# Laboratory 2: Plant Communicator

### Required Equipment

### TMP36 temperature sensor

### Phototransistor

### DIY moisture sensor

### Breadboard

### Required Programs

* Zapier
* ThingSpeak API
* Arduino Desktop Application

### References and Data Sheets

### <https://create.arduino.cc/projecthub/arduino/plant-communicator-7ea06f> (Project Manual)

### <https://store.arduino.cc/usa/arduino-mkr1000-with-headers-mounted> (Arduino Datasheet)

### Objectives

In this laboratory exercise, you will use the Arduino Lab kit to investigate the use of sensors to monitor temperature, light, and moisture in an environment. Additionally, you will be managing real time clocks and alarms.

### Introduction

### When it comes to growing plants, it is understood that in order to survive one must maintain a livable temperature along with adequate amounts of sunlight and water. Rather than monitor these conditions in a traditional manner, we will be using an Arduino and multiple sensors/techniques to create a system for recording and analyzing the environmental conditions remotely. Additionally, these records will be sent via email to the user, so proper action can be taken to adjust levels if need be.

### At the very base of this lab is an Arduino, which is in essence a microcontroller. This small board can be used to input signals, run code, and output signals accordingly. In today’s lab we will be using the Arduino to process electrical signals coming in from a variety of sensors to create an ideal environment for our “plant”. Below is a diagram of the Arduino MKR1000 WIFI

### model we will be using throughout the semester.

### 

### Additionally, the data sheet can be found at <https://store.arduino.cc/usa/arduino-mkr1000-with-headers-mounted>.

### In order to transmit the data that is collected, we will be utilizing APIs, short for Application Programming Interface. In a nutshell, APIs basically allow us to “talk” to outside applications and in the context of this lab, we will be using them to send update emails on our collected data. Zapier will act as an intermediate software for this process, allowing us to use more complex APIs that the Arduino would normally not include.

### Finally, we will be using ThingSpeak to convert collected data into a more readable graph.

### Laboratory Procedure

### Arduino Setup

### For the purpose of these labs, it is highly recommended that you use the desktop version of Arduino IDE rather than the browser version. The software can be downloaded at <https://www.arduino.cc/en/software>.

### From here you can follow this tutorial for basic setup of your board. <https://www.arduino.cc/en/Guide/MKR1000>

### After completing the full setup of your board (including testing the “Blink” example), you may proceed to the lab. <https://create.arduino.cc/projecthub/arduino/plant-communicator-7ea06f>

### IMPORTANT NOTES:

### *For zapier:*

### When editing url, remove the backslash on yours and only add from question mark forward --“?temperature=0&moisture=0&light=0&warning=0”

### *In first sketch:*

### After delay(2000) add semicolon

### After fixing, add this on the next line => exit(0);

### Moisture is misspelled in send email statement

### *On Arduino IDE*

### Under tools make sure the board is changed to mkr1000, if this is not an option you need to go through the setup tutorial again.

### Change port to correct port (probably com5)

### Make sure to hit upload certificates

### After completing all your sketch tests and going through procedure, you will get to the final sketch. The original sketch provided has many errors so I have uploaded a revised version of the final sketch under the lab assignments page. In order for this all to work properly, you must do the following.

### 1. Download the original final sketch from the Arduino website. It should download in a zip file. It is very important that you do not just copy and paste this code.

### 2. After unzipping and opening the file, you should see two tabs at the top of your screen. The "secrets" tab includes four lines of code which you must fill in according to your circumstances. It should be filled in this order: Network name, network password, URL from zapier WITHOUT the final backslash, and the write API for the thingspeak you set up.

### 3. Go back to the actual sketch tab, erase all the code and paste the corrected code I have supplied.

### LAB 3 REPORT

Name:

Student No:

### Directions

Lab report is to be in an organized format, submitted as a single Word file with this sheet as your coversheet. Figures and plots are to be neatly labelled showing units on both the vertical and horizontal axis within Matlab or Excel. Plots are to be copied from the Matlab application generated Figure or the Excel generated plot.

Note: Screen shots of the entire desktop will not be accepted.

Upload the file as a MS Word Docx only. Each group member is to submit their own report and attempt their own models.

**The same measured data may be submitted with each team member that is present; however, each student must turn in a report with that data.**

For turn in purposes, please supply the following in your final report:

* A screenshot of the email you receive
* A screenshot of the graphs found on ThingSpeak
* A photo of your circuit

Please allow the graph to collect at least 5 data points before screenshotting (it should collect a data point every minute or so).

If you have any other questions, you are free to email me at [alexismitchell@wustl.edu](mailto:alexismitchell@wustl.edu).

### Report

1. Explain in your own words how the DIY moisture sensor works. Specifically address why a 1 MΩ resistor is used in the circuit.
2. I am trying to grow a pepper plant to spice up my life. Pepper plants thrive at temperatures higher than 55 degrees Fahrenheit. Assume the temperature sensor outputs .5V. Use the code in the sketch to manually convert the reading to (a) Celsius and (b) Fahrenheit. Will my plant die?