

# SwitchDoc Labs

The Smart Garden System Version 2  
Extender 3D Print Assembly and Test

July 2020  
Version 1.1



**SmartGarden  
System**

## Table of Contents

<b>Cautions when building and using The Smart Garden System.....</b>	<b>1</b>
<b>Errata .....</b>	<b>1</b>
<b>What is The Smart Garden System?.....</b>	<b>2</b>
<b>Preparing for this Manual.....</b>	<b>3</b>
<b>What is needed for your 3D Printed box Assembly? .....</b>	<b>3</b>
<b>Other Parts Needed.....</b>	<b>8</b>
<b>Step by Step Assembly.....</b>	<b>8</b>
<b>What are we doing here? .....</b>	<b>8</b>
<b>Assembling the Smart Garden System Wireless Extender in the 3D Case .....</b>	<b>9</b>
<b>Testing your Smart Garden Setup .....</b>	<b>21</b>
<b>Installing the The Smart Garden System Software and Initial Testing on the Raspberry Pi .....</b>	<b>21</b>
<b>SGS Wireless Extender Board Setup .....</b>	<b>23</b>
<b>Final Testing Your SGS System on the Raspberry Pi.....</b>	<b>28</b>
<b>Testing your Smart Garden Setup .....</b>	<b>30</b>
<b>Installing the The Smart Garden System Software and Initial Testing on the Raspberry Pi .....</b>	<b>30</b>
<b>SGS Wireless Extender Board Setup .....</b>	<b>32</b>
<b>Final Testing Your SGS System on the Raspberry Pi.....</b>	<b>37</b>
<b>Disclaimer .....</b>	<b>39</b>

## Cautions when building and using The Smart Garden System

- 1) Keep all water away from the electronics and power supply at all times!**
- 2) The Smart Garden System is designed for indoor use only and should be placed in a dry environment where no water or rain can reach to avoid short circuiting the electronics**
- 3) Insert the moisture sensor into the CENTER of the flower pot, and keep it near the center of the plant and away from the water coming out of the holes cut in the watering pipe.**
- 4) This is not a toy! Keep it out of reach of young children and pets.**
- 5) SwitchDoc Labs assumes no liabilities in the use of this kit, beyond the refund of the purchase price.**

## Errata

## What is The Smart Garden System?

Did you ever want to build your own remote monitoring and management system for your indoor or outdoor garden? Do you want to share your garden and the weather world wide? This project is for you. You can learn the Raspberry Pi and how to connect to the real world through this easy to build **no-soldering** kit. You can measure soil moisture and then use that as feedback to provide your plant or garden just the right amount of water.

Highly expandable! You can have your Raspberry Pi Base Unit inside your house and have multiple wireless control units in your outdoor garden, greenhouse or in the upstairs bedroom. Up to 250 wireless control units can be connected up to one Raspberry Pi base unit. This allows you to control your truly MASSIVE garden. Or your small one. Either way!

This is a perfect highly expandable project kit for kids with some help from the adults and for adults trying to learn some new things. We have done this before with our successful OurWeather Kickstarter so we know what we are talking about. People all over the world have built the OurWeather weather station with great success. This project has **no soldering** involved and uses Grove connectors to wire everything up! You can't reverse them and blow things up. [Here is our tutorial on the Grove system.](#)

### The Smart Garden System Features

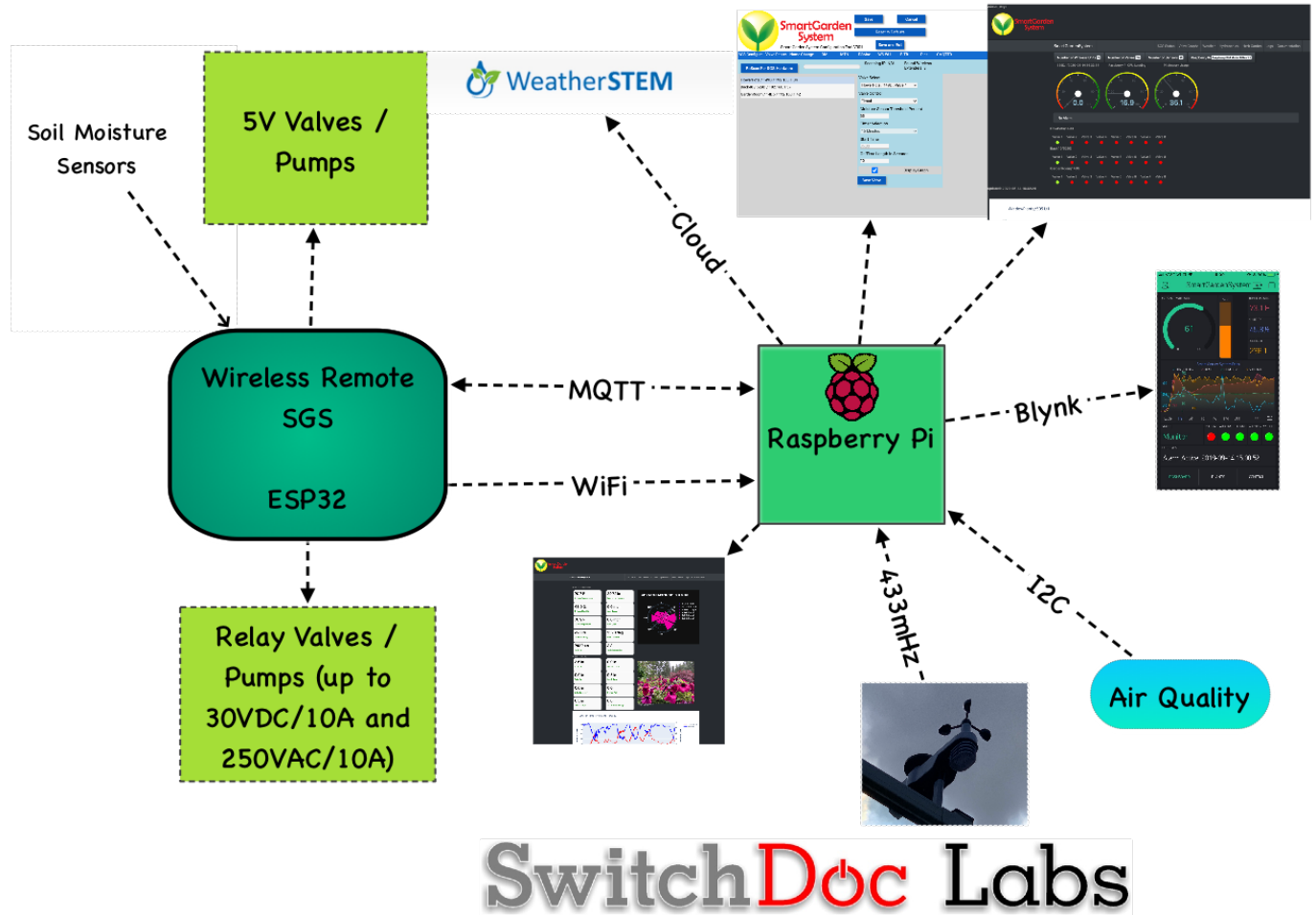
- Control Valves, Pumps and Lights
- Measure your Soil Moisture
- Measure your Sunlight
- Measure your Air Quality
- Measure your Weather
- Show your results on the Internet and your Phone

Easy to build. Easy to learn about the IOT (Internet Of Things) and the Raspberry Pi.



Plants

## Smart Garden System Block Diagram



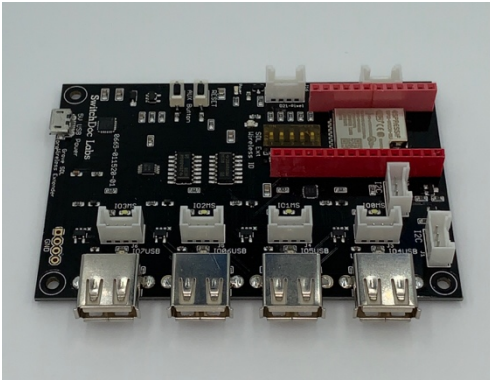
SwitchDoc Labs

### Preparing for this Manual

You should test your system components as shown in the “Smart Garden System Assembly and Test Manual” before assembling your unit in the 3D Printed box. It is much easier to debug your system before you assemble the 3D Printed box.

### What is needed for your 3D Printed box Assembly?

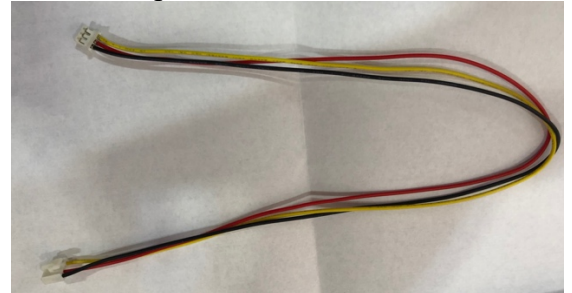
## Part A – Wireless Extender Board



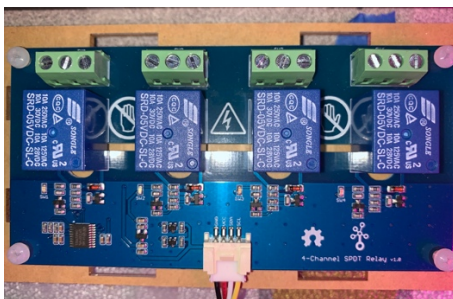
## Part B – Grove Capacitive Moisture Sensor



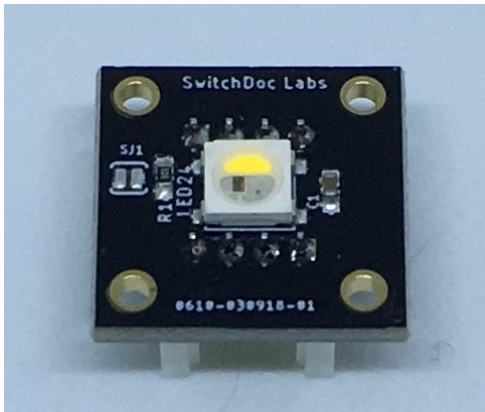
## Part C - Grove to Capacitive Moisture Sensor Cable



## Part D – I2C Grove 4 Relay Board



Part E – Single Pixel RGBW LED



Part F – USB Light Stick



Part G – WiFi Antenna



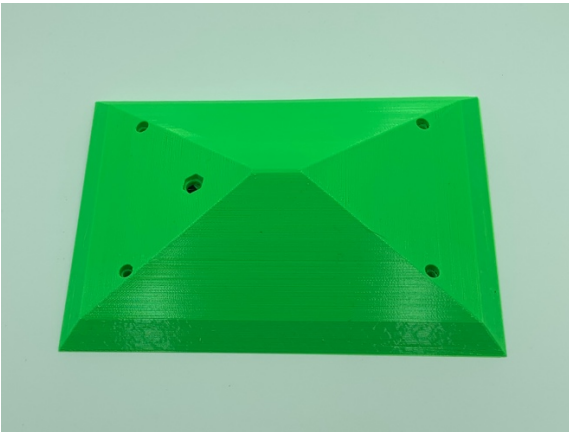
Part H – OLED Board



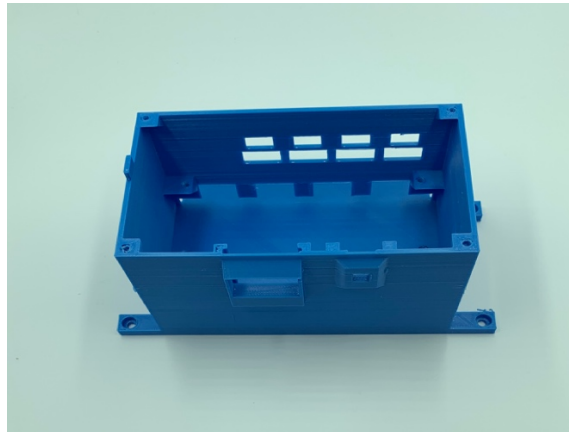
Part I – 20cm Grove Cables (3)



Part J – 3D Printed Extender Top



Part K – 3D Printed Extender Base Unit



Part L – 3D Printed Extender Platform



## Other Parts Needed

**Note: If you bought the 3D Printed Extender Kit from SwitchDoc Labs, these M3 Steel Button Head Screws are included.**

5 - M3 6mm screws  
5 - M3 16mm screws  
M3 Allen wrench



## Step by Step Assembly

**Cautions:** Keep your static charge to a minimum during your assembly and operation. Touch metal before handling parts. Avoid shuffling your feet. Before starting assembly, layout all the parts above and familiarize yourself with the various parts.

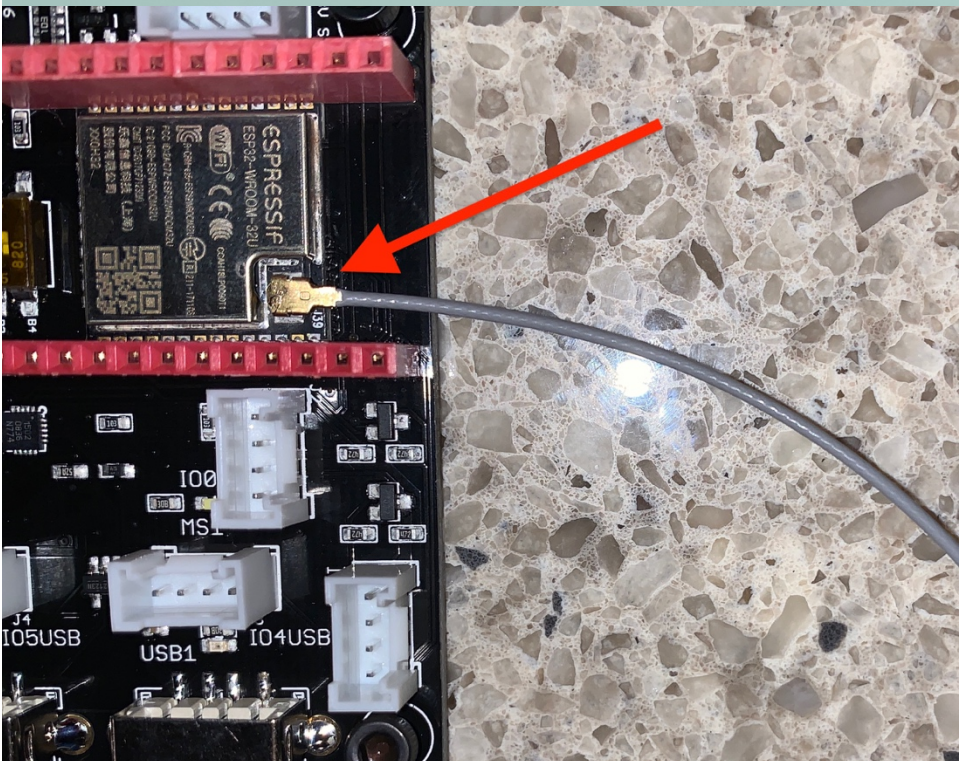
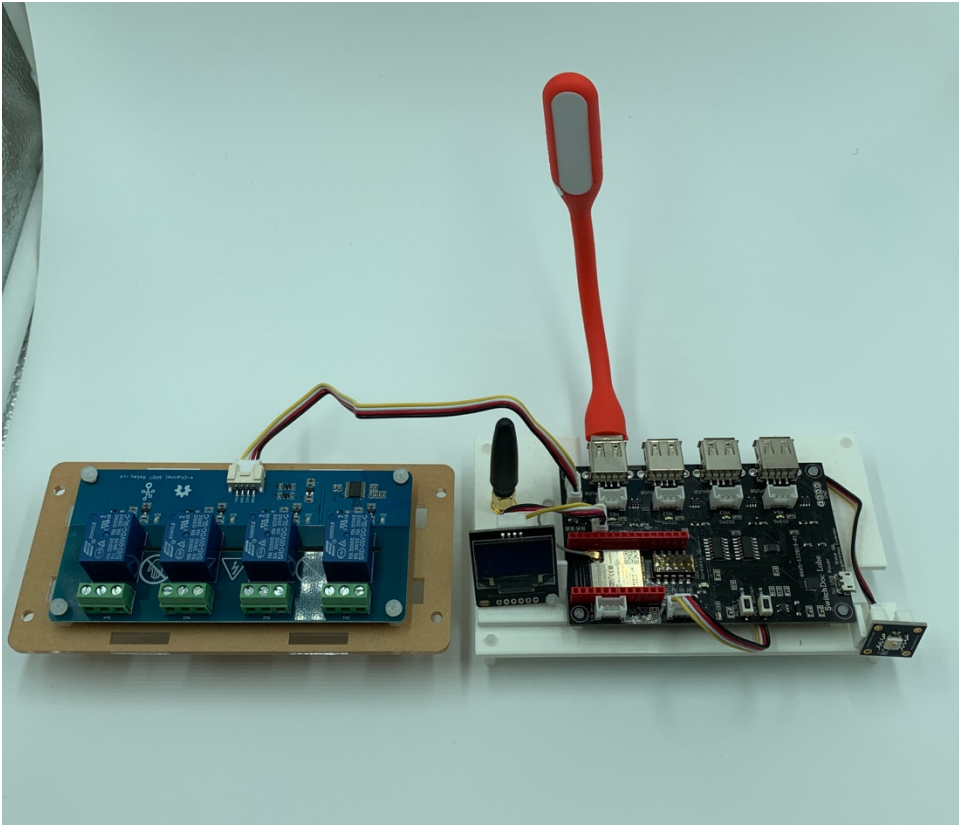
### What are we doing here?

You have already assembled and tested the units in the “Smart Garden Assembly and Test Manual”. You will be now disassembling the extender unit to put it inside the water resistant 3D Printed Extender Boxes.



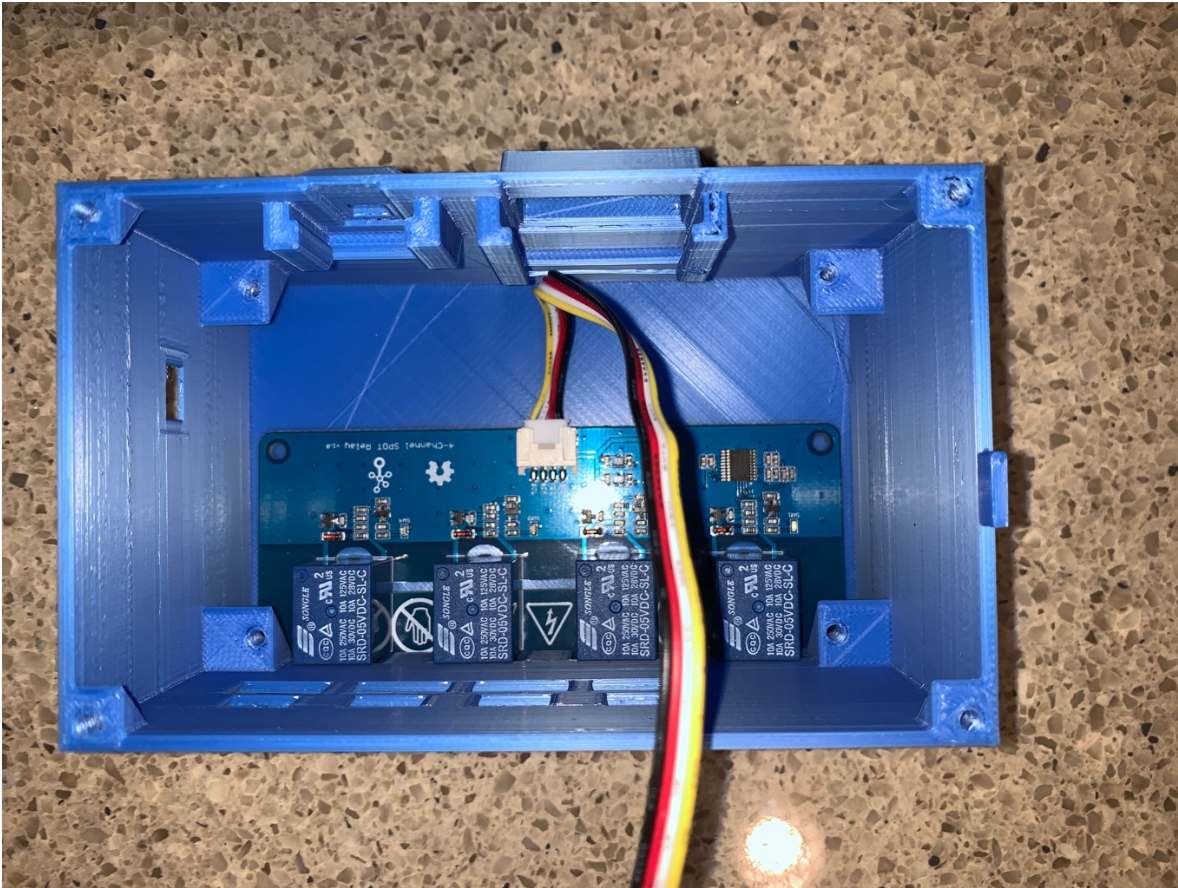
## Assembling the Smart Garden System Wireless Extender in the 3D Case

Step 1) Carefully disassemble the Smart Garden Extender that you built and tested on the 3D printed Stand. Be very careful when you remove the Antenna (Part G) from the Wireless Extender Board (Part A). Use a small flat blade screwdriver and use as little force as possible.



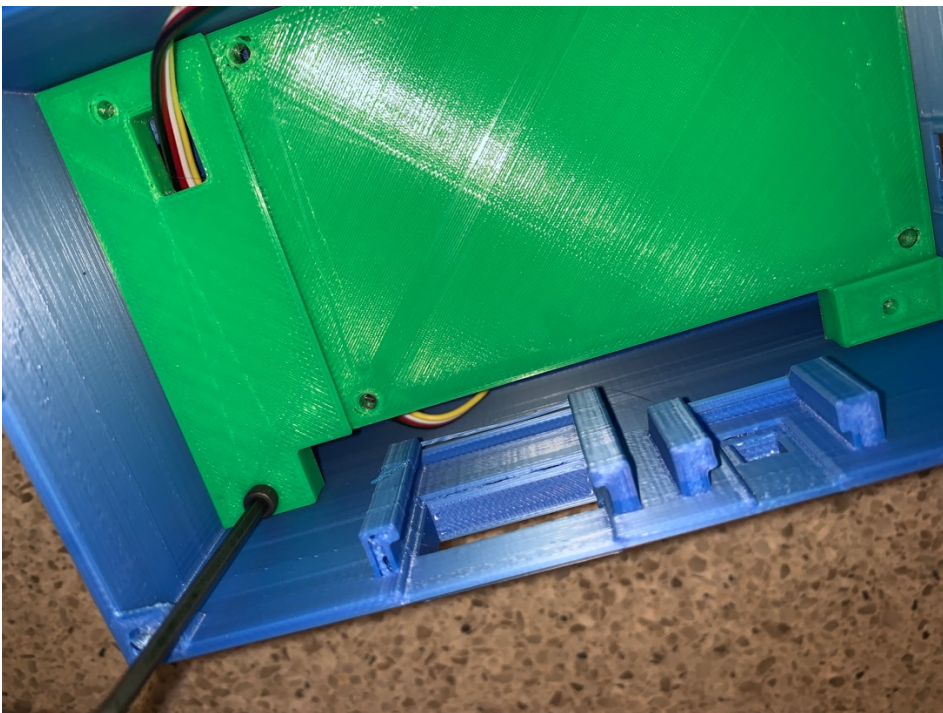
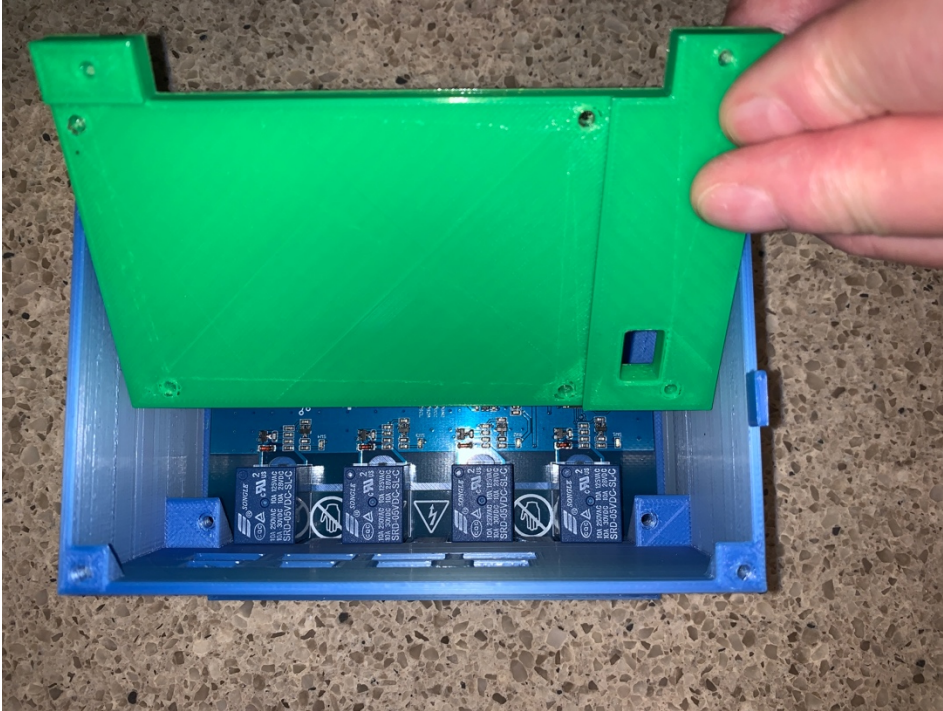


Step 2) Take the Relay Board (Part D) and plug a 20 cm Grove cable (Part I) into the Grove Connector on the Relay Board (Part D). Slide the Relay Board (Part D) into the 3D Printed Extender Base Unit (Part K). Using two M3 6mm screws attach the Relay Board (Part D) to the housing using the two screw holes outside of the 3D Printed Extender Base Unit (Part K).





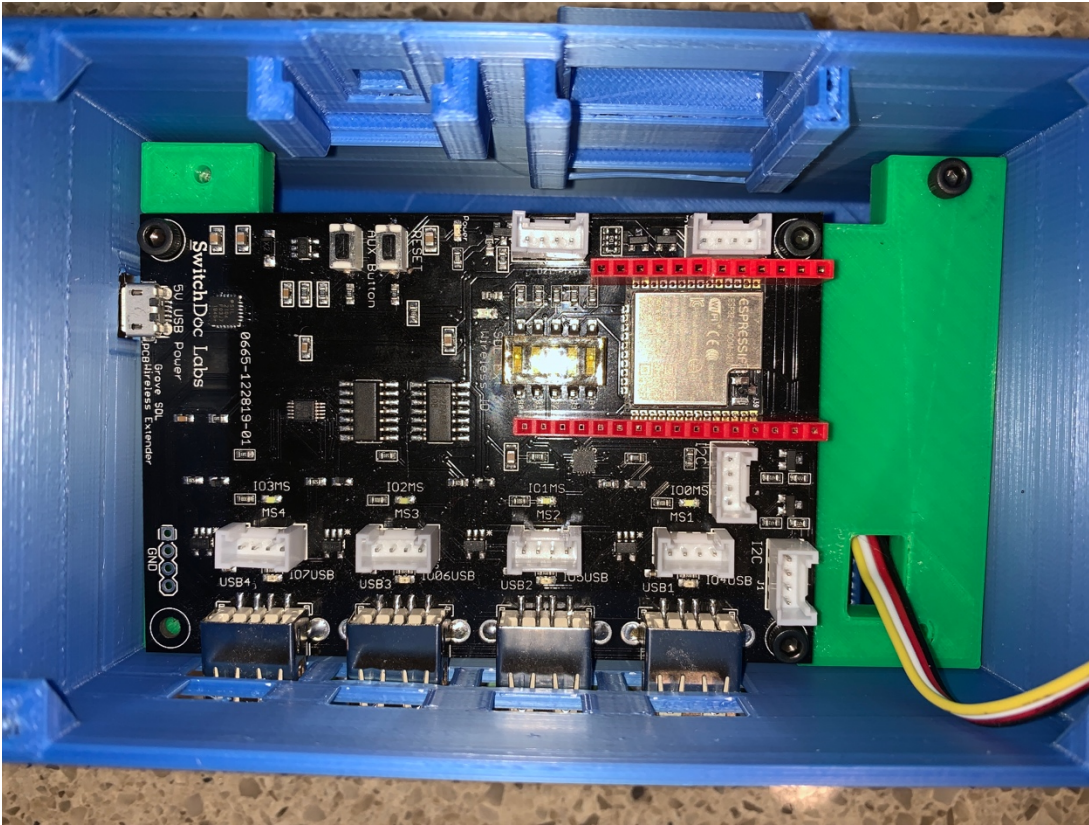
Step 3) Slide the 3D Printed Extender Platform (Part L) into the 3D Printed Extender Base Unit (Part K) with the Extender Platform (Part L) oriented as shown. Route the Grove Cable from the Relay Board (Part D) through the hole on the bottom right side of the Extender Platform (Part L). Make sure you have the Extender Platform (Part L) oriented exactly as shown. Note the cutouts and the indent on the top of the Extender Platform (Part L) for the Wireless Extender Board (Part A). Next use one M3 16mm screw to attach the Extender Platform (Part L) to the Base Unit (Part K). Although there are two mounting holes, one screw is sufficient.



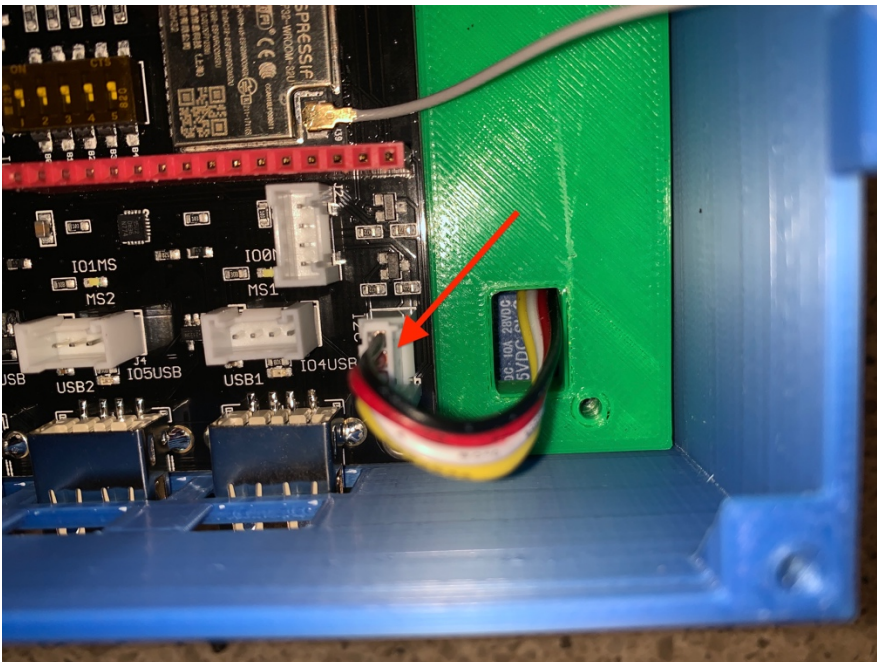




Step 4) Screw down the Wireless Extender Board (Part A) into the Extender Platform (Part L) that is now attached to the Base Unit (Part K).

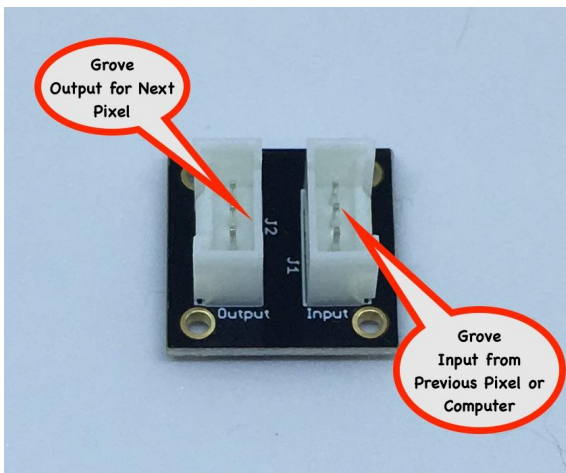


Step 5) Plug the Grove Cable coming from the Relay Board (Part D) into the Grove Connector marked J1 I2C on the Wireless Extender Board (Part A).

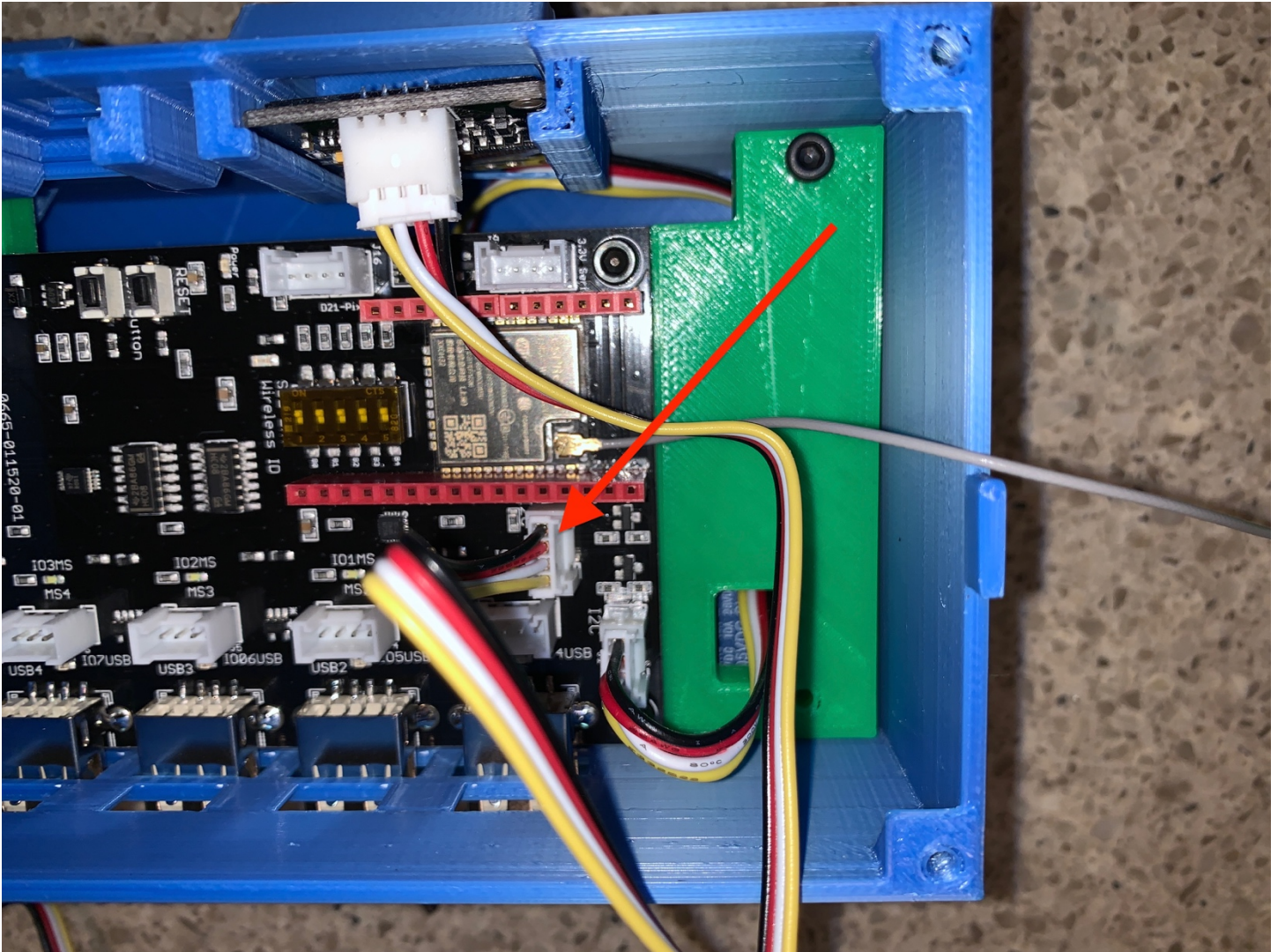




Step 6) Plug the Grove 20cm Cable (Part I) into the OLED Board (Part H) and a second Grove 20cm Cable (Part I) into the INPUT side of the Single Pixel RGBW LED (Part E). Check that you have plugged the Grove 20cm Cable into the correct side of the Single Pixel RGBW LED (Part E).

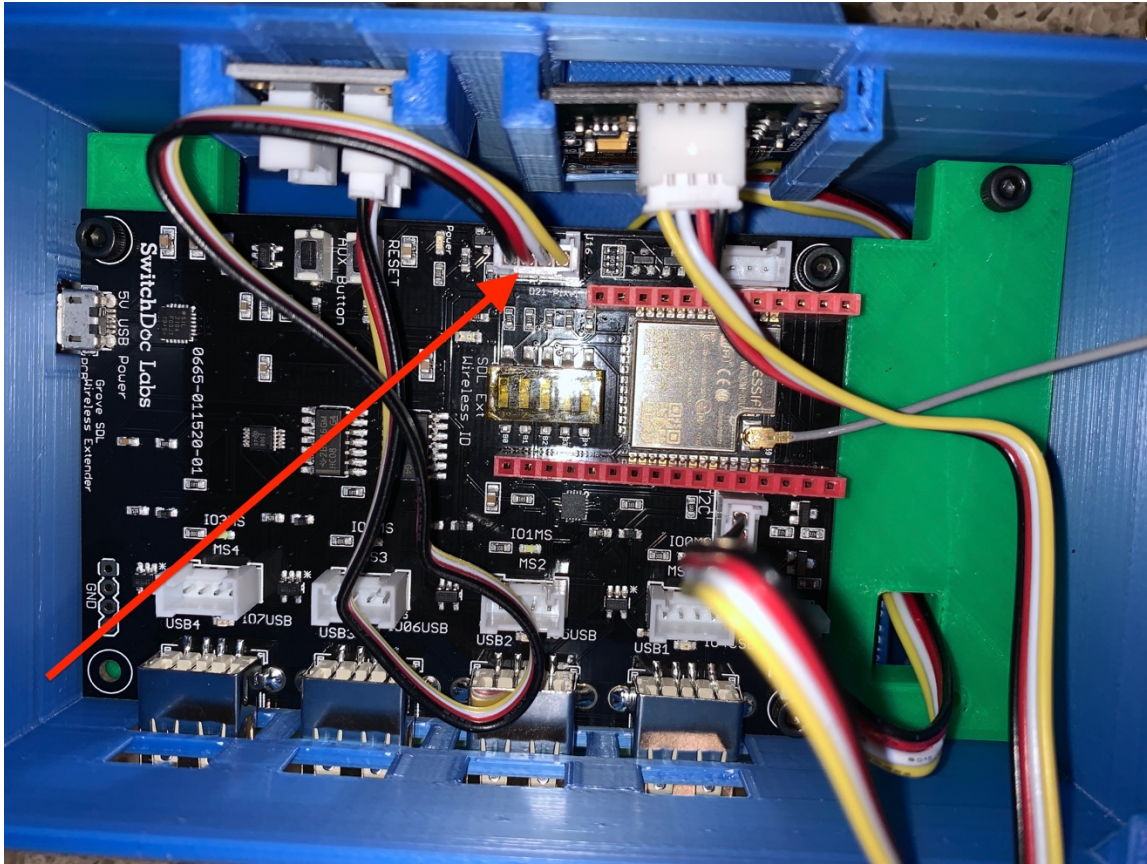


Step 7) Take the other end of the Grove Cable attached to the OLED (Part H) and plug it into the indicated Grove Connector marked J2 I2C on the Wireless Extender Board (Part A). Slide the OLED (Part H) into the slot on the side of the Base Unit (Part K) as shown.

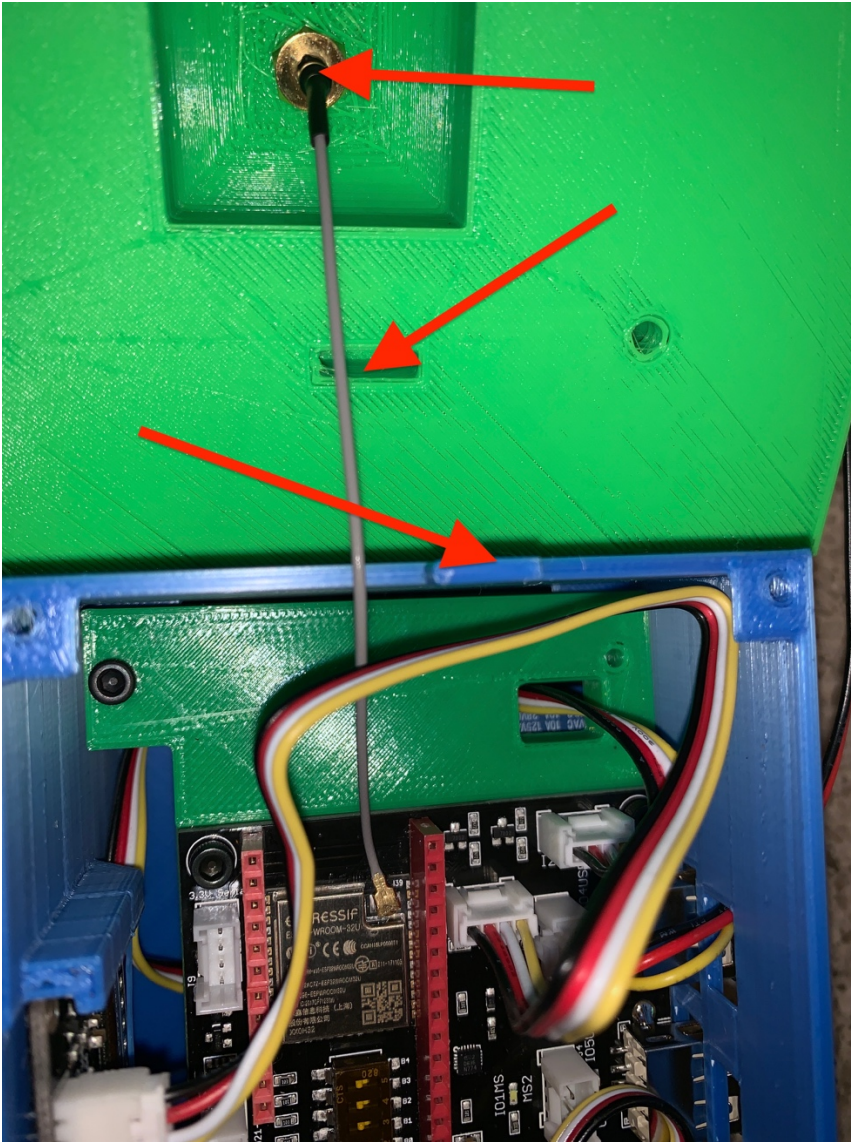




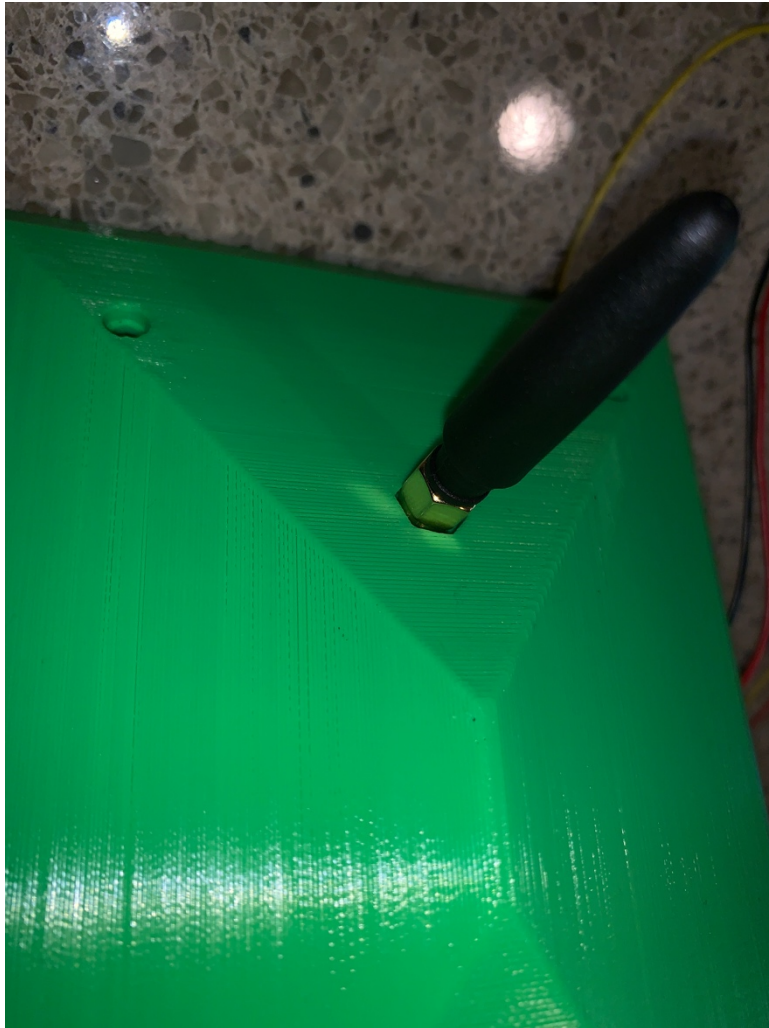
Step 8) Take the Single Pixel (Part E) and plug in the other end of the connected Grove Cable into the Grove Connector marked J16-D2-Pixel on the Wireless Extender Board (Part A) as indicated. Slide the Single Pixel (Part E) into the slot on the side of the Base Unit (Part K).



Step 9) Take the other end (with the screw connector) of the Antenna cable (Part G) and place it in the bottom hole of the 3D Printed Top (Part J). Align the top (Part J) so the slot on the Base (Part K) is on the same side as the slot hole in the Top (Part J).

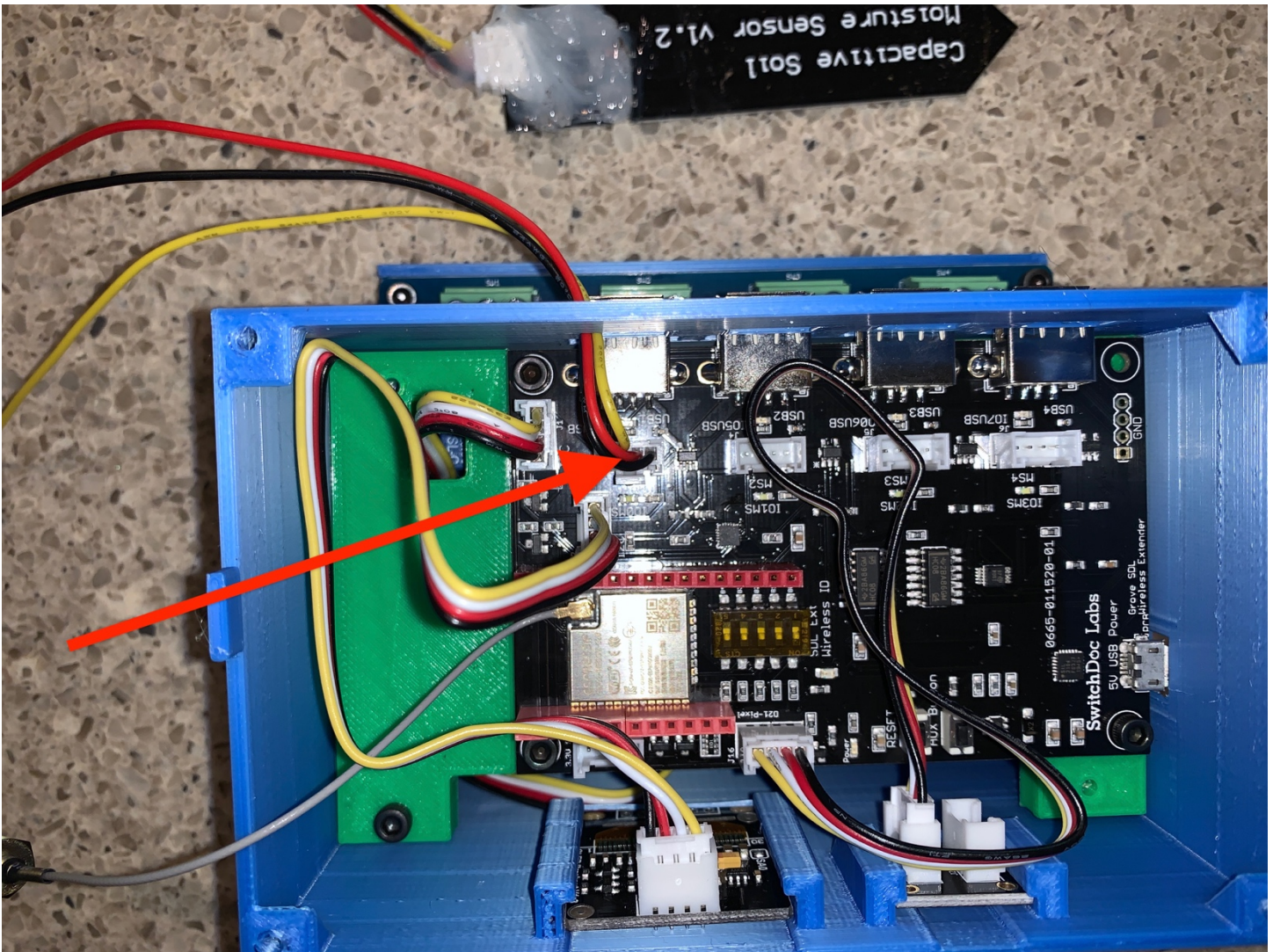


Step 10) Screw the Antenna (Part G) into the hole on top of the Extender Top (Part J). You will not use the Antenna Nut or spacer.





Step 11) Plug one (or more) of the Capacitive Moisture Sensors (Part B) (with the Grove Conversion Cable – Part C) into the Moisture Sensor Grove Connector ports on the Wireless Extender Board (Part A). Start with the port marked MS1 and route the cable through the hole above the USB plugs on the side of the Base Unit (Part K).



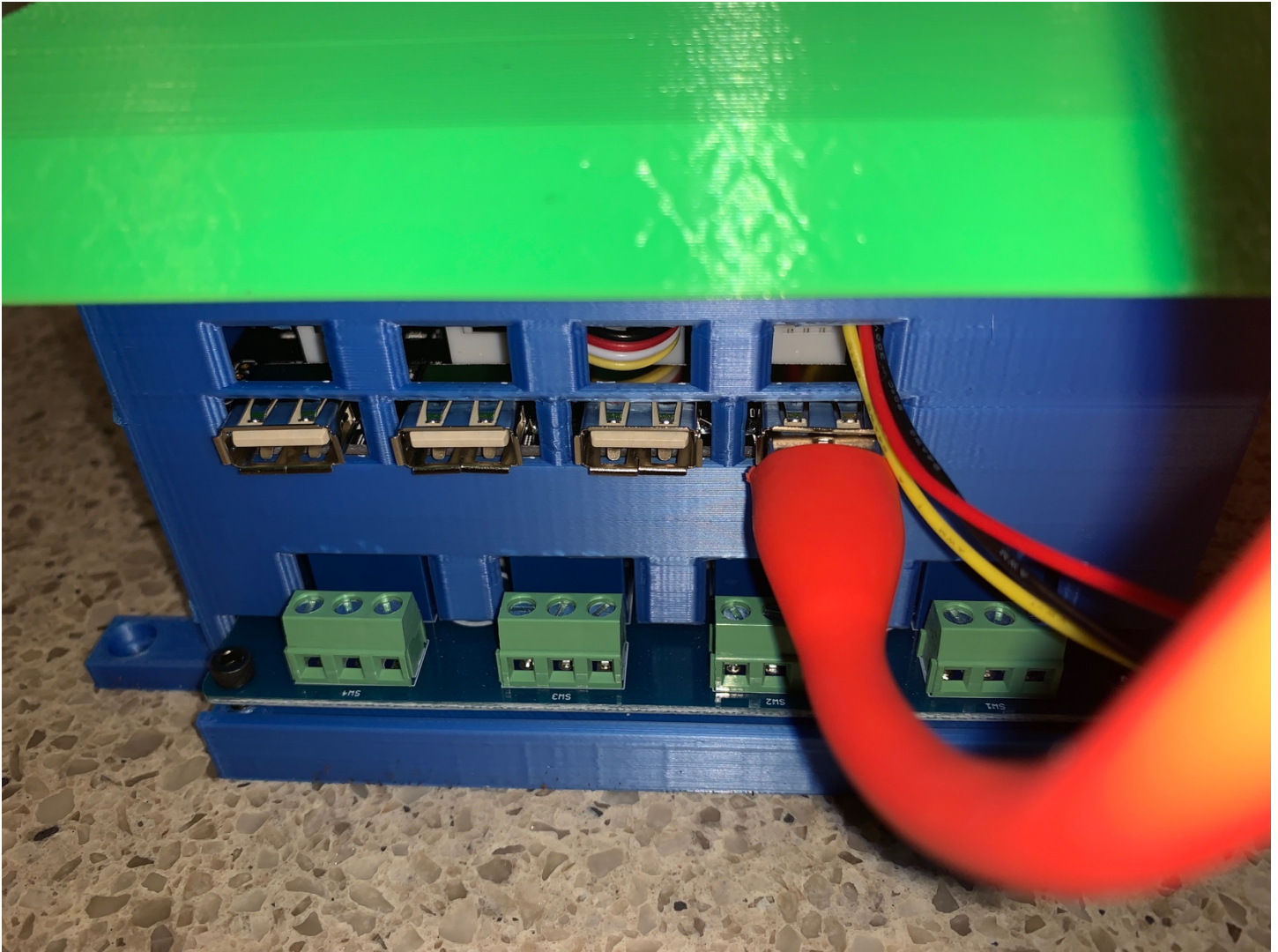
Note that the Capacitive Sensor has been waterproofed. Learn how to do that here:

<https://www.switchdoc.com/2020/07/tutorial-waterproofing-capacitive-moisture-sensors/>



Step 12) Place the Top (Part J) on the Extender Base Unit (Part K) aligning the slot of the of the Top to the Slot on the Base Unit. Loosely connect the Top to the Base Unit using 4 M3 16mm screws. Don't tighten them down as you probably will be opening this up during your garden design.

Step 13) Plug your USB Light Stick (Part ) in USB Port 1 for the initial system testing.



This completes your assembly of your Raspberry Pi for the Smart Garden System in the 3D Case

You have now completed the assembly of The Smart Garden System.

NOTE THAT THERE IS NO WATER ANYWHERE IN THE PROJECT AT THIS TIME!

We will now proceed to initial testing.

## Testing your Smart Garden Setup

This section assumes you have your Raspberry Pi up and running with either the SwitchDoc Labs SDCard, or you have installed the Smart Garden System Software yourself. You can get it on GitHub:

[https://github.com/switchdoclabs/SDL\\_Pi\\_SmartGardenSystem2](https://github.com/switchdoclabs/SDL_Pi_SmartGardenSystem2)

## Installing the The Smart Garden System Software and Initial Testing on the Raspberry Pi

In order to test The Smart Garden System before installing the water, you must be familiar with the Raspberry Pi and the terminal command line as suggested in the Learning Your Raspberry Pi Chapter. You cannot do this test without understanding how to use your Raspberry Pi. You must have Buster or a newer version of the Raspbian Operating System installed.

To save yourself the trouble of installing the Operating System and all the drivers, SwitchDoc Labs provides an SD Card that has the Raspberry Pi operating system, version Buster, and all the Smart Garden System software installed.

For beginners and people short of time, we strongly recommend getting the SD Card.

<https://shop.switchdoc.com/products/16gb-sd-card-with-stretch-smart-garden-system-groveweatherpi>

If you have this SD Card, you can skip steps 2 – 5 below.

Step 1) Plug in your Raspberry Pi. You should see one blue LEDs turn on the SGS Board. In addition, you will see a red LED on the Raspberry Pi board and an occasionally flashing yellow LED on the Raspberry Pi.

Step 2) Open a terminal window on your Raspberry Pi, either on the Pi Graphical Interface or using ssh from another computer.

Step 3) Run the following commands:

```
sudo apt-get update
```

Next, upgrade all your installed packages to their latest versions with the command:

```
sudo apt-get dist-upgrade
```

This may take quite a while.

Now we are ready to start installing the Smart Garden System software.

Step 3) First install I2C on Raspberry Pi – Follow the directions in this link:

<https://learn.adafruit.com/adafruit-raspberry-pi-lesson-4-gpio-setup/configuring-i2c>

Step 4) Next, Install GPIO and apscheduler

```
sudo apt-get install build-essential python-pip python-dev python-smbus git
```

```
git clone https://github.com/adafruit/Adafruit\_Python\_GPIO.git
```

21 Page

Version 1.1 July 2020

```
cd Adafruit_Python_GPIO
```

```
sudo python setup.py install
```

```
sudo pip install --upgrade setuptools pip
```

```
sudo pip install setuptools --upgrade
```

```
sudo pip install APScheduler
```

Step 5) Install the The Smart Garden System software

```
sudo git clone https://github.com/switchdoclabs/SDL_Pi_SmartGardenSystem2
```

Check the README.md in this directory for any new installation instructions for the software.

Step 6) I2C Test - Now we can run the first major test.

Type this into a terminal window:

```
sudo i2cdetect -y 1
```

Here is what you should see:

```
pi@SwitchDocLabs:~/SDL_Pi_SmartGardenSystem2 $ sudo i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  77
pi@SwitchDocLabs:~/SDL_Pi_SmartGardenSystem2
```

This shows the Barometer on the SGS Board at I2C address 0x77.

Next we will set up the Wireless Extender Board and then return to the Raspberry Pi for more testing.

## SGS Wireless Extender Board Setup

The SGS Wireless Extender Board is based on an ESP32 dual core processor with built-in WiFi. Most setup of the Extender Board is done by the Smart Garden System software on the Raspberry Pi, but we do have to connect up to your local WiFi.

The Wireless Extender Board comes pre-programmed with the Smart Garden System. The software is open source and you can see it up on [github.com](https://github.com/switchdoclabs/SDL_ESP32_WirelessSmartGardenExtender):

[https://github.com/switchdoclabs/SDL\\_ESP32\\_WirelessSmartGardenExtender](https://github.com/switchdoclabs/SDL_ESP32_WirelessSmartGardenExtender)

You can find more information about the Wireless Extender Board in the Advanced Operations manual for the Smart Garden System.

Step 1) Unplug the SGS Wireless Extender power supply (Part R) if you have it plugged in.

Step 2) Wait 10 seconds

**NOTE: If you completed the testing in the “Smart Garden System Assembly and Testing Manual”, you already have connected your local WiFi. You can skip Step 3 – Step 11 and move on to the next section in this manual, “Final Testing Your SGS System on the Raspberry Pi”**

Step 3) Plug the SGS Wireless Extender power supply (Part R) power supply back into the outlet and connect the Micro USB Plug to the Wireless Extender Board (Part B) micro USB input plug.

Step 4) You will see a very similar screen on the OLED as below:



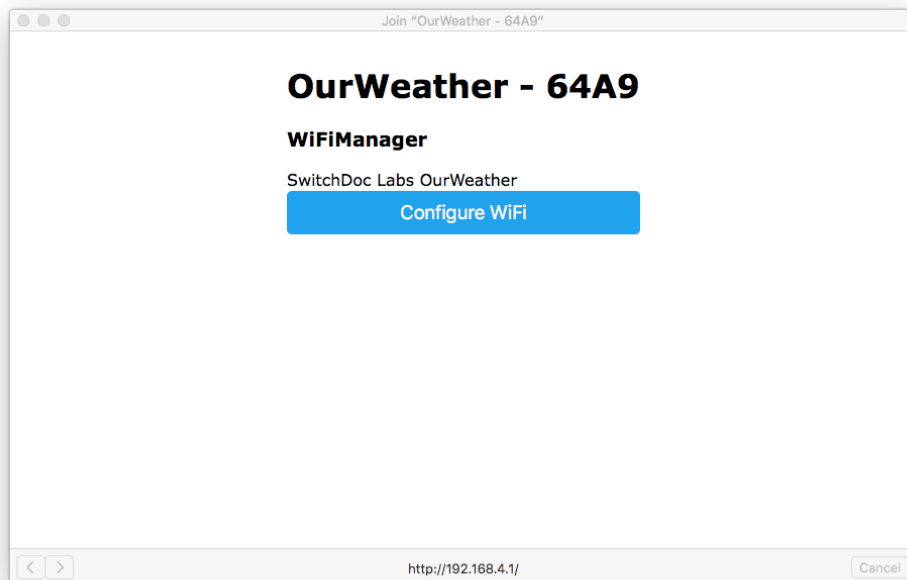
Step 5) Connect your computer or laptop WiFi to the Wireless Access Point name (SGS2WExt-1299 in our example above). Example on a Mac below.



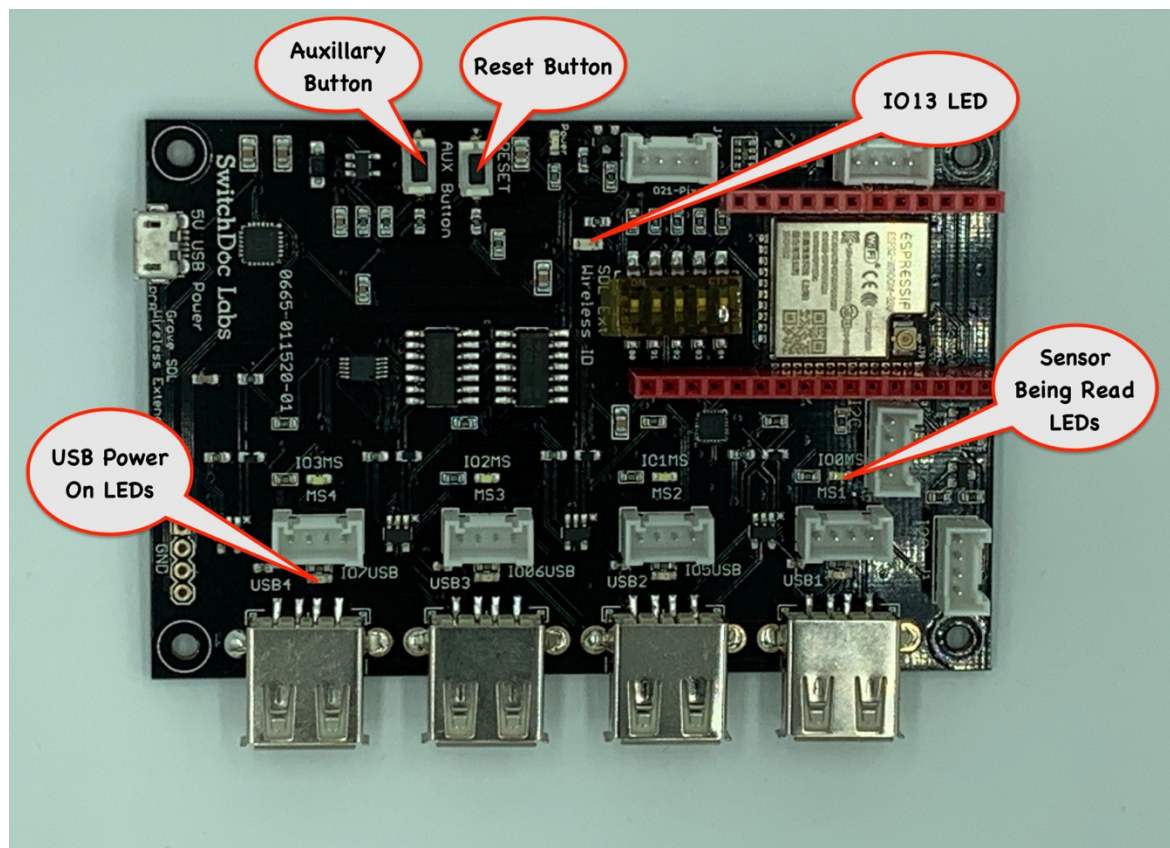
And yes, we are very aware of the Harry Potter reference.



Step 6) Open a web browser (Safari, Chrome preferred - will work with most Internet Explorer versions) and



type in "<http://192.168.4.1>" and shortly you will see a screen as below. Note that you only have 5 minutes to complete the next steps. Otherwise the SGS Wireless Extender will start displaying the system data. If you wish to restart, either repeat Steps 1-3 or push the reset button on the Wireless Extender Board.



Step 7) Click on the Configure WiFi Button on the screen that comes up. It may take a bit before your computer connects. If it doesn't, repeat Step 5 and 6. Rebooting the SGS Wireless Extender computer (by power cycle or using the Reset button) may be required, depending on your network.

Step 8) Fill out the displayed form to set up SGS Wireless Extender WiFi connection.

Each of these fields is described below:

Your Station Name – Optional - The name of your SGS Wireless Extender Station (ex. Howard Flowers)

WiFi Name - The name (SSID) of your local WiFi connection

WiFi Password - The password for your local WiFi connection

An example filled out page follows.

The screenshot shows a web-based configuration interface for a device named "Join 'SGS2WExt-1299'". At the top, there is a list of available WiFi networks with their signal strength indicators:

Network Name	Signal Strength
gracie	100%
arnold7	52%
NETGEAR45	30%
gracie_ext	30%
DIRECT-95-HP OfficeJet 4650	22%
HQJACK74	18%
NETGEAR-Guest	12%

Below the list, there are three input fields:

- Your Station Name**: A text input field with the placeholder "your Station Name".
- WiFi Name**: A text input field with the placeholder "yourwifipassword".
- WiFi Password**: A text input field (partially obscured by the WiFi Name field).

At the bottom of the form is a blue "save" button. The interface also includes a "Cancel" button and a status bar at the bottom showing the IP address "192.168.4.1".

Step 9) Hit Save. SGS Wireless Extender will try to connect. If it doesn't connect (for example if you typed a bad password or WiFi name - watch the screen) then it will reset itself back to Step 4 and you can try again.

Step 10) You are now connected to your local WiFi. SGS Wireless Extender will start displaying system information.



Step 11) When you see the screen below (reboot to see the screen again) write down the IP number displayed in the screen (in our example below, the IP Address is 192.168.1.2):



You may have to reboot your SGS Wireless Extender (by turning the power off or the reset button) to get it to connect to your network.

You should notice the OLED screen cycling through system information and the Pixel LED slowly pulsing green.



Now we will return to the Raspberry Pi for the final test of our system.

## Final Testing Your SGS System on the Raspberry Pi

Step 1) Return to the terminal window on your Raspberry Pi while leaving your SGS Wireless Extender plugged in and connected to the network.

Step 2) Change to the SDL SGS2 directory

```
cd ~/SDL_Pi_SmartGardenSystem2
```

Step 3) Using your favorite text editor, edit the testSGSSystem.py file and change the SGSEXT\_IP variable to the IP address of your SGS Wireless Extender that you copied down in the last section. For example, our test SGS WirelessExtender has an IP address of “192.168.1.2”.

We, in our example, changed the line:

```
SGSEXT_IP = ""
```

To:

```
SGSEXT_IP = "192.168.1.2"
```

Step 4) Run the following command:

```
sudo python3 testSGSSystem.py
```

If everything is good, then you should get a result such as this:

```
pi@SwitchDocLabs:~/SDL_Pi_SmartGardenSystem2 $ sudo python3 testSGSSystem.py
#####
Smart Garden System 2 System Test
#####
2020-07-12 17:23:30.644911

Wireless Extender Address = 192.168.1.2
#####
BMP280 Present and working
#####

#####
Starting Wireless Extender Test
#####
sending REST URL = http://192.168.1.2/setSingleValve?params=admin,1,1,20
#####
Wireless Extender: 192.168.1.2
Successfully Responding
#####
{'return_value': 0, 'id': '1299', 'name': '', 'ipaddress': '192.168.1.2', 'hardware': 'esp32',
'return_string': '', 'connected': True}
#####
sending REST URL = http://192.168.1.2/enableMoistureSensors?params=admin,1,1,1,1
sending REST URL = http://192.168.1.2/readMoistureSensors?params=admin
```

Moisture Sensors:

```
{'return_value': 0, 'id': '1299', 'name': '', 'ipaddress': '192.168.1.2', 'hardware': 'esp32',  
'return_string': '1,28.60,1,100.00,1,100.00,1,100.00,C1,C1,C1,C1', 'connected': True}  
#####
```

You will see the following.

- 1) The Green Glowing LED on your SGS Wireless Extender will Blink White (that's the message being received by the Extender)
- 2) You will see your USB Light Stick (plugged into USB1) turn on for 20 seconds and then turn off
- 3) Moisture Sensor #1 will read something less than 100% depending on your exact sensor.

If any of these tests fail, go back and check your wiring on your SGS Wireless Extender and make sure that both your Raspberry Pi and the SGS Wireless Extender are connected to the same WiFi Network.

If you are still having problems, a great place to go for help is [forum.switchdoc.com](http://forum.switchdoc.com)

Once you have this working, move onto the Smart Garden System Operation and Configuration Manual.

## Testing your Smart Garden Setup

This section assumes you have your Raspberry Pi up and running with either the SwitchDoc Labs SDCard, or you have installed the Smart Garden System Software yourself. You can get it on GitHub:

[https://github.com/switchdoclabs/SDL\\_Pi\\_SmartGardenSystem2](https://github.com/switchdoclabs/SDL_Pi_SmartGardenSystem2)

## Installing the The Smart Garden System Software and Initial Testing on the Raspberry Pi

In order to test The Smart Garden System before installing the water, you must be familiar with the Raspberry Pi and the terminal command line as suggested in the Learning Your Raspberry Pi Chapter. You cannot do this test without understanding how to use your Raspberry Pi. You must have Buster or a newer version of the Raspbian Operating System installed.

To save yourself the trouble of installing the Operating System and all the drivers, SwitchDoc Labs provides an SD Card that has the Raspberry Pi operating system, version Buster, and all the Smart Garden System software installed.

For beginners and people short of time, we strongly recommend getting the SD Card.

<https://shop.switchdoc.com/products/16gb-sd-card-with-stretch-smart-garden-system-groveweatherpi>

If you have this SD Card, you can skip steps 2 – 5 below.

Step 1) Plug in your Raspberry Pi. You should see one blue LEDs turn on the SGS Board. In addition, you will see a red LED on the Raspberry Pi board and an occasionally flashing yellow LED on the Raspberry Pi.

Step 2) Open a terminal window on your Raspberry Pi, either on the Pi Graphical Interface or using ssh from another computer.

Step 3) Run the following commands:

```
sudo apt-get update
```

Next, upgrade all your installed packages to their latest versions with the command:

```
sudo apt-get dist-upgrade
```

This may take quite a while.

Now we are ready to start installing the Smart Garden System software.

Step 3) First install I2C on Raspberry Pi – Follow the directions in this link:

<https://learn.adafruit.com/adafruit-raspberry-pi-lesson-4-gpio-setup/configuring-i2c>

Step 4) Next, Install GPIO and apscheduler

```
sudo apt-get install build-essential python-pip python-dev python-smbus git
```

```
git clone https://github.com/adafruit/Adafruit\_Python\_GPIO.git
```

```
cd Adafruit_Python_GPIO
```

```
sudo python setup.py install
```

```
sudo pip install --upgrade setuptools pip
```

```
sudo pip install setuptools --upgrade
```

```
sudo pip install apscheduler
```

Step 5) Install the The Smart Garden System software

```
sudo git clone https://github.com/switchdoclabs/SDL_Pi_SmartGardenSystem2
```

Check the README.md in this directory for any new installation instructions for the software.

Step 6) I2C Test - Now we can run the first major test.

Type this into a terminal window:

```
sudo i2cdetect -y 1
```

Here is what you should see:

```
pi@SwitchDocLabs:~/SDL_Pi_SmartGardenSystem2 $ sudo i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  77
pi@SwitchDocLabs:~/SDL_Pi_SmartGardenSystem2
```

This shows the Barometer on the SGS Board at I2C address 0x77.

Next we will set up the Wireless Extender Board and then return to the Raspberry Pi for more testing.

## SGS Wireless Extender Board Setup

The SGS Wireless Extender Board is based on an ESP32 dual core processor with built-in WiFi. Most setup of the Extender Board is done by the Smart Garden System software on the Raspberry Pi, but we do have to connect up to your local WiFi.

The Wireless Extender Board comes pre-programmed with the Smart Garden System. The software is open source and you can see it up on [github.com](https://github.com/switchdoclabs/SDL_ESP32_WirelessSmartGardenExtender):

[https://github.com/switchdoclabs/SDL\\_ESP32\\_WirelessSmartGardenExtender](https://github.com/switchdoclabs/SDL_ESP32_WirelessSmartGardenExtender)

You can find more information about the Wireless Extender Board in the Advanced Operations manual for the Smart Garden System.

Step 1) Unplug the SGS Wireless Extender power supply (Part R) if you have it plugged in.

Step 2) Wait 10 seconds

Step 3) Plug the SGS Wireless Extender power supply (Part R) power supply back into the outlet and connect the Micro USB Plug to the Wireless Extender Board (Part B) micro USB input plug.

Step 4) You will see a very similar screen on the OLED as below:

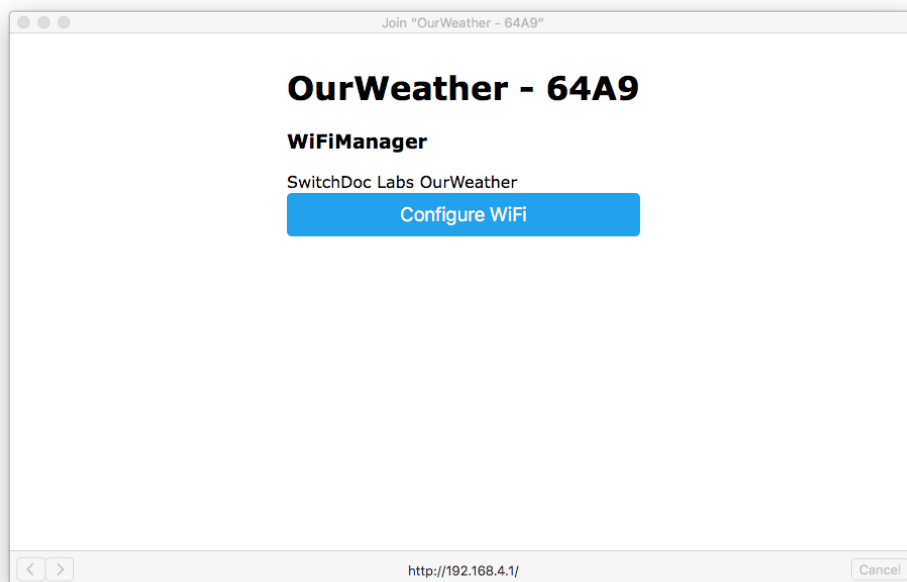


Step 5) Connect your computer or laptop WiFi to the Wireless Access Point name (SGS2WExt-1299 in our example above). Example on a Mac below.

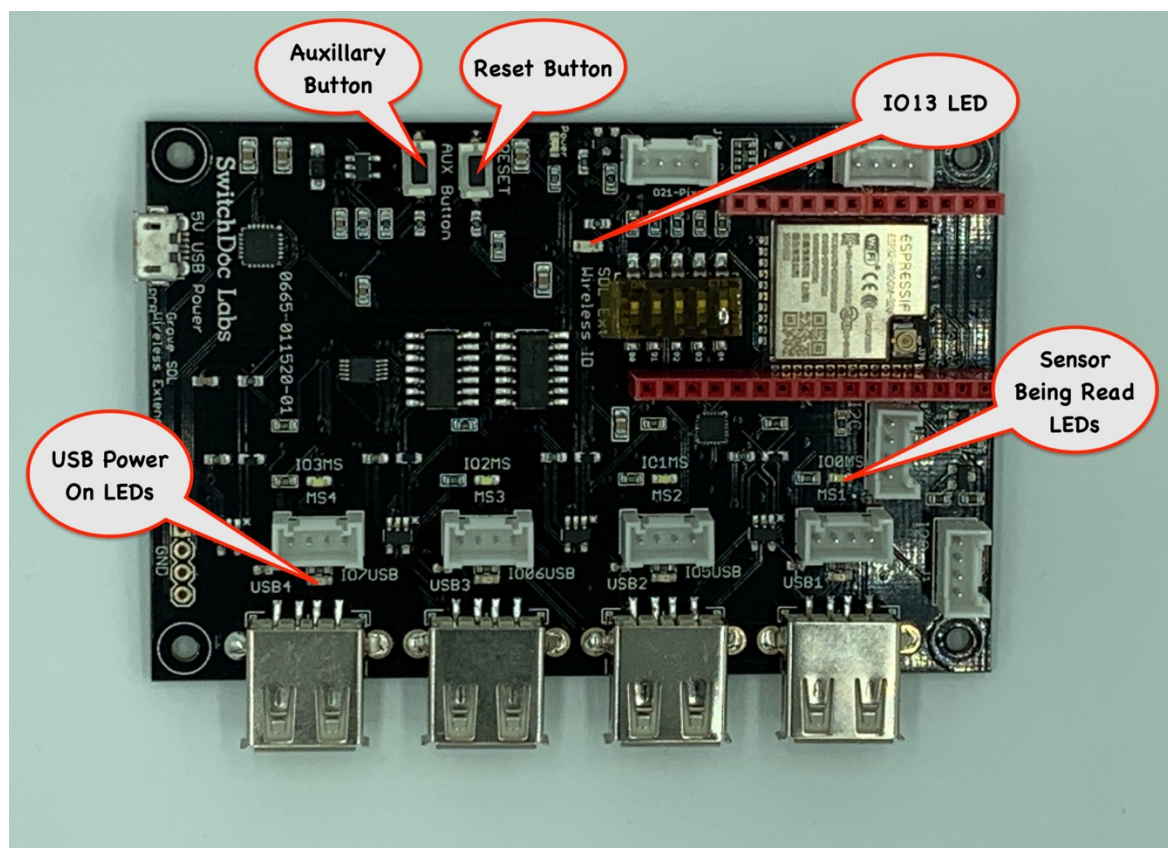


And yes, we are very aware of the Harry Potter reference.

Step 6) Open a web browser (Safari, Chrome preferred - will work with most Internet Explorer versions) and



type in "<http://192.168.4.1>" and shortly you will see a screen as below. Note that you only have 5 minutes to complete the next steps. Otherwise the SGS Wireless Extender will start displaying the system data. If you wish to restart, either repeat Steps 1-3 or push the reset button on the Wireless Extender Board.



Step 7) Click on the Configure WiFi Button on the screen that comes up. It may take a bit before your computer connects. If it doesn't, repeat Step 5 and 6. Rebooting the SGS Wireless Extender computer (by power cycle or using the Reset button) may be required, depending on your network.

Step 8) Fill out the displayed form to set up SGS Wireless Extender WiFi connection.

Each of these fields is described below:

Your Station Name – Optional - The name of your SGS Wireless Extender Station (ex. Howard Flowers)

WiFi Name - The name (SSID) of your local WiFi connection

WiFi Password - The password for your local WiFi connection

An example filled out page follows.

Network Name	Signal Strength
gracie	100%
arnold7	52%
NETGEAR45	30%
gracie_ext	30%
DIRECT-95-HP OfficeJet 4650	22%
HQJACK74	18%
NETGEAR-Guest	12%

your Station Name

Your Station Name

gracie

WiFi Name

yourwifipassword

WiFi Password

save

192.168.4.1

Cancel



Step 9) Hit Save. SGS Wireless Extender will try to connect. If it doesn't connect (for example if you typed a bad password or WiFi name - watch the screen) then it will reset itself back to Step 4 and you can try again.

Step 10) You are now connected to your local WiFi. SGS Wireless Extender will start displaying system information.



Step 11) When you see the screen below (reboot to see the screen again) write down the IP number displayed in the screen (in our example below, the IP Address is 192.168.1.2):



You may have to reboot your SGS Wireless Extender (by turning the power off or the reset button) to get it to connect to your network.

You should notice the OLED screen cycling through system information and the Pixel LED slowly pulsing green.

Now we will return to the Raspberry Pi for the final test of our system.

## Final Testing Your SGS System on the Raspberry Pi

Step 1) Return to the terminal window on your Raspberry Pi while leaving your SGS Wireless Extender plugged in and connected to the network.

Step 2) Change to the SDL SGS2 directory

```
cd ~/SDL_Pi_SmartGardenSystem2
```

Step 3) Using your favorite text editor, edit the testSGSSystem.py file and change the SGSEXT\_IP variable to the IP address of your SGS Wireless Extender that you copied down in the last section. For example, our test SGS WirelessExtender has an IP address of “192.168.1.2”.

We, in our example, changed the line:

```
SGSEXT_IP = ""
```

To:

```
SGSEXT_IP = "192.168.1.2"
```

Step 4) Run the following command:

```
sudo python3 testSGSSystem.py
```

If everything is good, then you should get a result such as this:

```
pi@SwitchDocLabs:~/SDL_Pi_SmartGardenSystem2 $ sudo python3 testSGSSystem.py
#####
Smart Garden System 2 System Test
#####
2020-07-12 17:23:30.644911

Wireless Extender Address = 192.168.1.2
#####
BMP280 Present and working
#####

#####
Starting Wireless Extender Test
#####
sending REST URL = http://192.168.1.2/setSingleValve?params=admin,1,1,20
#####
Wireless Extender: 192.168.1.2
Successfully Responding
#####
{'return_value': 0, 'id': '1299', 'name': '', 'ipaddress': '192.168.1.2', 'hardware': 'esp32',
'return_string': '', 'connected': True}
#####
sending REST URL = http://192.168.1.2/enableMoistureSensors?params=admin,1,1,1,1
sending REST URL = http://192.168.1.2/readMoistureSensors?params=admin
```

Moisture Sensors:

```
{'return_value': 0, 'id': '1299', 'name': '', 'ipaddress': '192.168.1.2', 'hardware': 'esp32',  
'return_string': '1,28.60,1,100.00,1,100.00,1,100.00,C1,C1,C1,C1', 'connected': True}  
#####
```

You will see the following.

- 4) The Green Glowing LED on your SGS Wireless Extender will Blink White (that's the message being received by the Extender)
- 5) You will see your USB Light Stick (plugged into USB1) turn on for 20 seconds and then turn off
- 6) Moisture Sensor #1 will read something less than 100% depending on your exact sensor.

If any of these tests fail, go back and check your wiring on your SGS Wireless Extender and make sure that both your Raspberry Pi and the SGS Wireless Extender are connected to the same WiFi Network.

If you are still having problems, a great place to go for help is [forum.switchdoc.com](http://forum.switchdoc.com)

Once you have this working, move onto the Smart Garden System Operation and Configuration Manual.

## Disclaimer

SwitchDoc Labs, LLC takes no responsibility for any physical injuries and possession loss caused by those reasons which are not related to product quality, such as operating without following the operating manual and cautions, natural disasters or force majeure.

SwitchDoc Labs, LLC has compiled and published this manual which covers the latest product description and specification. The contents of this manual are subject to change without notice.