# Lab and Team Project Development for Engineering Problem Solving using MATLAB, with Emphasis on Solar Power and Engineering for Sustainability

Paper ID# 6114

**Stanley Hsu**, Rajeevan Amirtharajah, André Knoesen

Electrical & Computer Engineering Department,
University of California, Davis

# Outline

- **Introduction**
- Course Flow and Lab Topics
- Arduino Solar Module
- Individual and Team Project
- Assessment Results
- Summary

# Introduction

- Goal: Update Existing Curriculum
  - ***Sustainability*** focused
  - ***Hands-on*** experience
  - ***Project-based*** learning

**UC Davis Robert Mondavi Institute Teaching Winery**

- Course Structure
  - Size: Fall, 175+; Winter, 250+; Spring, 150+
  - Demographics: multiple engineering disciplines, mostly freshmen and sophomores but all classes represented
  - Objective: engineering problem solving using programming

- Course Components
  - Lecture: twice a week, 90 minutes each
  - Lab: weekly timed programming exercises, 50 minutes
  - Projects: one individual and one team
  - Homework: weekly

MathWorks



UCDAVIS

# Outline

- Introduction
- **Course Flow and Lab Topics**
- Arduino Solar Module
- Individual and Team Project
- Assessment Results
- Summary

# Course Flow and Lab Topics

Quarter Starts

**Fundamentals**

- 1D and 2D Array
- Flow Control
- Looping Constructs
- 2D and 3D Plotting
- Text and File I/O
- Interpolation and Numerical Integration

**Advanced**

- Monte Carlo Technique
- String Manipulation
- Graphical User Interface
- Object-Oriented Programming
- Regular Expressions

Quarter Ends

| **Lecture Topic** | **Sustainability-related Lab Topics** |
|---|---|
| 1D arrays | Solve for average temperature given degree-day data, for multiple cities. |
| 2D arrays | Modeling a solar panel using rows and columns of photodiodes. |
| Flow control, loops, logical operators | Total cost of a solar panel array including volume discount. Solar energy investment and analysis. |
| Custom functions | Write a function that computes the output power of an N by N solar panel array. |
| Numerical computing | Newton's method for solving a nonlinear equation such as the I-V equation of a photodiode. |
| Curve fitting | Compare performance between fixed and tracking solar panel by curve fitting I-V data. |
| GUI and OOP | Gather luminance data using the Arduino Solar Module. |
| File I/O | Importing weather database from Excel. |

www.ece.ucdavis.edu/**sustainableENG6**

# Weekly Lab Exercises

- Problem 0: Challenge Problem
  - Single part problem based on previous homework and lab
  - Minimal instruction given
  - Expected completion time: 10-20 minutes

- Problem 1: Skill Building Problem
  - Multi-part problem based on newly introduced concept
  - Step by step instructions given
  - Expected completion time: 30-40 minutes

# Outline

- Introduction
- Course Flow and Lab Topics
- **Arduino Solar Module**
- Individual and Team Project
- Assessment Results
- Summary

# Arduino Solar Module

- Hands-on Experience
  - Improve student engagement in engineering
  - Increase student interest in electronics
- Hardware Module
  - Based on Arduino UNO
  - Solar cell to measure light
  - Programmable via MATLAB

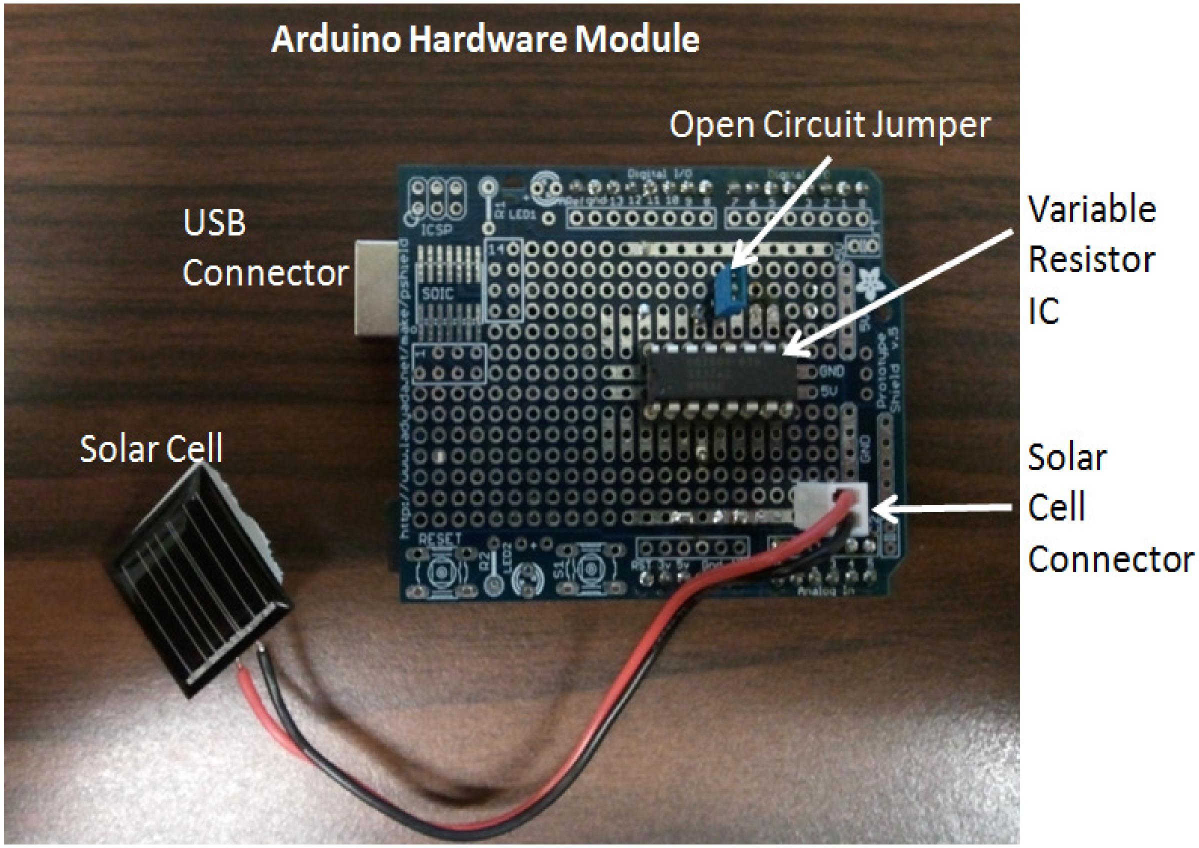

www.ece.ucdavis.edu/**sustainableENG6**



MathWorks

UCDAVIS

# Arduino Solar Module BOM

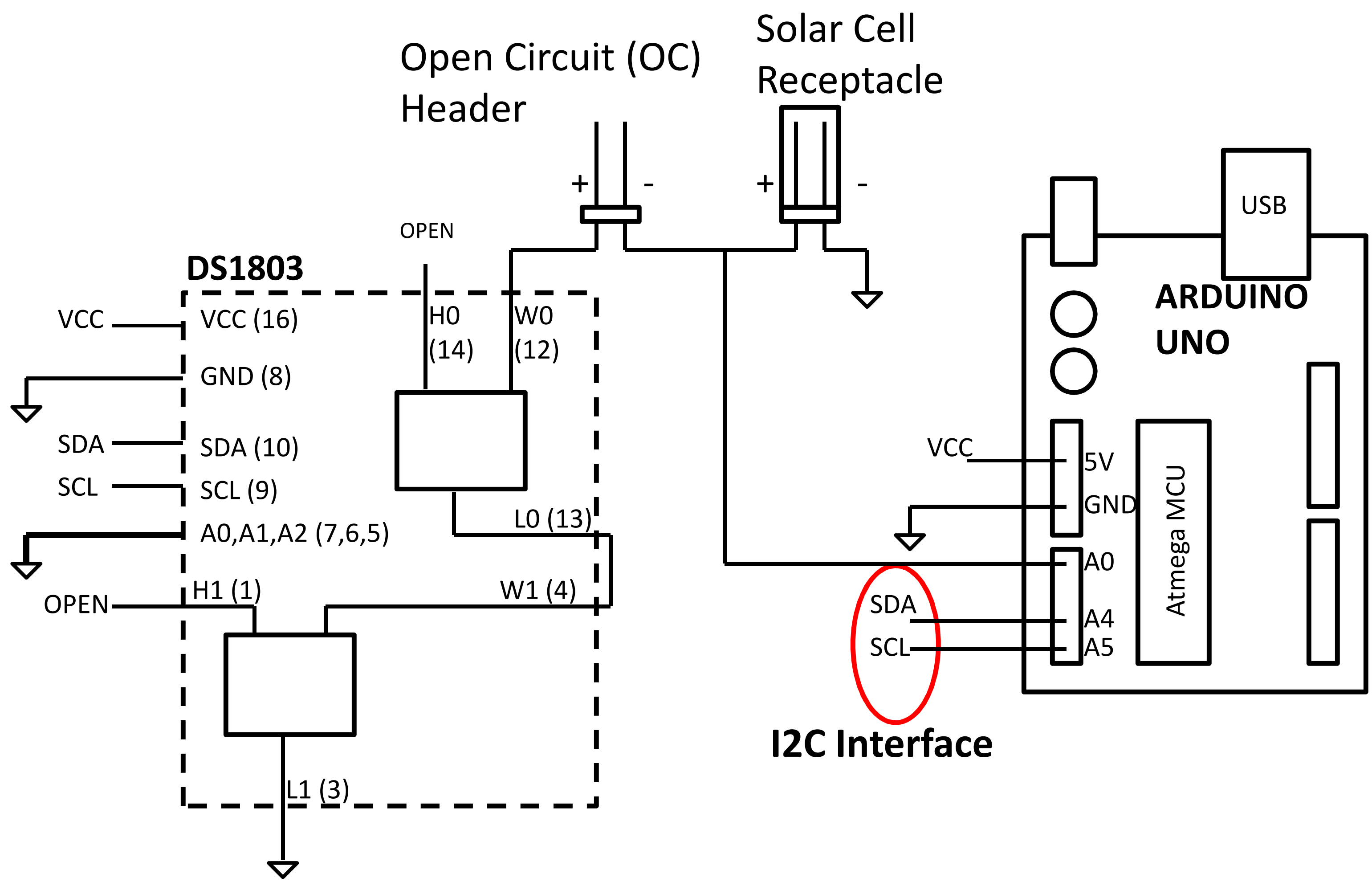



Bill of Materials

| | Unit Cost |
|---|---|
| Arduino UNO | $26.96 |
| USB Cable | $1.67 |
| DS1803 Potentiometer | $3.22 |
| Solar Cell | $3.15 |
| Connector Receptacle, Header, and Crimp | $0.15 + $0.09 + $0.06 |
| Arduino Proto Shield | $11.00 |
| Jumper | $0.08 |
| | |
| ***Cost of One Unit*** | ***$46.38*** |

www.ece.ucdavis.edu/**sustainableENG6**

# Arduino Solar Module Schematic




www.ece.ucdavis.edu/**sustainableENG6**

# Arduino Solar Module Software

MATLAB script
written by students

arduino
object
(provided)

**arduino**

Property:
...
...
Method:
...
...

USB
port

"adiosrv.ino"
Pre-programmed C code running on the UNO

```
void setup() {
// setup pin mode
// setup wire interface
}
void loop() {
  ...
  cmd = serial.read();
  switch(cmd) {
    ...
    case 30: Read analog input pin
    ...
    case 50: Set potentiometer
    ...
    }
}
```

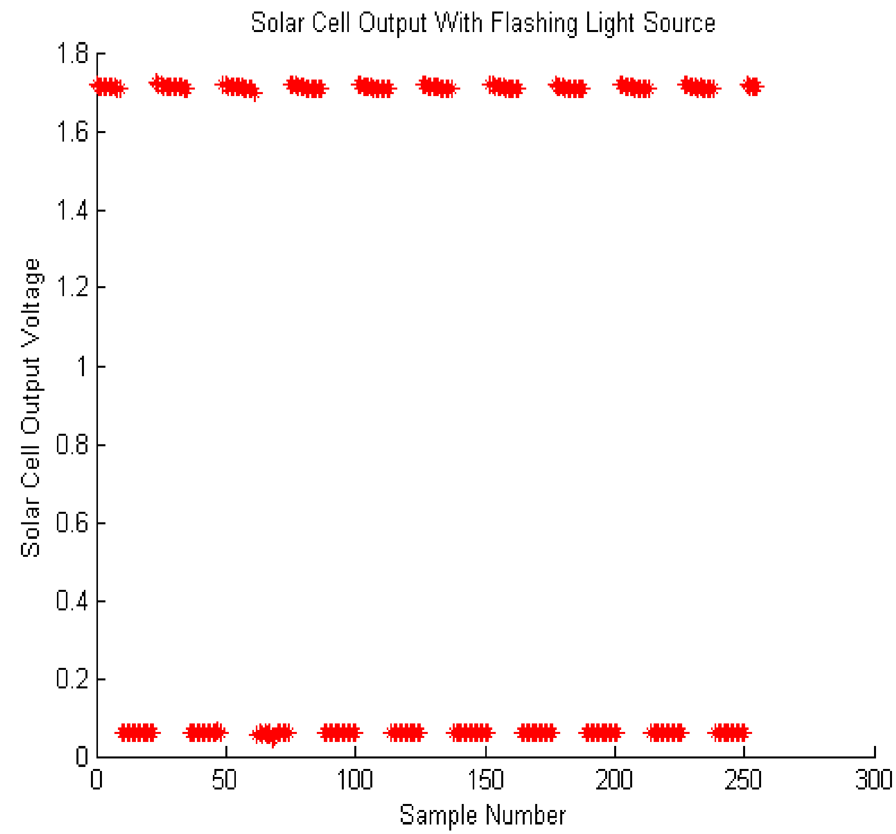
**I2C**
(digital potentiometer)

**analog input pin**
(solar cell)

www.ece.ucdavis.edu/**sustainableENG6**

# Arduino Solar Module Output

- IV Curve
- Flashing Lights


Lab given after fundamental topics

# Outline

- Introduction
- Course Flow and Lab Topics
- Arduino Solar Module
- **Individual and Team Project**
- Course Assessment
- Summary

# Individual Term Project

- Time: 2 weeks
- Topic: Analysis of weather data of 53 cities across 12 months
  - Data given
    - Precipitation
    - Solar radiation
    - Wind speed
    - Land area
    - Population
  - Compute solar and wind power available
  - Estimate available rainwater for collection
  - Comparison between types of renewable energy

# Final Team Project

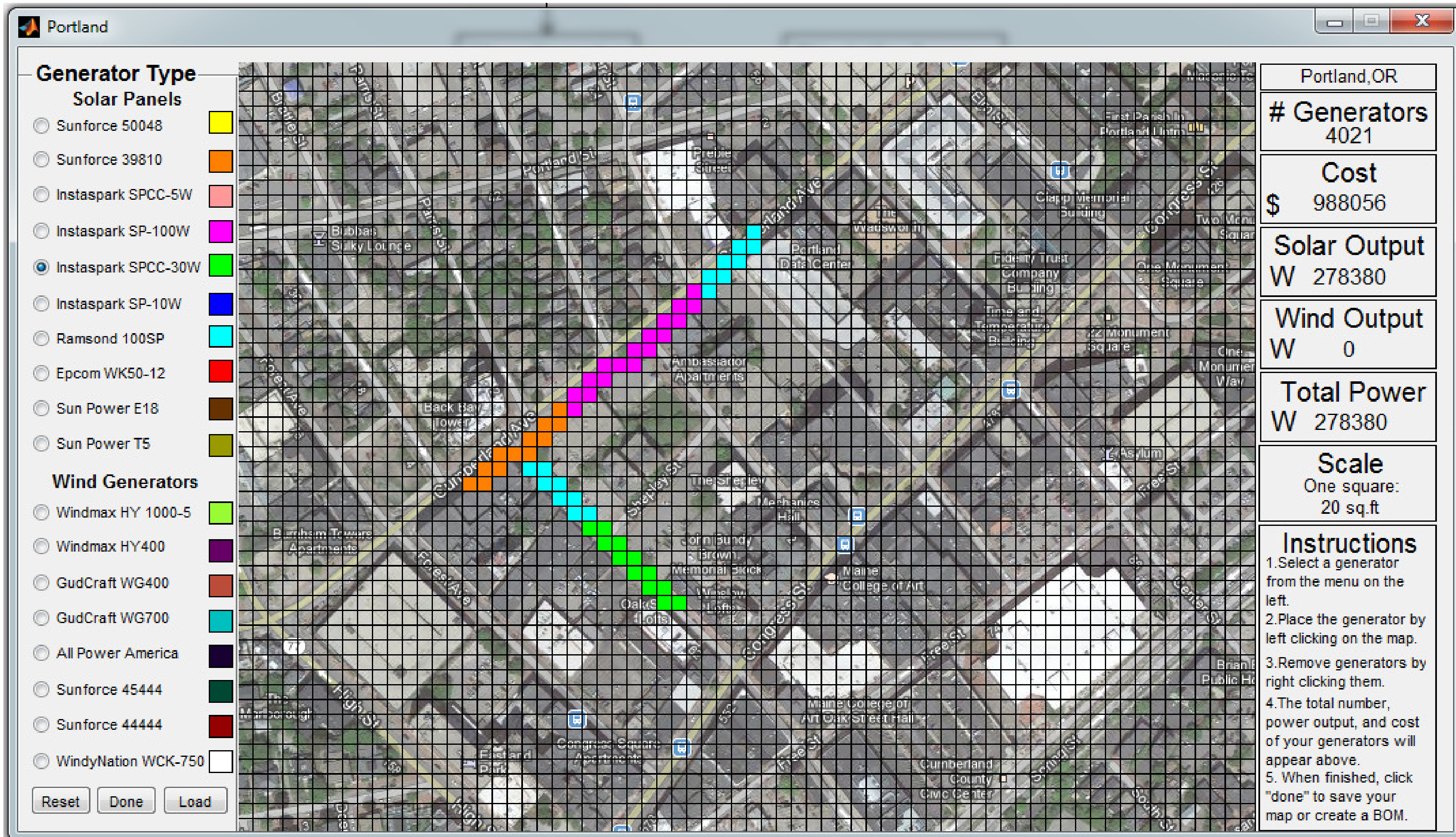
- Time: 4 weeks
- Team of 3 students
- Lecture on design cycle, project planning, team organization
- Teams choose from multiple project options
- ***Design*** project
  - GUI flow and design
  - Data organization and structure
- Submit:
  - Preliminary report, final report, code, video presentation

MathWorks

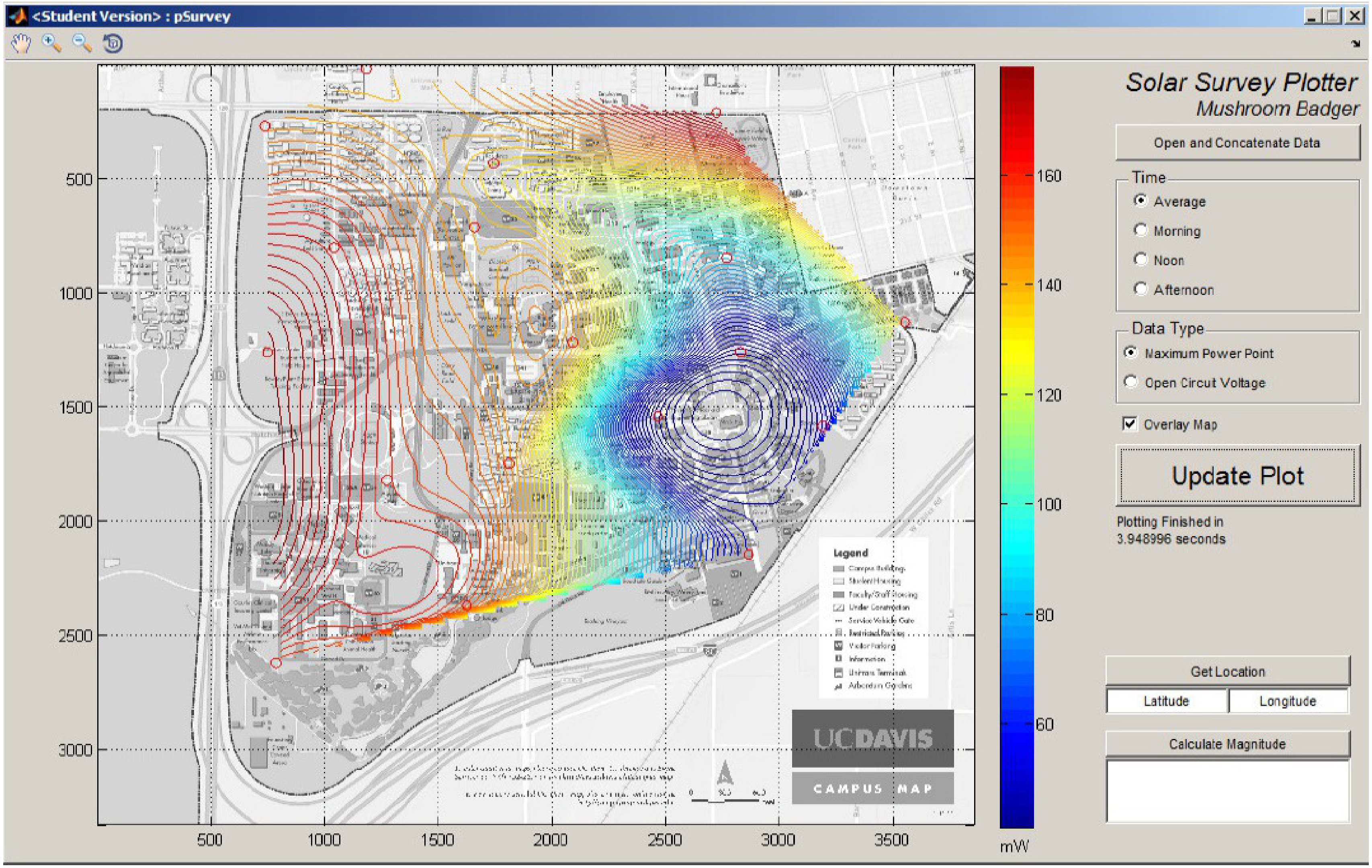


UCDAVIS

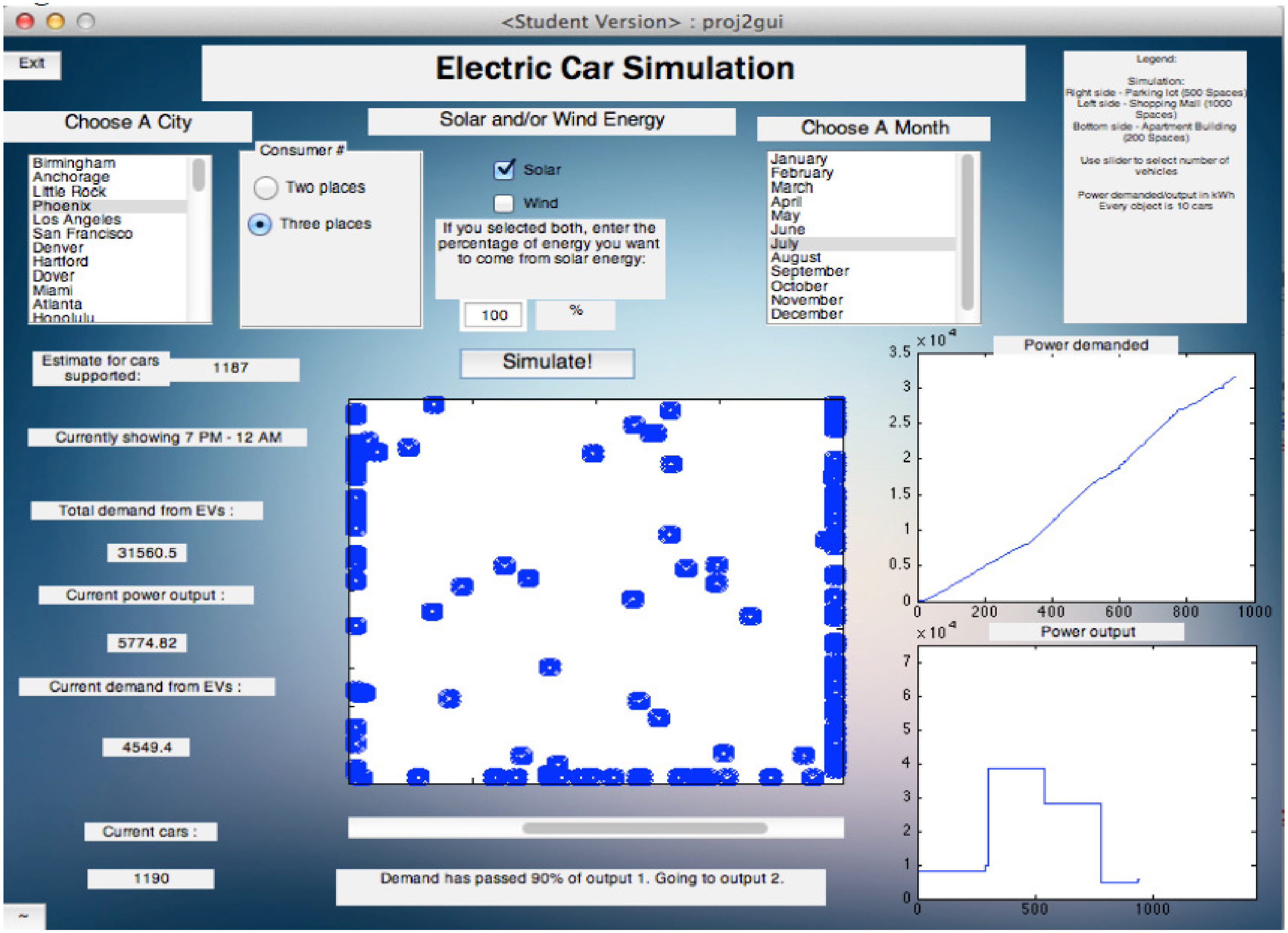
# Renewable Energy Planning and Development Tool



# Solar Survey



# Electric Vehicle Transportation Simulator






MathWorks

UCDAVIS

# Outline

- Introduction
- Course Flow and Lab Topics
- Arduino Solar Module
- Individual and Team Project
- **Assessment Results**
- Summary

# Assessment Results

- Questionnaire implemented using Google Form
- 403 students from four quarters
- 105 responses were received

# Assessment Results

| | **1** - *Much less effective* | **2** – *Less effective* | **3** - *Similar* | **4** – *More effective* | **5** - *Much more effective* | | |
|---|---|---|---|---|---|---|---|
| **Development of problem solving skills.** | 8 | 9 | 31 | **34** | 23 | | **Mean: 3.52** Std. Dev.: 1.15 |
| **Ability to design and implement an algorithm to solve a given problem.** | 12 | 11 | 27 | **35** | 20 | | **Mean: 3.38** Std. Dev.: 1.23 |
| **Ability to define a problem and specification.** | 9 | 11 | **36** | 29 | 20 | | **Mean: 3.38** Std. Dev.: 1.16 |
| **Engagement in course materials.** | 11 | 18 | 20 | **46** | 10 | | **Mean: 3.24** Std. Dev.: 1.16 |
| **Hands-on experience with electronic hardware.** | 9 | 13 | 23 | 24 | **36** | | **Mean: 3.62** Std. Dev.: 1.3 |
| **Comfortable using MATLAB to solve problems.** | 10 | 10 | 28 | **29** | 28 | | **Mean: 3.52** Std. Dev.: 1.25 |

MathWorks



UCDAVIS

# Outline

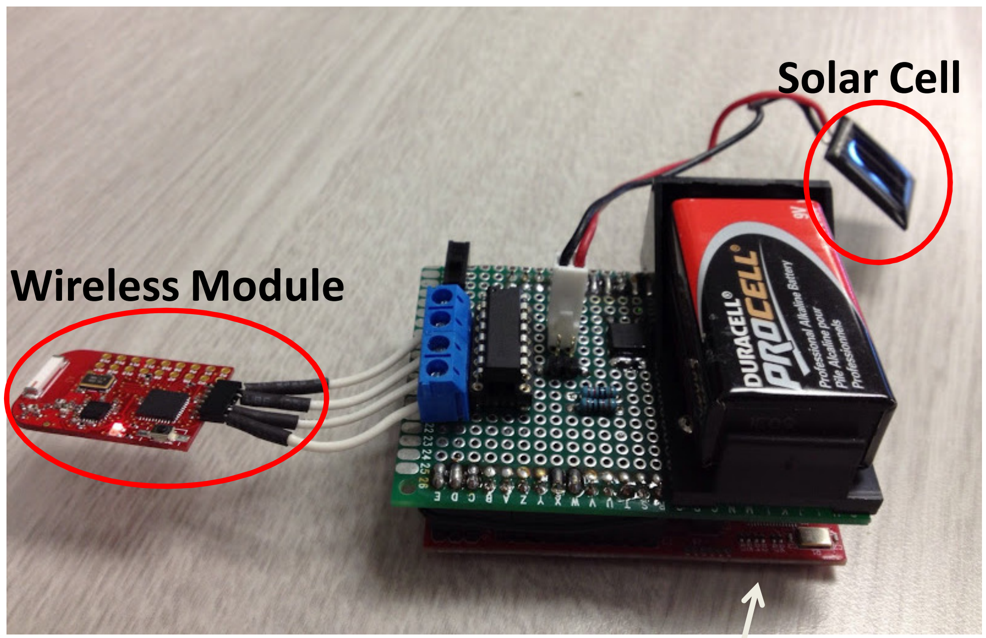
- Introduction
- Course Flow and Lab Topics
- Arduino Solar Module
- Individual and Team Project
- Assessment Results
- **Summary**

# Summary

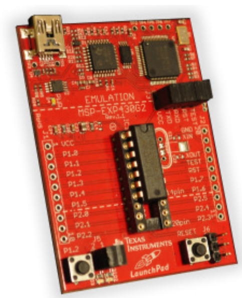
- Revised curriculum for problem solving class using programming
  - **Sustainability** theme incorporated in labs and projects
  - **Hands-on** experience enabled by low-cost custom Arduino Solar Module
  - **Project-based** learning through individual and team projects

# Future Work

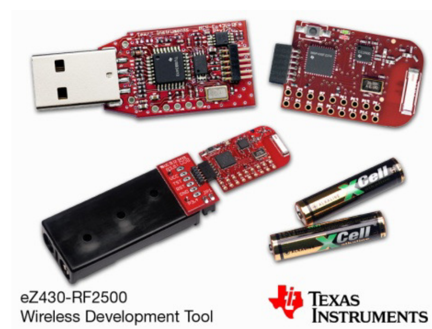
- Lower cost and increase functionality
  - MSP430 microcontroller and wireless module




**TI MSP430 Launch-pad**



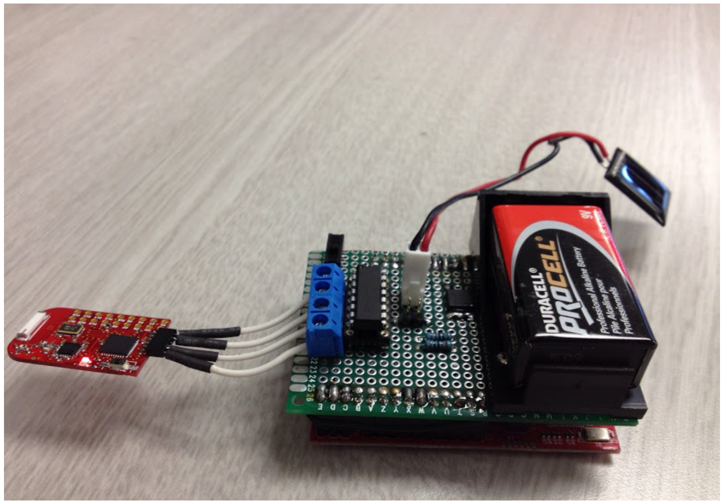
TI MSP430 Launchpad

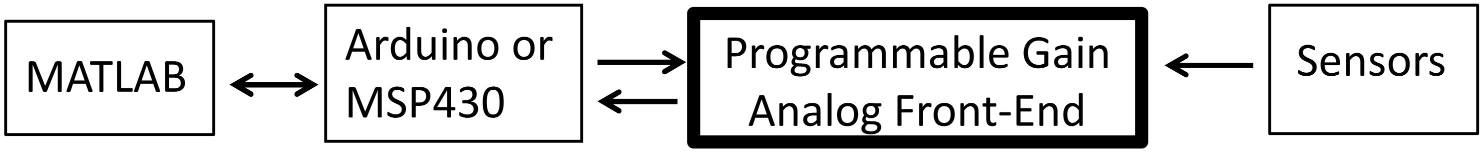



eZ430-RF2500 Wireless Dev. Kit

# Future Work

- Increase MATLAB programmability and functionality of module
- Increase hands-on lab exercises
- Multi-sensor interface:






MathWorks

UCDAVIS

# Acknowledgements


- Mathworks for funding and guidance.
  - Dr. Giampiero Campa
- University of California, Davis for implementation and development.
  - Mikaela Huntzinger and Karen Dunn-Haley (assessment)
  - Lance Halsted (Arduino development)
  - Shane Austin (course work and Arduino dev.)
  - Ryan Wilkerson and Alex Chan-Kai (TI wireless module)
  - Alexander McCourt and other ENG6 Teaching Assistants (implementation)
  - Chik Brenneman (RMI teaching winery tour)
  - Prof. Roger Boulton
- Texas Instruments (MSP430 dev. kit donation)

# Back up slides

# Course Flow and Lab Topics

- Challenge Problem

The average cost of solar energy technology steadily decreases at a rate of 6% every 18 months, and the average cost of coal-based electricity steadily increases at a rate of 3% every 18 months. How long (in months) until the average cost of the two technologies becomes equal? The current average cost of solar energy to be $156.9 per MWh, and the average cost of coal-based electricity to be $99.6 per MWh.

- Solution

```
solar = 156.9;
coal = 99.6;
while solar > coal
    solar = solar - solar*0.06;
    coal = coal + coal*0.03;
    i = i+1;
end
fprintf('In %d months, solar energy becomes cheaper than coal-based electricity.\n', i*18)
fprintf('Solar cost: $%5.2f;   Coal-based cost: $%5.2f;\n\n', solar, coal);
```

# Course Flow and Lab Topics

- Skill Building Problem

**Task 1:** Write a (stand-alone, not anonymous) function that return the output voltage and current of a solar panel given the number of rows and columns of photodiodes in the solar panel. The equation that gives the output voltage and current of a solar panel is below.

The function header: [Vout Iout] = solar pane(Nrow, Ncol)
Output Voltage (Volts), Vout=Number of rows*0.6 Vols
Output Current (mA), Iout=Number of columns*50 mA
(mA = milli-Amperes = 10^-3 Amperes)

**Task 2**: (Complete this task in the same script for Problem 1)
The maximum output power of the solar panel is 90% of the product of the output voltage and output current. Using the function you wrote in Task 1, write a script that calls this function to calculate the maximum output power for square solar panels with sides ranging from 10 photodiodes to 100 photodiodes. In a square solar panel, the number of rows is equal to the number of columns.

**Task 3**: Write code (in the same script as Task 2) to plot the maximum output power for square solar panels versus the total number of photodiodes (for Nrow = Ncol = 10 to 100) in the solar panel. (x-axis is the number of photodiodes and y-axis is the maximum output power) Do not modify the function 'solar panel()'.