SwitchDoc Labs

WeatherSense
Solar Powered AfterShock Earthquake Sensor
Assembly and Test Manual

June 2021
Version 1.1

Table of Contents

Cautions when building and using WeatherSense Sensors

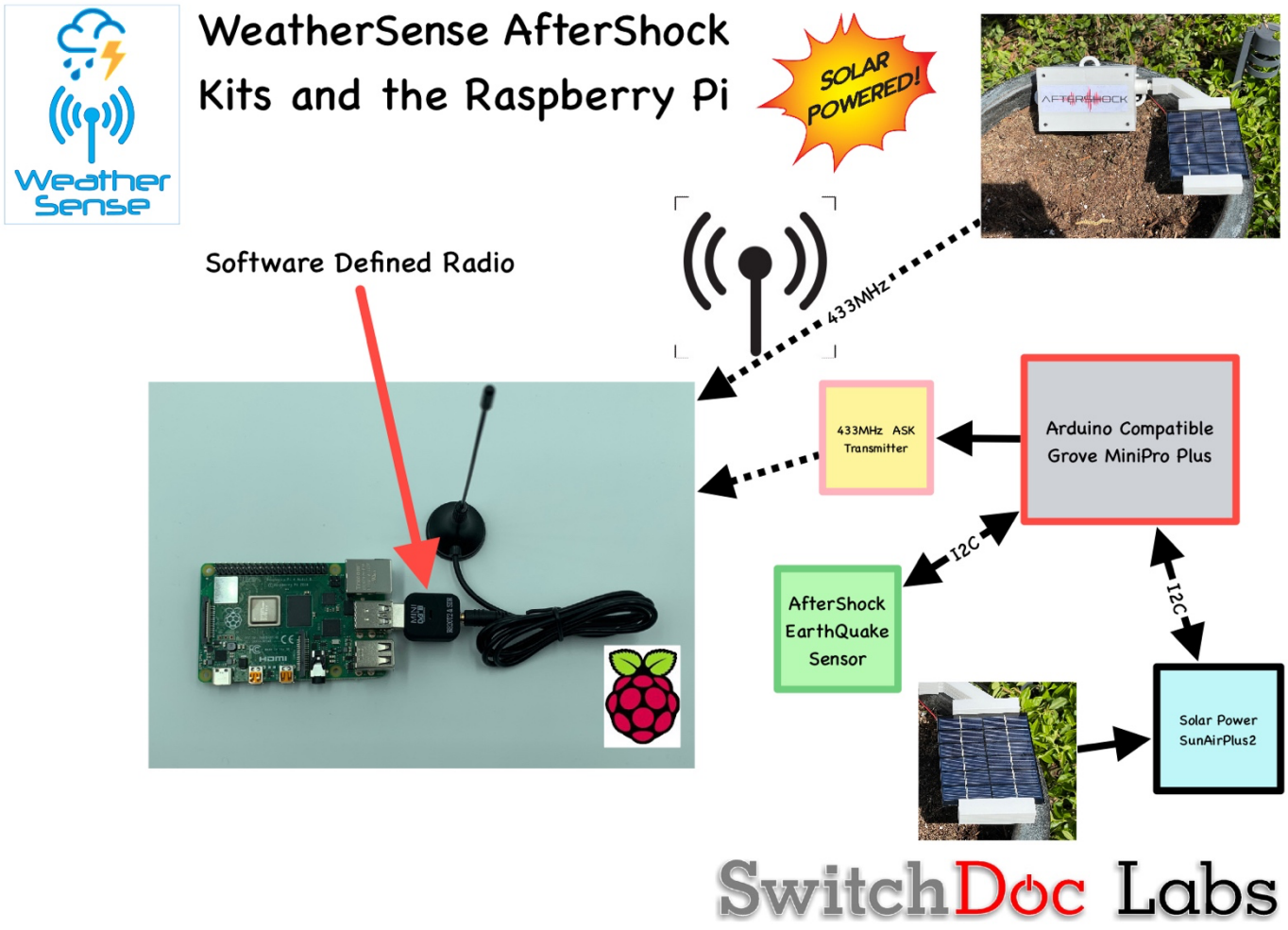
1) Keep all water away from the electronics and power supply at all times!
2) This is not a toy! Keep it out of reach of young children and pets.
3) SwitchDoc Labs assumes no liabilities in the use of this kit, beyond the refund of the purchase price.

Errata

What is The WeatherSense AfterShock Solar Powered Earthquake Detector Sensor?

Easy to build. Easy to learn about the IoT (Internet of Things) and the Raspberry Pi.

The **heart** of the new WeatherSense Sensor is the our new 433MHz MiniProPlus CPU board in working in conjunction with the SwitchDoc Labs AfterShock Earthquake Detector Board.



The **WeatherSense AfterShock** kit is so simple that even middle school children can build it with just a little adult help for configuration and installation.

Full Open Source Arduino IDE compatible C Software that you can Modify and the open source Python3 software for the Raspberry Pi.

We provide the Python3 software (for the Raspberry Pi) and C for the WeatherSense AfterShock Board. All open source with the kit. The Pure Python software can be modified to add new sensors, support new cloud software and connect up to your own projects and software.

Before You Build Your WeatherSense AfterShock System

You should build and test your WeatherSense AfterShock system as below BEFORE you put it in the optional 3D Printed case. Get it working first, then put it in the case. Believe us, it is always easier to debug the system before you close it up in the case! The manual for the case and weatherproofing is called the "WeatherSense AfterShock Weatherproofing, Assembly and Test Manual". All manuals are available on the WeatherSense AfterShock Product page on shop.switchdoc.com.

Step by Step Assembly and Parts List

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.

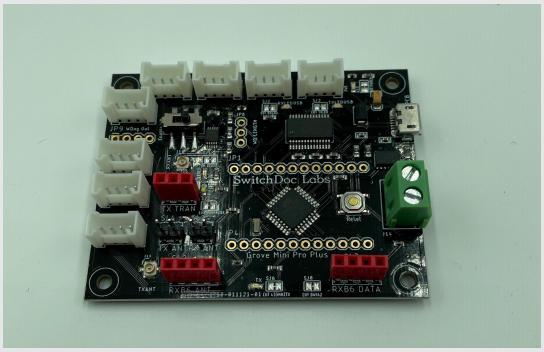
What are we doing here?

We are assembling the WeatherSense AfterShock Earthquake System.

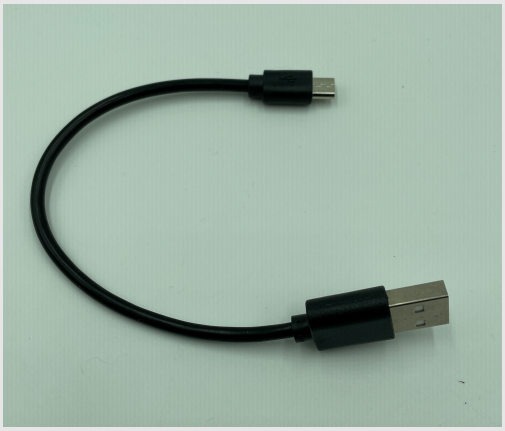
In this manual, we are going to assemble the WeatherSense AfterShock system and test all the functions. If you plan to put WeatherSense AfterShock outside, and after you complete this manual, proceed to the "WeatherSense AfterShock Weatherproofing, Assembly and Test Manual" on the WeatherSense AfterShock Product Page.

Parts List

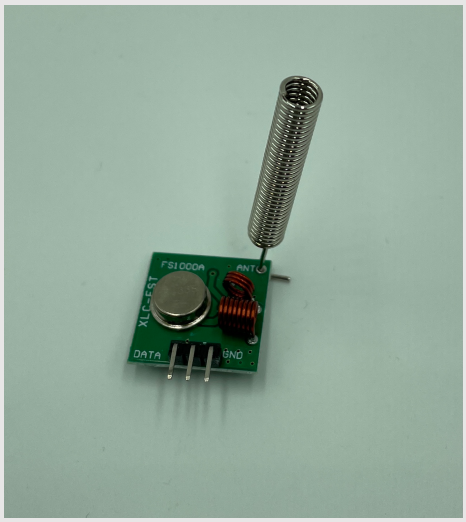
Part A
SwitchDoc Labs Grove Mini Pro Plus



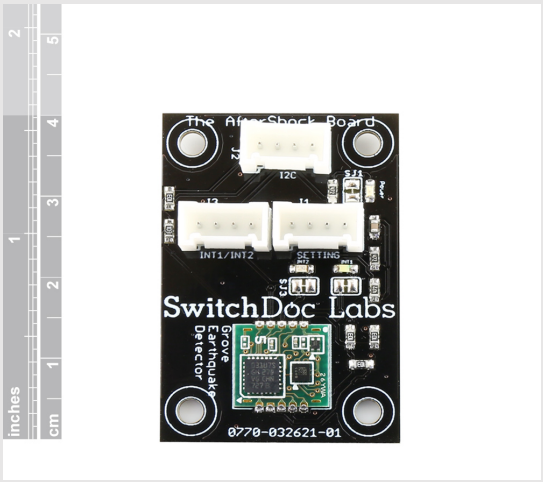
Part B
Short USB A to Micro USB



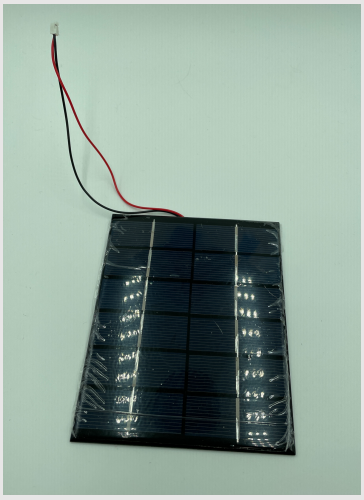
Part C
433MHz Transmitter with Antenna



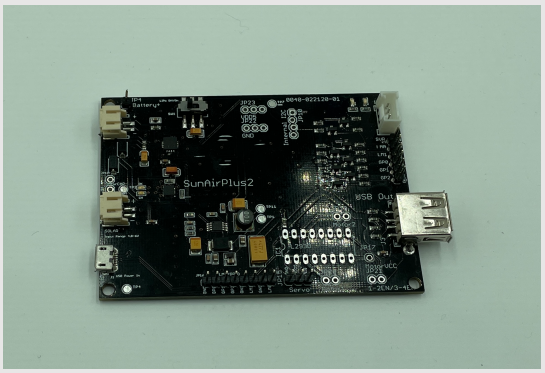
Part D
AfterShock Earthquake Detector



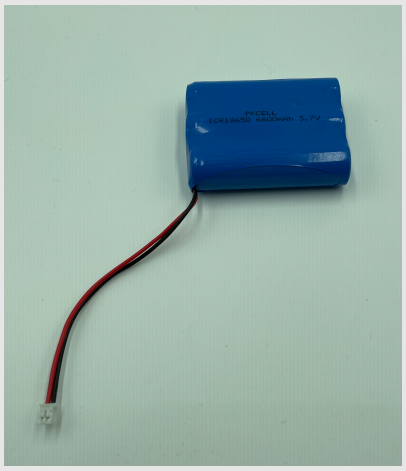
Part E
330mA 6V Solar Panel



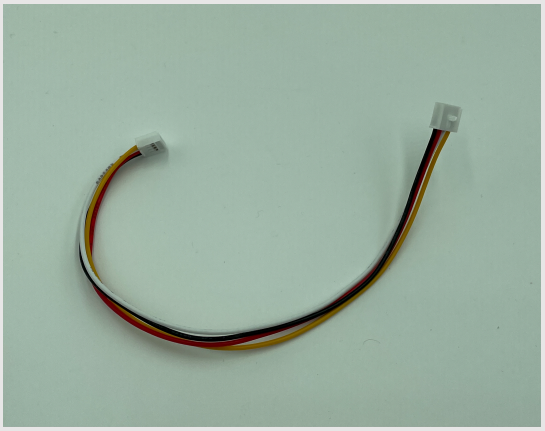
Part F
SunAirPlus2 Solar Power Data Collector and Controller



Part G
3.7V LiPo Battery (Not included)



Part H
Three (3) 20cm Grove Cables



Parts you need to buy separately from the kit

- 3.7V LiPo Battery
- Raspberry Pi
- Compatible SDR Software Defined Radio (For example: https://hpjhlytllzrwf4qn-24552113.shopifypreview.com/products/software-defined-radio-sdr-and-antenna)
- 16GB SD Card (unless you bought the SD Card from SwitchDoc Labs)
- Power Supply for the Raspberry Pi

How to select a LiPo Battery

The WeatherSense AfterShock sensor requires a 3.7V LiPo battery.

How large of a LiPo battery you need depends on how much sun you get and how often you have cloudy weather. Generally, we would recommend a 6600mAh battery such as https://www.adafruit.com/product/353. Adafruit has a great selection and you can find good ones on SparkFun.com too.

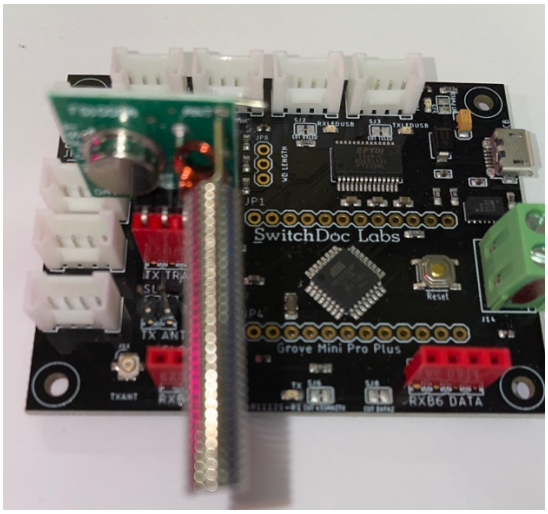
WARNING: if you get them off of Amazon, check the wiring. Most of them are wired backwards. Here's a great website showing the problem and how to rewire the batteries if you wish:

https://docs.particle.io/tutorials/learn-more/batteries/

Remember you are putting together the WeatherSense AfterShock to do testing and debugging. You will need to disassemble the unit and follow the assembly instructions in the WeatherSense AfterShock WeatherProofing and System Testing Manual.

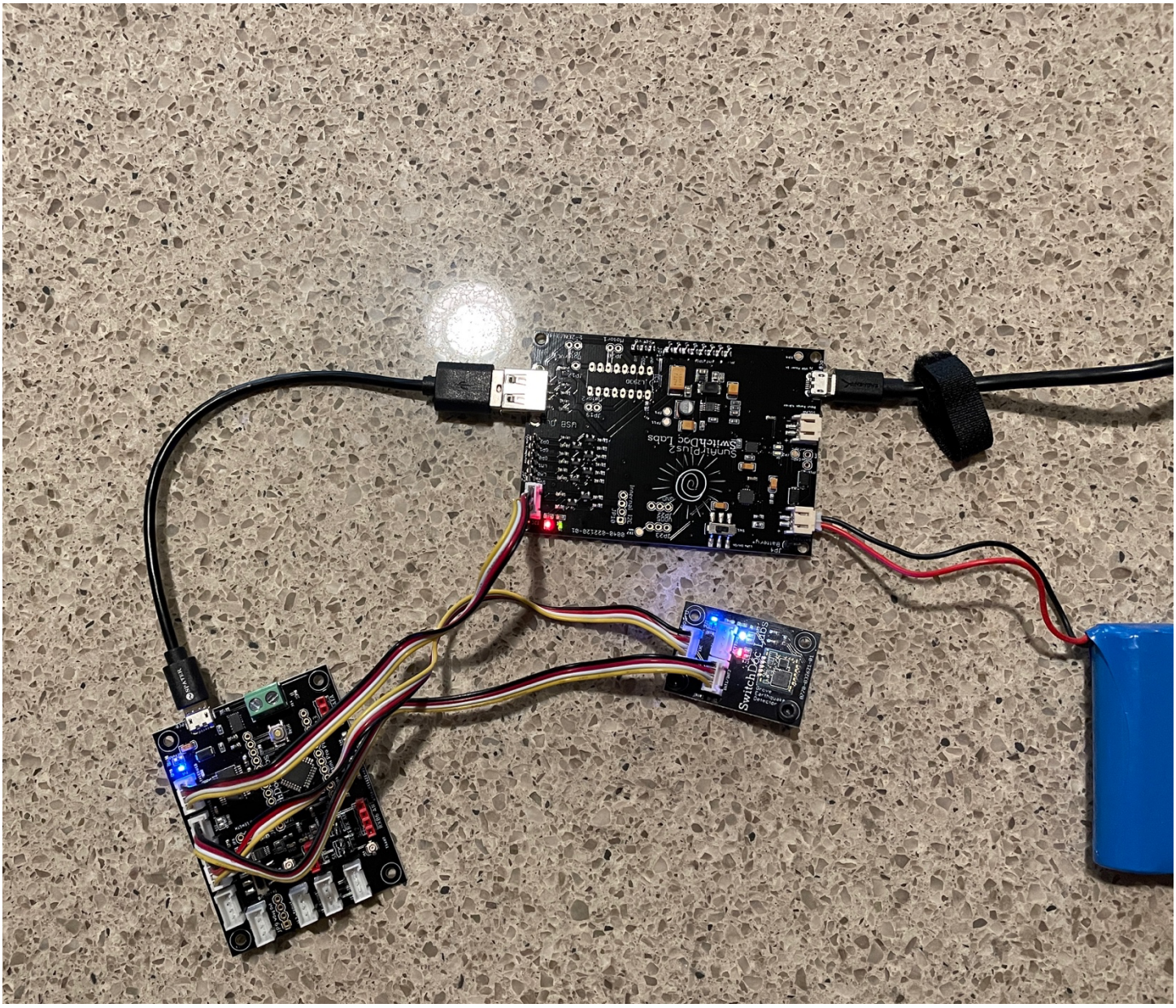
Step 1: Lay out all parts on a flat non-conductive (☺) surface.

Step 2: Plug in the 433MHz Transmitter (Part C) into the MiniProPlus (Part A) three pin female header market TX Tran. Make sure it is oriented as shown in the picture. Make sure the pins are slightly bent out as in the second picture. This makes sure that the pins are connected in the header. Bend pin 1 and pin 3 slightly out.

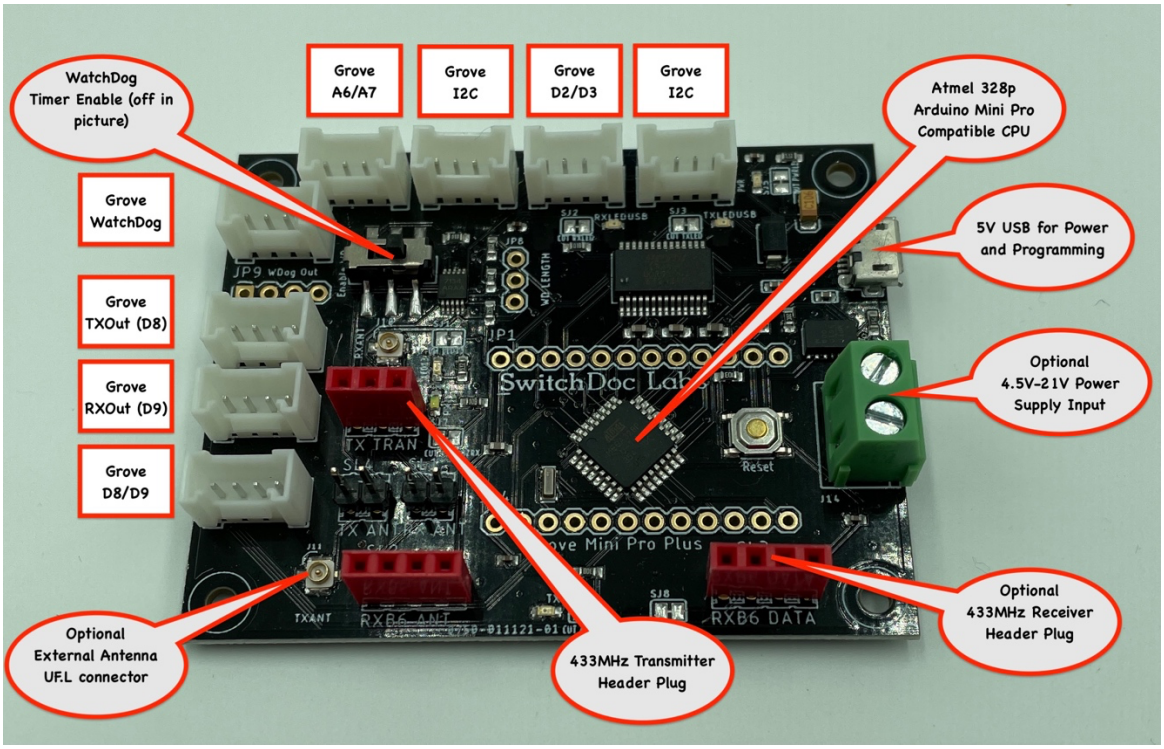


Step 3: Take a 20cm grove cable (Part I) and plug it into the grove connector on the Mini Pro Plus (Part A) marked J5-I2C and the other end into the I2C grove connector on the AfterShock Grove connector marked I2C (Part D). A green dot on the bottom means that the Mini Pro Plus has been programmed as an AfterShock Sensor. A white dot means it has been programmed as an Air Quality Sensor, while a red dot means it has been programmed as a lightning sensor.

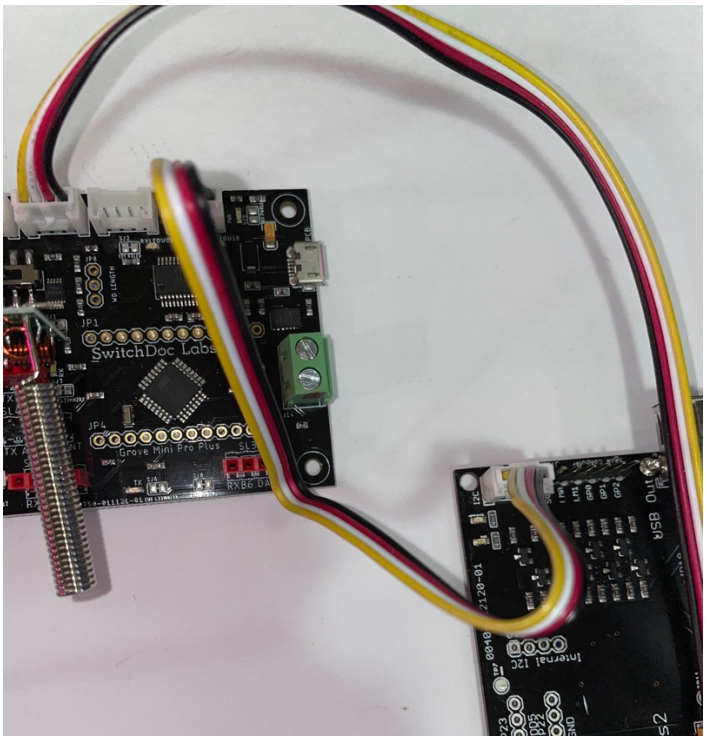
Step 4: Take a 20cm grove cable (Part I) and plug it into the grove connector on the Mini Pro Plus (Part A) marked J8-D2/D3 and the other end of the cable is plugged into the grove connector marked INT1/INT2 on the AfterShock board.



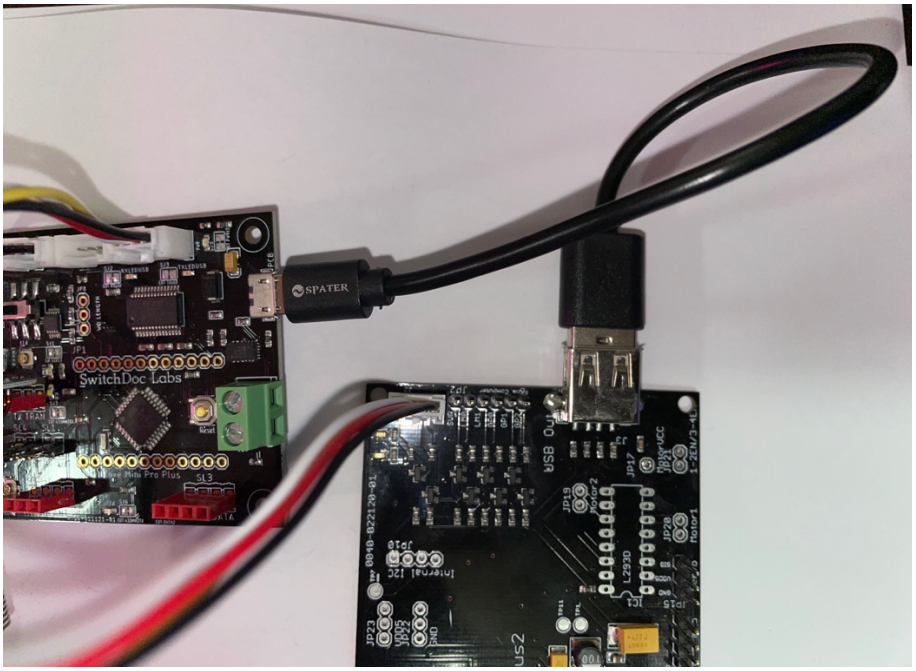
Step 5: Push the WatchDog Time Enable Switch to the Left (enable) on the Mini Pro Plus board (Part A).


WatchDog Timer Enable (off in picture)
Grove A6/A7
Grove I2C
Grove D2/D3
Grove I2C
Atmel 328p Arduino Mini Pro Compatible CPU
Grove WatchDog
5V USB for Power and Programming
Grove TXOut (D8)
Optional 4.5V-21V Power Supply Input
Grove RXOut (D9)
Grove D8/D9
Optional External Antenna UF.L connector
433MHz Transmitter Header Plug
Optional 433MHz Receiver Header Plug
SwitchDoc Lab
RXB6 DATA
RXB6 ANT
TX TRAN
Reset

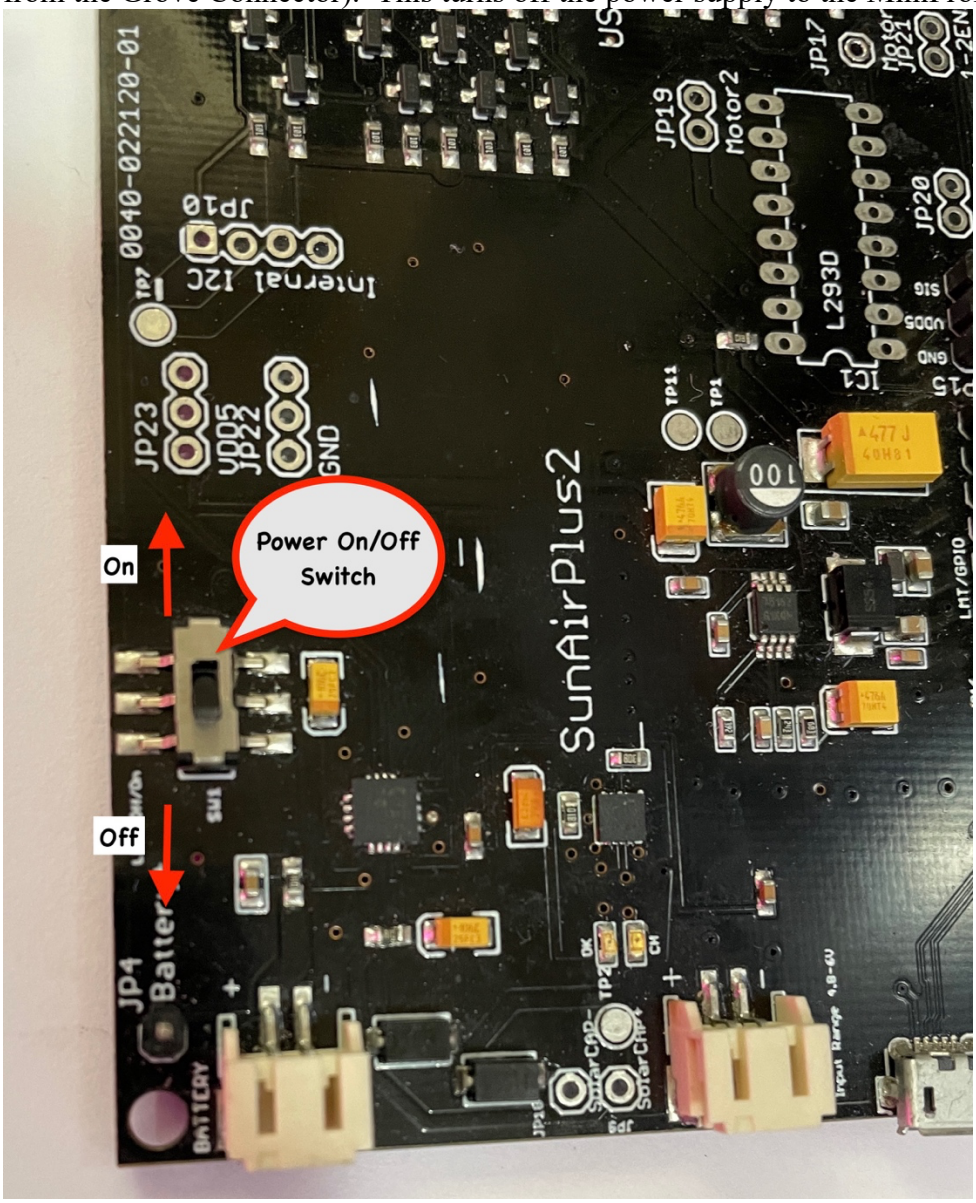
Step 6: Take an additional grove cable (Part I) and plug it in into the Mini Pro Plus (Part A) grove connector marked J7-I2C and the other end into the grove connector on the SunAirPlus2 board (Part F).



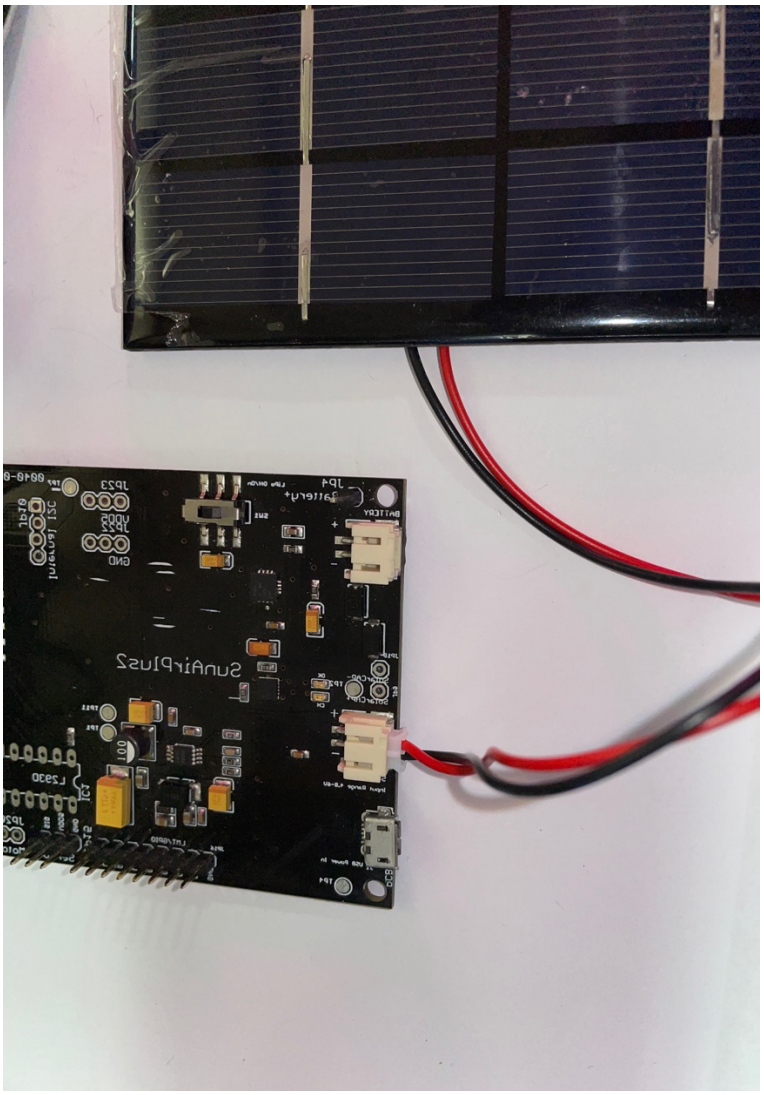
Step 7: take the USB micro to Type A cable (Part B) and plug it into the USB micro connector on the MiniProPlus board (Part A) and the other end into the TypeA connector on the SunAirPlus2 Board (Part F).



Step 8: Make sure the switch on the SunAirPlus2 board (Part F) is pushed toward the Battery Connector (away from the Grove Connector). This turns off the power supply to the MiniProPlus (Part A)

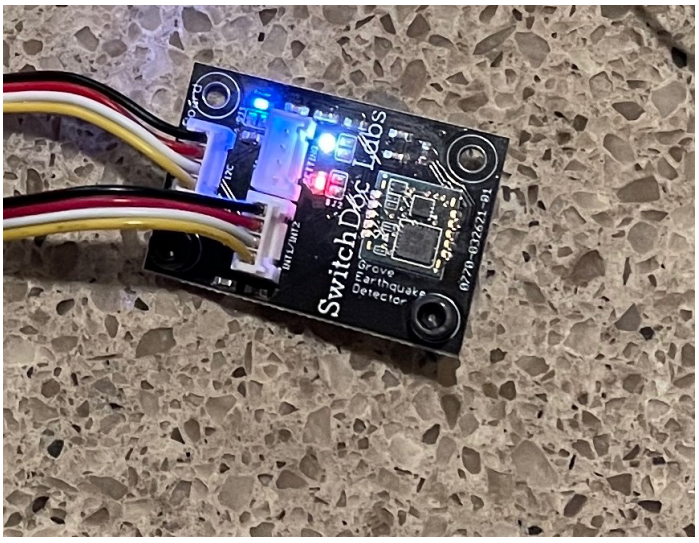


Step 9: Plug the Solar Panel (Part E) JST-2 connector into the SunAirPlus2 (Part F) board JST-2 plug marked SOLAR (next to the USB Micro connector on SunAirPlus2 (Part F)). You can take the protective plastic off of the solar panel (Part E) at this time.



Step 10: Plug your LiPo Battery (Part G – not included) into the JST-2 connector marked Battery (near the power switch) on the SunAirPlus2 board (Part F).

Step 11: Push the switch on the SunAirPlus2 (Part F) board away from the battery connector. You should see LED lights come on on both boards and see the white and Red Interrupt LED lights (and then go off after a while) on the AfterShock Board. If the white and red LEDs on the AfterShock (Part D) does not go out, you have plugged your 20cm Grove cable into the wrong plugs on the AfterShock Board (Part D) or on the MiniProPlus (Part A). Go back to Step 3.



Step 12: On powerup, after initialization, the WeatherSense AfterShock will send a message. You will see the LED market TX flash on the MiniProPlus (Part A). A message has been sent and should have been received by your Raspberry Pi software.

When you see the Red and White LEDs go off on the AfterShock board, tilt or lightly drop the AfterShock board and you will see the lights turn on again showing the board is evaluating a potential earthquake. It takes about 2 minutes for the board to complete the analysis and send a message out to your Raspberry Pi.

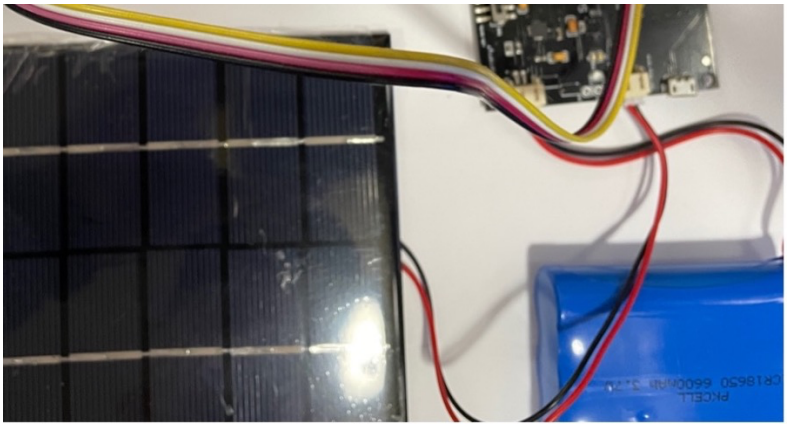
You can also connect a micro USB cable to the SunAirPlus2 board to your computer to watch the serial port – lots of data printing there. Check out the WeatherSense Advanced Programming Manual to see how to do this with the Arduino IDE.

If there is an earthquake or shaking event, the AfterShock board will send a message after about 2 minutes.

Step 13: The AfterShock board will send a message every ~60 minutes giving the latest AfterShock readings and all the solar power data. You can see the graphs on the dash_app once you have started the WeatherSense Software (see below).

Step 14: Turn off the SunAirPlus2 (Part F) board by pushing the switch towards the battery connector

You have now completed initial assembly. Now on to testing.



Testing WeatherSense AfterShock

If you have the SwitchDoc Labs SD Card, you can proceed to Step 4.

Step 1: Install the SDL modified version of rtl_433.

In a terminal window on your Pi at /home/pi type:

```
git clone https://github.com/switchdoclabs/rtl_433.git
```

Then compile it on the Raspberry Pi:

```
cd rtl_433/
mkdir build
cd build
cmake ..
make
make install
```

Step 2: Install the WeatherSense Software using these commands:

```
cd
git clone https://github.com/switchdoclabs/SDL_Pi_WeatherSense.git
```

Step 3: Add needed python modules to your system (a list will be provided on forum.switchdoc.com in the near future). If you want to jump into it, run

```
sudo python3 WeatherSenseMonitor.py
```

And start adding in the missing libraries as they come up as missing imports, otherwise do at least the following:

Installing mariadb:
https://pimylifeup.com/raspberry-pi-mysql/

```
sudo apt-get install python3-dev libmysqlclient-dev
```

```
sudo pip3 install mysqlclient
```

Next:

```
sudo -u root -p < WeatherSenseWireless.sql
```

Step 4: Note the IP Address of your Raspberry Pi. Type in the following commands to get your IP address:

```
hostname -I
```

You will get something like this:

```
pi@SwitchDocLabs:~/SDL_Pi_WeatherSense$ hostname -I
192.168.1.44
```

Step 5: Update WeatherSense to the latest version. Type the following commands into a terminal window.

```
cd
cd SDL_Pi_WeatherSense
git pull
```

You will see something like this:

```
pi@SwitchDocLabs:~/SDL_Pi_WeatherSense$ git pull
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 12 (delta 10), reused 12 (delta 10), pack-reused 0
Unpacking objects: 100% (12/12), done.
From https://github.com/switchdoclabs/SDL_Pi_WeatherSense
   f2193a0..c98a45c  master     -> origin/master
Updating f2193a0..c98a45c
Fast-forward
 README.md              |  2 ++
 SkyCamera.py           |  2 +-
 WeatherSenseMonitor.py         |  2 +-
 pclogging.py           | 12 +++++++-----
 state.py               |  3 +++
 testWirelessSensors.py | 18 ++++++++++----------
 6 files changed, 22 insertions(+), 17 deletions(-)
```

Or, if your software is up to date:

```
pi@SwitchDocLabs:~/SDL_Pi_WeatherSense$ git pull
Already up to date.
```

Step 6: First we will test the reception of the wireless weather sensors. Test your SDR and WeatherSense installation as follows. Note you must have completed the WeatherSense AfterShock assembly.

```
cd /home/pi/SDL_Pi_WeatherSense

sudo python3 WeatherSenseMonitor.py
```

Now start your AfterShock sensor by turning the SunAirPlus2 switch to on. You will then see something similar to this on your terminal window:

```
pi@SwitchDocLabs:~/SDL_Pi_WeatherSense $ sudo python3 WeatherSenseMonitor.py
Pending jobs:
    readSensors (trigger: date[2021-02-27 12:13:55 PST], pending)
-----------------

Scheduled Jobs
######
-----------------
Read Wireless Sensors
######
Jobstore default:
    No scheduled jobs
-----------------
starting 433MHz scanning
######
processing AfterShock Data
This is the raw data: {"time" : "2021-06-03 14:42:36", "model" : "SwitchDoc Labs AfterShock", "len"
: 53, "messageid" : 1280, "deviceid" : 1, "protocolversion" : 1, "softwareversion" : 2,
"weathersenseprotocol" : 18, "eqcount" : 30, "finaleq_si" : 0.061, "finaleq_pga" : 2.220,
"instanteq_si" : 0.032, "instanteq_pga" : 0.870, "loadvoltage" : 5.008, "batteryvoltage" : 4.192,
"batterycurrent" : 48.400, "loadcurrent" : 26.400, "solarpanelvoltage" : 6.272, "solarpanelcurrent"
: 4.800, "auxa" : 3, "solarpresent" : 1, "aftershockpresent" : 1, "keepalivemessage" : 0,
"lowbattery" : 0, "mic" : "CRC"}
```

What does the JSON from the WeatherSense AfterShock mean?

The WeatherSenseMonitor.py python3 program reads in the 433MHz signals into the Raspberry Pi, decodes them and formats the results as a JSON packet that will be provided to the WeatherSenseMonitoring software for processing and storage in the database. More about the SDL_Pi_WeatherSense software in the WeatherSense Software Manual.

If you have other WeatherSense Sensors (such as SkyWeather2, WeatherRack2, F016TH Indoor sensors, etc), you will see them show up in the list too.


```
This is the raw data: {"time" : "2021-06-03 14:42:36", "model" : "SwitchDoc Labs AfterShock", "len"
: 53, "messageid" : 1280, "deviceid" : 1, "protocolversion" : 1, "softwareversion" : 2,
"weathersenseprotocol" : 18, "eqcount" : 30, "finaleq_si" : 0.061, "finaleq_pga" : 2.220,
"instanteq_si" : 0.032, "instanteq_pga" : 0.870, "loadvoltage" : 5.008, "batteryvoltage" : 4.192,
"batterycurrent" : 48.400, "loadcurrent" : 26.400, "solarpanelvoltage" : 6.272, "solarpanelcurrent"
: 4.800, "auxa" : 3, "solarpresent" : 1, "aftershockpresent" : 1, "keepalivemessage" : 0,
"lowbattery" : 0, "mic" : "CRC"}
```


The values and units of the are:

- timestamp: When the packet has been received
- model: SwitchDoc Labs AfterShock
- len: Number of bytes in the packet
- messageid: ID of message. Not reset by bootup
- deviceid: 1 – If you have more than one AfterShock sensor, you can change this ID by recompiling the software
- protocolversion: Current protocol of theAfterShock sensor
- softwareversion: This shows the current version of the software on the Mini Pro Plus
- weathersenseprotocol: 18 for WeatherSense AfterShock
- eqcount: Count of earthquakes since bootup
- finaleq_si: SI reading as evaluated by the Earthquake Algorithm (m/sec).
- finaleq_pga: PGA (Peak Acceleration Readionmg) reading during evaluation by the Earthquake Algorithm (m/sec**2).

- instanteq_si: Instantaneous SI reading during evaluation of the Earthquake Algorithm (m/sec).
- instanteq_pga: PGA (Peak Acceleration Reading) reading during the evaluation of the Earthquake Algorithm (m/sec**2).

The solar data collected by the device is:

- loadvoltage: Voltage supplied to the computer (V)
- batteryvoltage: Voltage of the LiPo Battery (V)
- batterycurrent: Current being supplied by the battery (mA)
- loadcurrent: Current being supplied to the computer (mA)
- solarpanelvoltage: Voltage of the Solar Panel (V)
- solarpanelcurrent: Current being supplied by the Solar Panel (mA)

The AuxA variable contains state information about what sensors the WeatherSense AfterShock are connected to and if the sensor is in low voltage mode.

Each bit of the lower four bits of the AuxA variable (0CAB) are coded as:

AuxA has state information coded in the bottom byte of the 32 bit AuxA as 000EDCBA

// A = 1, AfterShock Present, 0 not present
// B = 1, IN3221 (Solar) Present, 0 not present (SunAirPlus)
// C = 1, Low battery, Chip shut off (too many false alarms in low voltage mode)
// D = 1, I'm alive message
// E = 1, EQ evaluation started

You now have completed testing of your WeatherSense AfterShock sensor.
Print your 3D Prints and move on over to the WeatherSense AfterShockWeatherproofing manual. Then take a look at the WeatherSense software manual for the Raspberry Pi.

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.