Instruction Manual for the SolarSENSE System

March 11, 2022

# Introduction

The SolarSENSE Agricultural Project seeks to improve dynamic farming practices. There are two major components that are required to complete this task. These are the SolarSENSE Sensor (hereafter called just SolarSENSE), and the SolarSPELL Raspberry Pi. The Sensor Assembly takes measurements of the soil and sends it to a website hosted by the Raspberry Pi. The Raspberry Pi publishes this data onto a website that is only accessible on a particular WIFI network.

This document is a set of instructions for setting up the SolarSENSE and WIFI host. This set of instructions is accurate for March of 2022. If you are reading this at least one year into the future, some of these instructions will likely be outdated.

# The Hardware You Are Given

The Sensor is a meter-long rod with a set of electronics inside of it.

* The Printed Circuit Board.

A picture containing text

Description automatically generated

* Wires leading to the sensor’s solar panel, battery and LED system.
* A switch to connect or disconnect the power system to the circuit board

You should already have access to a SolarSPELL Raspberry Pi. This contains:

A picture containing electronics, circuit

Description automatically generated

* A Raspberry Pi with a clear case. The Raspberry Pi is a green Printed Circuit Board. Much like the sensor, it will have many components soldered onto it. It has several USB ports.
* Wires connecting it to the solar power system.

# Complete Steps for Setup

## Required Components:

* The SolarSENSE Sensor
* The SolarSPELL Digital Library Kit
* An Ethernet cable
* A USB-C cable
* A laptop

## Upgrading the SolarSPELL:

1. Use the Digital Library Management System to add a library for the SolarSENSE sensor system. This will add pages to the SolarSPELL website so that you can see the readings from the soil sensors.
2. Connect the Raspberry Pi to your laptop with the ethernet cable.
3. If you are using Mac OS:
   1. Navigate to System Preferences -> Sharing.
   2. Turn on internet sharing for thunderbolt ethernet.
   3. Connect your raspberry pi using an ethernet cable.
   4. Navigate to system preferences -> Network
   5. Wait until thunderbolt ethernet listed in the left changes from "Not connected" to "self-assigned ip"
   6. Navigate to /private/var/db/
   7. Open up dhcpd\_leases and record the ip address listed under raspberrypi
   8. Open the terminal and enter: $ ssh pi@YOUR\_IP where YOUR\_IP is the ip you found.
   9. When prompted for a password enter raspberry.
4. If you are using Windows:
   1. Go to control panel -> network sharing center
   2. Click change adapter settings
   3. Right click on your wireless network connection and select properties. (\* You will need an internal network adapter otherwise this connection won't be visible \*)
   4. Click the sharing tab.
   5. Make sure both both boxes are selected and the home network connection is set to LAN.
   6. Apply the changes and reboot your computer.
   7. Attach the raspberry pi to your computer with an ethernet cable.
   8. Connect a screen and keyboard to your raspberry
   9. Type $ ifconfig into the raspberry pi commandline
   10. record the ip address listed under eth0
   11. Set up your preferred SSH client.
   12. Attempt to connect to the raspberry pi and when prompted for a password enter raspberry.
5. If you are using Ubuntu:
   1. Go to control panel -> network sharing center
   2. Click change adapter settings
   3. Right click on your wireless network connection and select properties. (\* You will need an internal network adapter otherwise this connection won't be visible \*)
   4. Click the sharing tab.
   5. Make sure both both boxes are selected and the home network connection is set to LAN.
   6. Apply the changes and reboot your computer.
   7. Attach the raspberry pi to your computer with an ethernet cable.
   8. Connect a screen and keyboard to your raspberry
   9. Type $ ifconfig into the raspberry pi commandline
   10. record the ip address listed under eth0
   11. Set up your preferred SSH client.
   12. Attempt to connect to the raspberry pi and when prompted for a password enter raspberry.

Steps 3-5 came from this site: <https://github.com/bbaikie/SolarSENSE>

1. In the command line, paste “sudo apt install -y mosquitto mosquitto-clients”
2. Enter. Wait for it to finish.
3. Paste “sudo systemctl enable mosquitto.service”
4. Enter. Wait for it to finish.
5. Paste “sudo nano /etc/mosquitto/mosquitto.conf”
6. Enter. It will open a file.
7. Hold down the down-arrow key until you reach the end of the file.
8. Paste “listener 1883”
9. Enter.
10. Paste “allow\_anonymous true”
11. Press CTRL and X to save and exit.
12. Press Y and then enter.
13. Key in “sudo reboot”
14. Enter, and wait for it to reboot.
15. Once it has rebooted, key in “hostname -I”
16. Enter, and see what was returned. The first thing that shows up should be an IP address. Save this address.

Original tutorial: <https://randomnerdtutorials.com/how-to-install-mosquitto-broker-on-raspberry-pi/>

## Troubleshooting the Server:

1. If either the WIFI network does not show up, or the website does not appear, it is possible that the Raspberry Pi failed to setup either during startup.
2. First verify that power is reaching the Pi by checking if the onboard LEDs are lit up.
3. If the LEDs are lit, and the problem persists, unplug and replug the power cord connected to the Raspberry Pi. Wait one minute for the Pi to boot up.
4. If the problem persists, unplug the power cord and then remove the SD card from the Raspberry Pi.
   1. The SD card is a flat chip measuring about 11 mm by 15 mm.
   2. To remove it from the Pi, press down on the end of the chip into the body of the Pi. Then, release and grab the chip.
5. Insert the SD card into a laptop, if there is one available.
   1. Load onto that SD the Raspberry Pi files found at this Github link <https://github.com/bbaikie/SolarSENSE>
   2. Remove the SD card from the laptop
   3. Insert the SD card into the unpowered Raspberry Pi, and power it up.
6. If the problem persists, send the SD card and Raspberry Pi to SolarSPELL.

## Setting up the Sensor:

1. Flip the power switch on the circuit board. The LEDs should turn on to indicate that the system is active.
2. Test the sensor.
   1. Have the SolarSENSE web page open and ready.
   2. Place the sensor end in water or wet soil.
   3. Check that one of the sensor’s humidity has changed on the website, and that a blue LED on the sensor assembly has lit up.
   4. Check that the sensor’s number matches the number marked on the physical sensor itself.
3. If the sensor is working, close the sensor assembly. Try to seal it as best as you can, so that no water may enter.
4. Plant the sensor tip-first into the ground.
5. When you are done with the sensor, pull it out and open the sensor assembly box.
6. Toggle the switch off. All LEDs should remain off.

## Troubleshooting the Sensor:

Fixing the Software of the sensor is much more complicated than fixing the hardware. Follow the next steps carefully.

1. There are files at the GitHub linked here: <https://github.com/Xinyuan-LilyGO/LilyGo-HiGrow>. [REPLACE WITH OUR GITHUB] Download them. Here is a description of the files included.
   1. 3d\_file: Unimportant
   2. Bin: This was added recently. Its purpose is unknown at the moment.
   3. Image: Unimportant
   4. Lib: Includes required libraries.
   5. Schematic: Unimportant
   6. Src: This has the Arduino and header files. Important.
   7. .gitignore: Unknown as of the time of writing. Likely unimportant.
   8. Platformio: Unknown as of the time of writing. Likely unimportant.
   9. README: Instructions. These will be ignored.
2. There are instructions that come with the README. We will not be following those.
3. Relocate all files INSIDE the ‘lib’ folder into the libraries folder in your Arduino directory. Mine was C:\Users\Max\Documents\Arduino\libraries
4. Open the Arduino IDE
   1. If this is not installed, go to the Arduino website and install the IDE (Integrated Development Environment) there. <https://www.arduino.cc/en/software>
5. Change your Arduino device to the ESP32 Dev Kit. This is a two-step process:
   1. Go to File > Preferences and paste <https://dl.espressif.com/dl/package_esp32_index.json> into the “Additional Boards Manager URLs:” text field.
   2. Then, select Tools > Board > ESP32 Arduino > ESP32 Dev Module
6. Now, make changes to the code. Make sure that the code in Arduino matches the code in this document’s appendix.
7. Make tuning changes to the code:
   1. Change the WIFI settings. Near the top of the code page, there is a comment   
      “INSERT WIFI SSID HERE ->”. Key in the name of the SolarSENSE wifi there.
   2. Change the Sensor Number. Find the comment that says   
      “INSERT SENSOR NUMBER HERE ->”. Key in the number of the sensor, matching the number found marked on the sensor.
8. Upload the sketch. (Click the  button)

(add Bluetooth details)