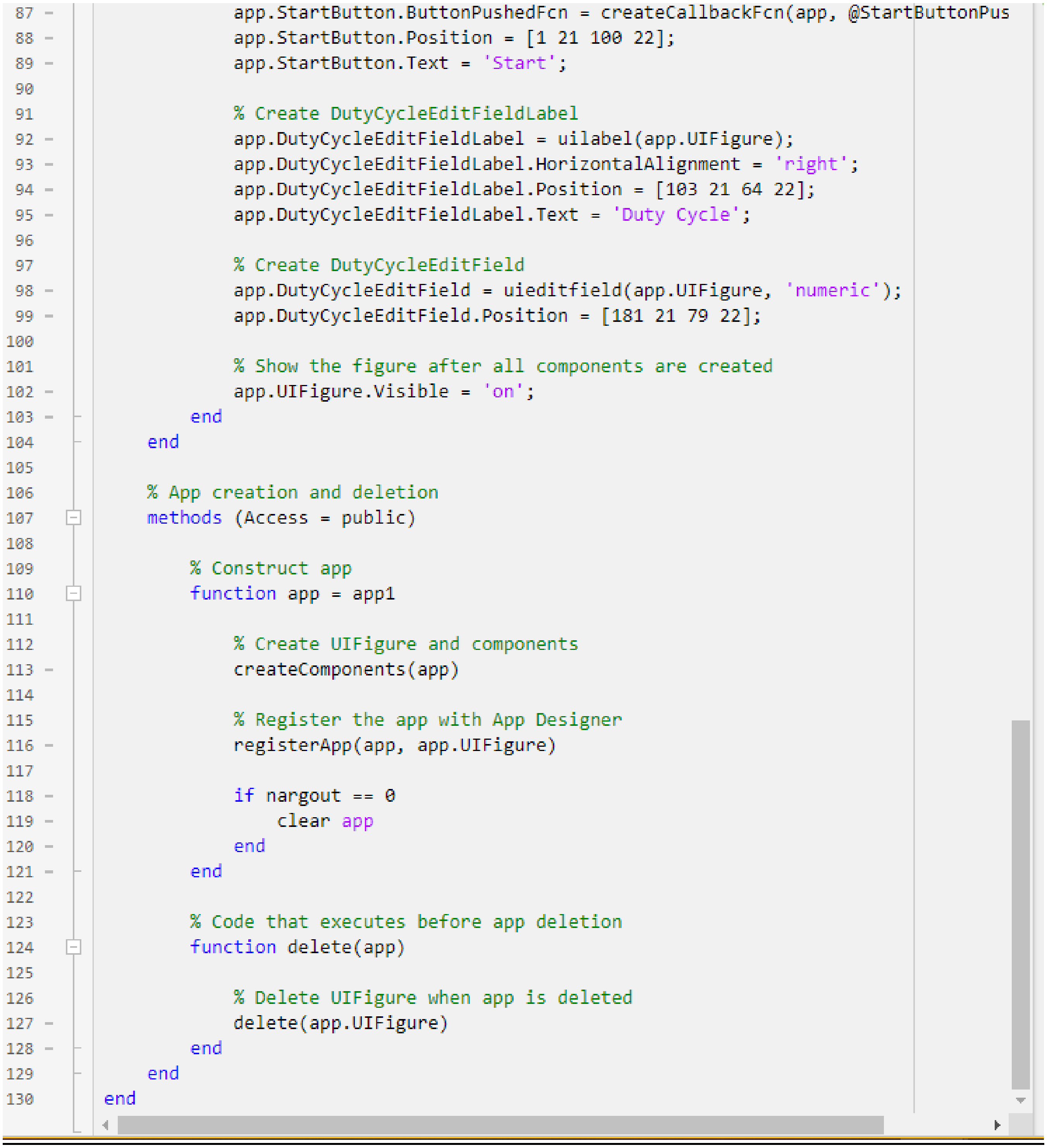

MADE IN ITALY
TX
RX
L
PWM
AREF
GND
13
12
11
10
9
8
7
6
5
4
3
2
TXD ◄ 1
RXD ► 0
TX3 14
RX3 15
TX2 16
RX2 17
TX1 18
RX1 19
SDA 20
SCL 21
COMMUNICATION
IOREF
RESET
3V3
5V
GND
GND
VIN
POWER
A0
A1
A2
A3
A4
A5
A6
A7
ANALOG IN
A8
A9
A10
A11
A12
A13
A14
A15
Arduino
MEGA
ON
DIGITAL
GND



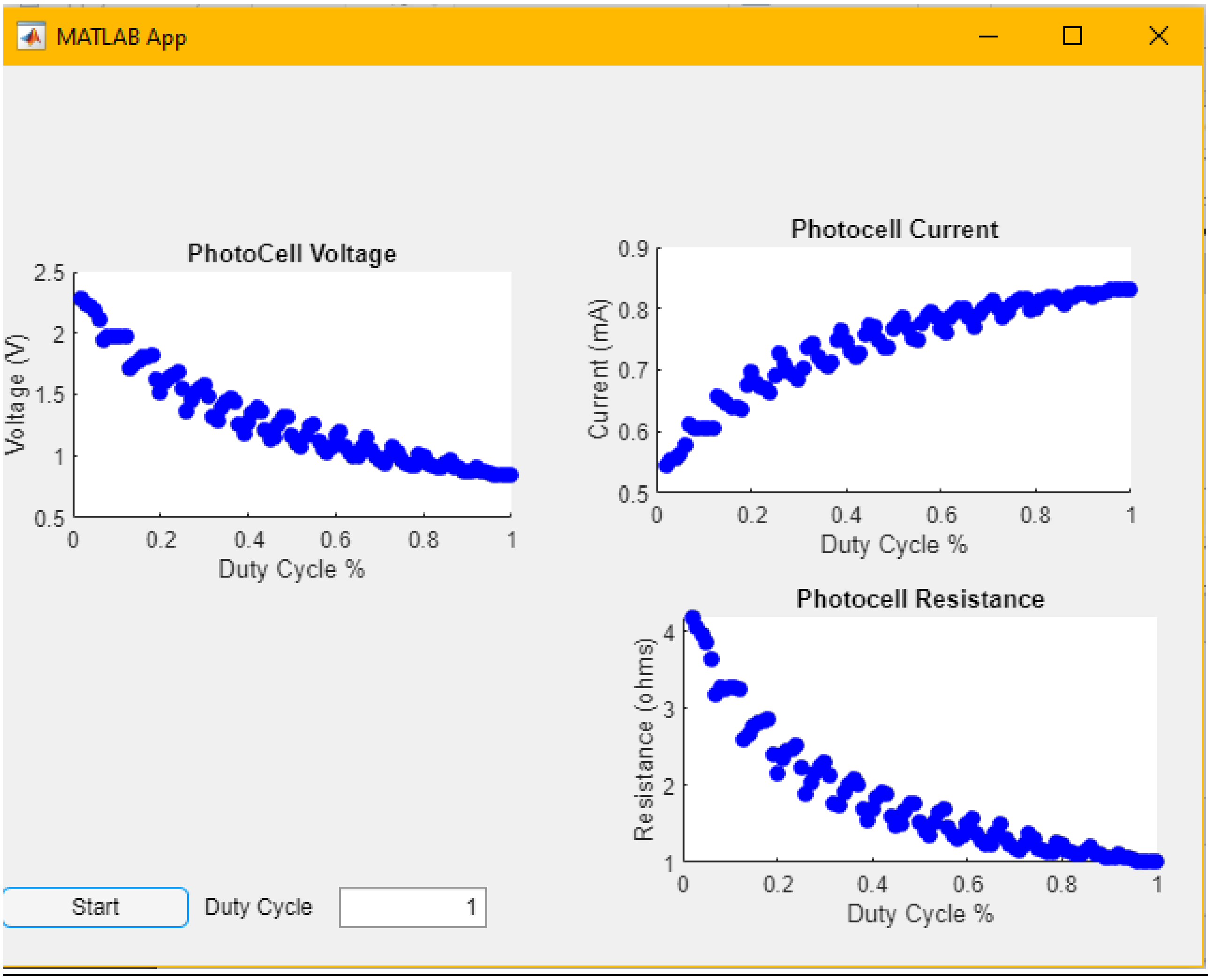
```
classdef app1 < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                 matlab.ui.Figure
        DutyCycleEditField       matlab.ui.control.NumericEditField
        DutyCycleEditFieldLabel  matlab.ui.control.Label
        StartButton              matlab.ui.control.Button
        UIAxes3                  matlab.ui.control.UIAxes
        UIAxes2                  matlab.ui.control.UIAxes
        UIAxes                   matlab.ui.control.UIAxes
    end

    % Callbacks that handle component events
    methods (Access = private)

        % Button pushed function: StartButton
        function StartButtonPushed(app, event)
            arduino = serialport("COM4",9600,"Timeout",15);
            pause(1)
            num_points = 255;
            DC = zeros(1,num_points);
            V_res = zeros(1,num_points);
            V_pc = zeros(1,num_points);
            I = zeros(1,num_points);
            R = zeros(1,num_points);

            for K = 0:(num_points-1)

                flush(arduino)
                write(arduino,2,'string')
                pause(0.5)
                a=read(arduino,4,'string');
                b=read(arduino,6,'string');
                flush(arduino)
                DC(K+1)=a;
                V_res(K+1)=str2double(b);
                V_pc(K+1)=5-V_res(K+1);
                I(K+1)=V_res(K+1)/5000*1000;
                R(K+1)=V_pc(K+1)/I(K+1);
                app.DutyCycleEditField.Value=DC(K+1);
                plot(app.UIAxes,DC(1:(K+1)),V_pc(1:(K+1)),'bo',"MarkerFaceColor",'b'
                plot(app.UIAxes2,DC(1:(K+1)),I(1:(K+1)),'bo',"MarkerFaceColor",'b')
                plot(app.UIAxes3,DC(1:(K+1)),R(1:(K+1)),'bo',"MarkerFaceColor",'b')
```



```
            end
            delete(arduino)
        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.Position = [100 100 640 480];
            app.UIFigure.Name = 'MATLAB App';

            % Create UIAxes
            app.UIAxes = uiaxes(app.UIFigure);
            title(app.UIAxes, 'PhotoCell Voltage')
            xlabel(app.UIAxes, 'Duty Cycle %')
            ylabel(app.UIAxes, 'Voltage (V)')
            zlabel(app.UIAxes, 'Z')
            app.UIAxes.Position = [0 205 281 185];

            % Create UIAxes2
            app.UIAxes2 = uiaxes(app.UIFigure);
            title(app.UIAxes2, 'Photocell Current')
            xlabel(app.UIAxes2, 'Duty Cycle %')
            ylabel(app.UIAxes2, 'Current (mA)')
            zlabel(app.UIAxes2, 'Z')
            app.UIAxes2.Position = [311 218 300 185];

            % Create UIAxes3
            app.UIAxes3 = uiaxes(app.UIFigure);
            title(app.UIAxes3, 'Photocell Resistance')
            xlabel(app.UIAxes3, 'Duty Cycle %')
            ylabel(app.UIAxes3, 'Resistance (ohms)')
            zlabel(app.UIAxes3, 'Z')
            app.UIAxes3.Position = [325 21 300 185];

            % Create StartButton
            app.StartButton = uibutton(app.UIFigure, 'push');
            app.StartButton.ButtonPushedFcn = createCallbackFcn(app, @StartButtonPus
            app.StartButton.Position = [1 21 100 22];
            app.StartButton.Text = 'Start';
```

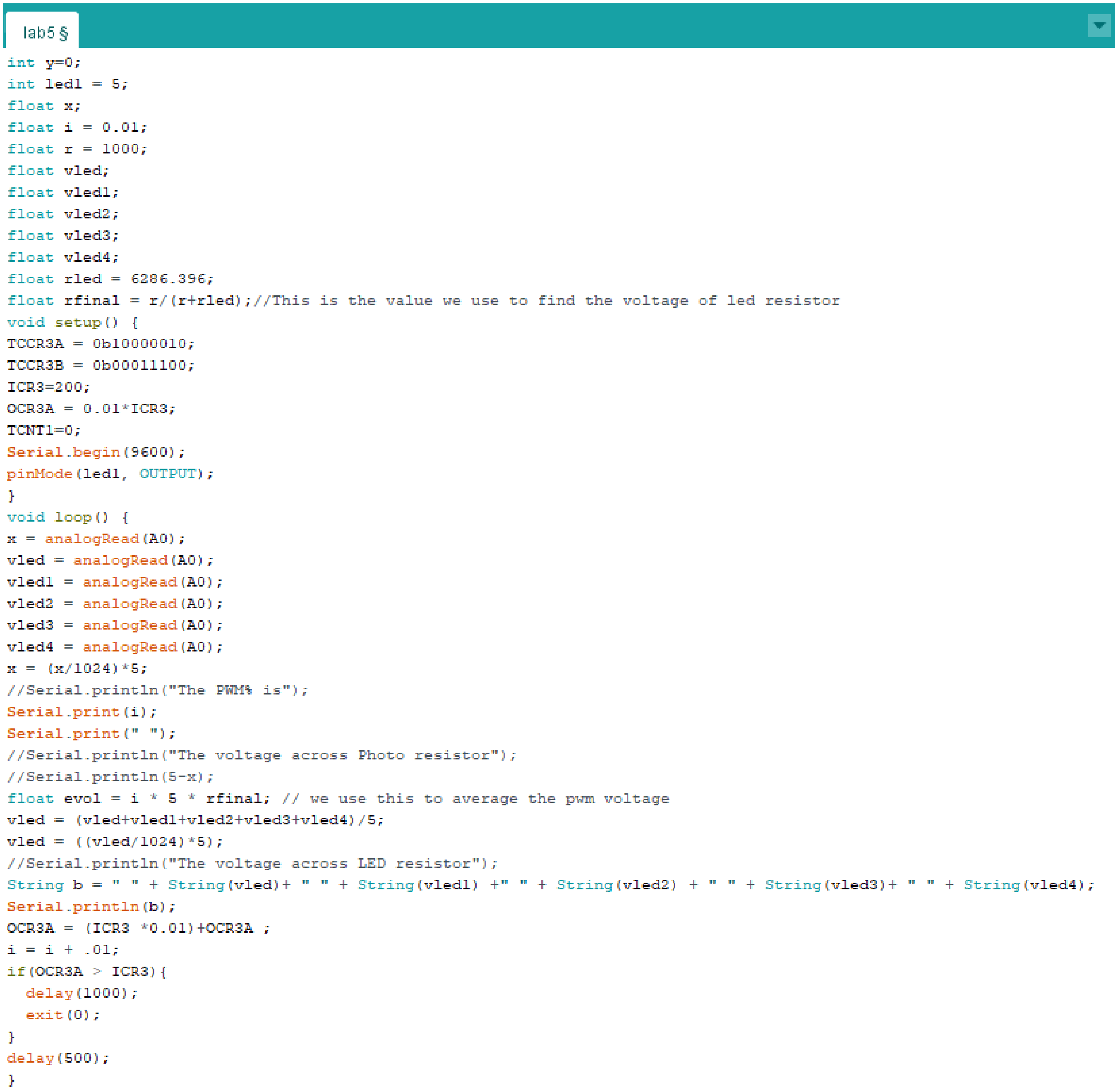


```
            app.StartButton.ButtonPushedFcn = createCallbackFcn(app, @StartButtonPus
            app.StartButton.Position = [1 21 100 22];
            app.StartButton.Text = 'Start';

            % Create DutyCycleEditFieldLabel
            app.DutyCycleEditFieldLabel = uilabel(app.UIFigure);
            app.DutyCycleEditFieldLabel.HorizontalAlignment = 'right';
            app.DutyCycleEditFieldLabel.Position = [103 21 64 22];
            app.DutyCycleEditFieldLabel.Text = 'Duty Cycle';

            % Create DutyCycleEditField
            app.DutyCycleEditField = uieditfield(app.UIFigure, 'numeric');
            app.DutyCycleEditField.Position = [181 21 79 22];

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = app1

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```



Duty Cycle 1

Arduino code



```
int y=0;
int led1 = 5;
float x;
float i = 0.01;
float r = 1000;
float vled;
float vled1;
float vled2;
float vled3;
float vled4;
float rled = 6286.396;
float rfinal = r/(r+rled);//This is the value we use to find the voltage of led resistor
void setup() {
TCCR3A = 0b10000010;
TCCR3B = 0b00011100;
ICR3=200;
OCR3A = 0.01*ICR3;
TCNT1=0;
Serial.begin(9600);
pinMode(led1, OUTPUT);
}
void loop() {
x = analogRead(A0);
vled = analogRead(A0);
vled1 = analogRead(A0);
vled2 = analogRead(A0);
vled3 = analogRead(A0);
vled4 = analogRead(A0);
x = (x/1024)*5;
//Serial.println("The PWM% is");
Serial.print(i);
Serial.print(" ");
//Serial.println("The voltage across Photo resistor");
//Serial.println(5-x);
float evol = i * 5 * rfinal; // we use this to average the pwm voltage
vled = (vled+vled1+vled2+vled3+vled4)/5;
vled = ((vled/1024)*5);
//Serial.println("The voltage across LED resistor");
String b = " " + String(vled)+ " " + String(vled1) +" " + String(vled2) + " " + String(vled3)+ " " + String(vled4);
Serial.println(b);
OCR3A = (ICR3 *0.01)+OCR3A ;
i = i + .01;
if(OCR3A > ICR3){
  delay(1000);
  exit(0);
}
delay(500);
}
```