Assembly & Operating Instructions

For

D1M-WX1 and D1S-WX1

Solar Powered IoT Weather Stations

Version 2.2
16 June 2017

IoT Kits©

Berger Engineering

W4KRL.com/iot-kits/

Copyrights and Licenses

License

Contact Berger Engineering, 5667 Stone Road, Suite 465, Centreville, VA 20120, sales@w4krl.com for commercial licensing opportunities.

Third Party Software Disclaimer

Any links to third-party software available in this document are provided "as is" without warranty of any kind, either expressed or implied and such software is to be used at your own risk.

The use of the third-party software links on this website is done at your own discretion and risk and with agreement that you will be solely responsible for any damage to your computer system or loss of data that results from such activities. You are solely responsible for adequate protection and backup of the data and equipment used in connection with any of the software linked to this website, and we will not be liable for any damages that you may suffer in connection with downloading, installing, using, modifying or distributing such software. No advice or information, whether oral or written, obtained by you from us or from this website shall create any warranty for the software.

Electrostatic Discharge Warning

ESD Warning

This product, like all microcontroller products, uses semiconductors that can be damaged by electrostatic discharge (ESD). When handling, care must be taken so that the devices are not damaged. Damage due to inappropriate handling is not covered by the warranty.

The following precautions must be taken:

- Do not open the protective conductive packaging until you have read the following, and are at an anti-static work station.
- Use a conductive wrist strap attached to a good earth ground.
- Use a soldering iron or station that is marked as ESD-safe.
- Always discharge yourself by touching a grounded bare metal surface or anti-static mat before picking up an ESD-sensitive electronic component.
- Use an anti-static mat to cover your work surface.

Contents

Terms

ADC	Analog to Digital Converter. Converts an analog signal, usually a voltage, into a digital signal.
APRS	Automatic Packet Reporting System developed by Bob Bruninga, WB4APR.
APRS-IS	APRS Internet Service. Permits licensed amateur radio operators to send data to the APRS system without an RF connection. Data is viewable by anyone on the Internet.
Breakout	A small printed circuit board that breaks out the connections of one or more components, usually surface mount devices, to connections suitable for prototyping and manual construction.
Bus	An electrical connection that allows power or communication between multiple devices such as sensors and microcontrollers.
dBm	A power ratio measured in decibels referenced to 1 milliwatt.
DC	Direct current.
DMM	Digital Multi-Meter.
ESD	Electrostatic Discharge.
Firmware	Software that is programmed into read-only memory.
Ground	A reference point in electrical circuits from which voltages are measured.
Header	A form of electrical connector consisting of one or more regularly spaced male pins.
I2C	Inter-Integrated Circuit. A serial bus widely used in microcontroller applications. Pronounced eye-squared-see.
IDE	Integrated Development Environment. A collection of related software development tools combined into one user interface.
IEEE	Institute of Electrical and Electronic Engineers.
IoT	Internet of Things. The interconnection of devices with Internet addresses that can send and receive data over a network. Pronounced eye-oh-tee.
LiPo	Lithium Polymer. A lithium-ion cell packaged in a flexible plastic case.
mA	Milliampere. A measure of electrical current equal to one-thousandth of an ampere.
mAh	Milliampere hour. A measure of the energy storage capacity of a cell or battery.
Microcontroller	A microcontroller is a computer in a single integrated circuit dedicated to executing one specific application. It contains memory, programmable input/output peripherals as well a processor.
NTP	Network Time Protocol. NTP provides accurate time information over a network.
PCB	Printed Circuit Board.
Pin	A connecting point of an electrical device. It may refer to the function of the connection (for example – CLOCK) or the physical connection itself (pin 4).

RF	Radio Frequency.
RTC	Real Time Clock. An RTC continues operation when power is not applied.
RSSI	Received Signal Strength Indicator. An arbitrary unit roughly equivalent to dBm.
Sensor	A device that detects or measures a physical property such as heat or light.
Shield	A printed circuit board that can be plugged into a microcontroller board such as an Arduino or D1 Mini that extends its capabilities.
SMD	Surface Mount Device. An electronic component that is soldered to pads on a PCB without the use of through hole leads.
SoC	A System on a Chip or System on Chip is an integrated circuit that integrates all components of a computer or other electronic system. It may contain digital, analog, mixed-signal, and often radio-frequency functions.
ThingSpeak	A free and open Internet of Things service.
Through-Hole	Through-hole, also spelled "thru-hole", refers to the mounting of electronic parts with leads on the components inserted into printed circuit board holes and soldered to pads on the opposite side.
V	Voltage.
Wi-Fi	Wi-Fi provides for wireless local area networking with devices based on IEEE 802.11 standards. Wi-Fi is a trademark of the Wi-Fi Alliance. Pronounced to rhyme with my pie. It is specified by IEEE Standard 802.11.

How to use this Instruction Manual

This is an intermediate level kit. You should be able to solder small components and have basic knowledge of operating a personal computer. You must have a wireless Internet connection (Wi-Fi) and be familiar with downloading software and installing programs. You do not need to be a C++ programmer as all programming operations are given "cookbook" style.

It is recommended that you check off each step as you go so that you can pause construction and restart later.

This manual contains assembly and operation instructions for two versions of the IoT Kits© Solar Powered Weather Station:

- **D1M-WX1** – A single board kit incorporating a lithium polymer cell charger, microcontroller, and sensors mounted on one PCB. See page 12 for detailed instructions.
- **D1S-WX1** – A "shield" version of the weather station kit with three stacked PCBs. The base shield is a lithium polymer cell charger. The middle shield is a microcontroller breakout board. The top shield mounts sensors. See page 19 for detailed instructions.

Both kits use the same Arduino-based firmware and have the same weather reporting functions. The major difference between the kits is the overall size of the project box.

The first set of steps install the Arduino IDE and the ESP8266 core on your PC. They are common to both kits. These steps will ensure that the D1 Mini microcontroller is operating and that you will be able to successfully load the weather station firmware after you complete the construction.

Do not start assembly of your kit until you have confirmed operation of the IDE and D1 Mini.

Tools Needed

If you've built kits before you probably have all these tools. These are some suggested tools for new kit builders. Most are inexpensive tools that you will use again if you build other kits.

1. Soldering iron or soldering station with good quality rosin-core solder. A 30-watt iron is satisfactory. It should have a conical tip. Look for an iron with a three-prong plug that grounds the tip for ESD safety. The X-Tronic Model #3020 is an excellent choice for a temperature controlled iron. Good choices are also available from Micro Center and Marlin P. Jones.
2. Small diagonal cutters such as Harbor Freight #40695.
3. Hot melt glue gun such as the Walmart Low-Temp Mini Glue Gun #557243715.
4. 1/8-in and 1/4-in (3mm & 6mm) drill bits. Harbor Freight #69470 handle & #61621 drill bits.
5. Phillips screwdriver. Size PH2 is a good choice.
6. Flush cutter to trim excess leads. A good choice is the Hakko CHP-170. Harbor Freight #90708 is also acceptable.
7. Micro USB programming cable. Any cable suitable for Android smart phones will work.
8. Needle-nose pliers. Useful for picking up parts and bending leads. Harbor Freight #40696.
9. Optional digital multimeter for voltage calibration. Harbor Freight #90899.
10. Optional magnifying lens to inspect the solder joints.

Outline of Assembly Steps

1. Read this Assembly Manual and view the YouTube videos.
2. Identify the parts, collect your tools, and find a good work location.
3. Install the Arduino IDE software and ESP8266 core. Test the D1 Mini.
4. Assemble the PCBs and install the breakout boards and other electronic parts.
5. Create a ThingSpeak account and program the D1 Mini with the weather station firmware.
6. Optional: If you are a licensed radio amateur, install the APRS firmware after you have the ThingSpeak firmware working.
7. Prepare the enclosure.
8. Install the lithium polymer cell, PCB, and solar cell in the enclosure.
9. Mount the station outdoors and monitor your data on ThingSpeak and APRS-IS.

YouTube Videos

Links to YouTube videos are on the D1M-WX1 Weather Station project web page.

- Soldering tutorial
- PCB Assembly
- Install the ESP8266 Core in the Arduino IDE
- Box Assembly

Install Arduino IDE and ESP8266 Software

We are going to take a somewhat unconventional approach and install the Arduino software on your PC before starting construction. This will ensure that the D1 Mini microcontroller is operating and that you will be able to successfully load the weather station firmware after you complete the construction.

The Arduino software is an Integrated Development Environment (IDE) that makes it easy to create, edit, compile, and upload firmware to Arduino compatible microcontrollers. A dedicated group of volunteers have written an add-in for the IDE that extends its capabilities to the ESP8266 System on a Chip as used on the D1 Mini in this kit. Almost all the extensive Arduino library files can be used with the ESP8266.

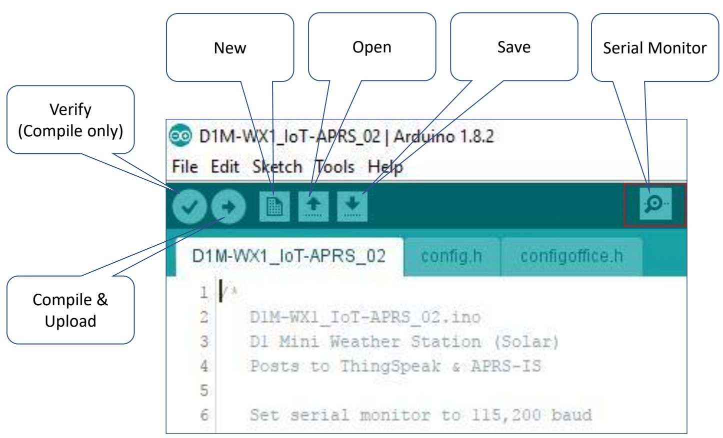
These steps install the Arduino IDE and the ESP8266 core onto your personal computer.

1. Download the Arduino IDE from https://www.arduino.cc/en/Main/Software. Be sure to select the version appropriate to your PC. Detailed instructions for Windows PCs are provided at https://www.arduino.cc/en/Guide/Windows.
2. The installation adds a desktop icon for Arduino. Click on it to run the IDE.

The following steps 3 to 6 are illustrated in a YouTube video.

3. Open menu item File > Preferences.
4. Enter `http://arduino.esp8266.com/stable/package_esp8266com_index.json` into the Additional Board Manager URLs field. You may copy from this document and paste it in the text box to avoid making a typing error.
5. Check the box for `Display line numbers`. Click `OK` to close the Preferences dialog.

6. Open `Boards Manager` from `Tools > Board` menu and enter "`esp8266`" in the search box. Click on "`online help`". Click on the Install button to install. Close the Boards Manager dialog when finished.
7. Click on IDE menu `File > Examples`. A dropdown box will open with a long list of examples. Scroll to the section "`Examples for WeMos D1 R2 & mini`" and select `Esp8266 > Blink` to open a simple blink firmware example.
8. Use IDE menu `Tools > Board` and select **`WeMos D1 R2 & mini`** in the section labelled ESP8266 Modules.
9. Plug in the Micro USB cable to your computer and the D1 Mini. While it may seem odd to have the D1 Mini dangling from the end of a cable, this is quite safe – the voltages are very low. Your computer should install a COM port. Select the new COM port from menu item `Tools > Port:` You can use Windows Device Manager to find the port number.
10. Click on the IDE upload button to compile and upload the firmware to the D1 Mini. This is the blink.ino sketch that you opened in step 7. A small LED on the ESP8266 chip will rapidly flash during the upload. When the upload is finished the sketch will immediately start running. It will make the LED flash one second on and two seconds off.
11. If you have any difficulty with the above steps, the latest version of these instructions is located at https://github.com/esp8266/Arduino. Use the instructions for "Installing with Boards Manager." There are nicely illustrated instructions at http://randomnerdtutorials.com/how-to-install-esp8266-board-arduino-ide/



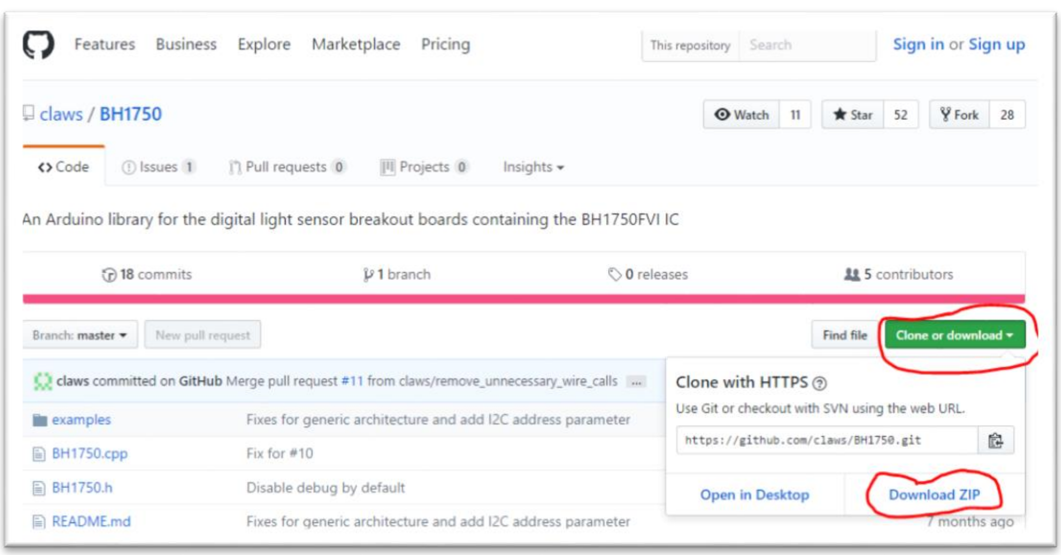
Download the D1M-WX1 Weather Station Firmware

These steps will download the weather station firmware to your PC. You will compile and upload the Arduino software in later steps after the kit is assembled.

1. Download the weather station firmware:
    1. Navigate you PC web browser to http://w4krl.com/d1m-wx1-solar-powered-weather-station/. Scroll to Weather Station Firmware.
    2. Click on D1M-WX1 Weather Station. This will download the D1M-WX1_Weather_Station.zip file to your download folder.
    3. Double-click on the ZIP file and choose Extract all. Click Extract. The files will appear in a new folder called D1M-WX1_Weather_Station.

4. Open the D1M-WX1_Weather_Station folder. Hold down your Control key and click on each of the three folders: D1M-WX1_Calibration, D1M-WX1_IoT, and D1M-WX1_IoT-APRS. Right-click to copy these three folders.
5. Locate your Arduino sketch folder. This is usually \Documents\Arduino. Left-click to Paste.
6. In the IDE, click menu item `File > Open…` You should see the three folders you just copied to your Arduino directory.

2. Additional library software provided by others is needed for the sensors. These instructions are all you need to do. More detail is provided in https://www.arduino.cc/en/guide/libraries.
   1. Install the library for the BME280 sensor:
      1) In the IDE, open menu item `Sketch > Include Library > Manage Libraries…`
      2) Type `BME280` in the Filter your search… box. Multiple libraries will appear.
      3) Install the library for BME280 by Tyler Glenn.
   2. Install the library for the BH1750 sensor:
      1) Open https://github.com/claws/BH1750 in your browser.
      2) Click on the ***Clone or download*** button and choose Download ZIP. This will download BH1750-master.zip to your PC's Downloads folder.
      3) In the IDE, open menu item `Sketch > Include Library > Add .ZIP Library…` Navigate to your Downloads folder and select the BH1750 file.



Assembly Instructions for D1M-WX1 Single Board Weather Station

After completing the Arduino IDE installation on page 9, you can be confident that you will be able to program the microcontroller and that it operates correctly. You are now ready to assemble the printed circuit board. Find a good location to build the kit, collect your tools, and identify the parts. This is also a good time to read the ESD notice on page 3.

Assemble the Printed Circuit Board

1. Mount spacers to the PCB. Insert the screws from the top of the PCB and make the spacers finger tight in four places. The spacers permit using the PCB as a stable fixture for installing the breakout boards and other parts. The next steps install the four breakout boards from left to right.
2. TP-4056 Charger Assembly and Installation
    1. Cut four individual pins from the male header stock using diagonal or end cutters. Cut at the divisions between the pins. They do not take much force to cut. Be neat!
    2. Place the four pins on the PCB with the long tails through the PCB.
    3. Carefully place the TP-4056 breakout board on the pins. Observe proper orientation of the pins, that is, BAT+ to BAT+, etc.
    4. Solder the pins to the TP4056 (4 places).
    5. Turn over the PCB. Keep the TP4056 flush on the PCB and solder the four TP4056 pins to the PCB.
    6. Trim the pins with the diagonal or flush cutters.
    7. Return the board to the face up position.
3. D1 Mini Assembly and Installation
    1. Place two 8-pin male headers on the PCB at the D1 Mini outline with the SHORT tails of the pins through the PCB.
    2. Insert Micro USB cable into the socket on the D1 Mini. This will ensure enough clearance for the cable connector in the future.

    3. Place the D1 Mini on the LONG header pins with the ESP8266 side up. Try to keep it parallel to the PCB. Soldering just one or two pins on the D1 Mini will allow you to adjust the board.
    4. Solder pins on the four corners off the D1 Mini being careful to keep the D1 Mini square with the PCB by holding it against the board with a finger. The corner pins are RST, TX, 5V, and 3V3. Make sure you can insert and remove the Micro USB connector with no difficulty.
    5. Solder the remaining pins on the D1 Mini.
    6. Turn over the board, keeping the D1 Mini flush with the PCB, and solder all 16 D1 Mini pins to the PCB.
    7. Trim the pins on the top side of the D1 Mini.
4. BME280 Assembly and Installation
    1. Place the 4-pin header on the PCB with the LONG tails through the PCB.
    2. Carefully place the BME280 breakout board on the pins.
    3. Solder the VIN pin to the BME280. Place the tip of your iron on the short side of the board to raise the board away from the PCB. Try to keep the board parallel to the PCB.
    4. Solder the remaining pins (3 places) to the BME280.

    5. Turn over the PCB and solder the four BME280 pins to the PCB.
    6. Trim the pins.
5. BH1750 Assembly and Installation
    1. Place the 5-pin header on the PCB with the LONG tails through the PCB.
    2. Carefully place the BH1750 on the pins.
    3. Solder the VCC pin to the BH1750. Place the tip of your iron on the short side of the board to raise the board away from the PCB. Try to keep the board parallel to the PCB.
    4. Solder the remaining pins (4 places) to the BH1750.
    5. Turn over the PCB and solder the five BH1750 pins to the PCB.
    6. Trim the pins.
6. I2C Header Assembly and Installation
    1. Cut a stick of four (4) pins from the male header stock.
    2. Place the SHORT tails of the pins through the PCB at the I2C location.
    3. Turn over the PCB and solder the four pins to the PCB.
7. LOCAL Header Assembly and Installation
    1. Cut a stick of two (2) pins from the male header stock.
    2. Place the SHORT tails of the pins through the PCB at the LOCAL location.
    3. Turn over the PCB and solder the two pins to the PCB.
    4. Place the shunt jumper on the LOCAL header pins.
8. Resistor Installation
    1. Place the 180 kilohm resistor in the resistor location.
    2. Turn over the PCB and solder the two (2) resistor leads to the PCB.
    3. Trim the resistor leads.
9. Slide Switch Installation
    1. Place one Slide Switch in the PROG-RUN location.
    2. Turn over the PCB and solder the three (3) Slide Switch pins to the PCB.
    3. Trim the pins.
    4. Repeat the installation steps for the POWER Slide Switch.
10. Screw terminal installation
    1. Carefully insert the 2-point screw terminal into the PCB at the CELL location. Make sure the wire openings point to the outside of the PCB.
    2. Solder the terminals (2) on the underside of the board.
11. XH Connector Installation
    1. Carefully insert the 2-point XH connector in the PANEL location. Make sure the notch points to the outside of the PCB.
    2. Solder the two pins on the underside of the board.
12. Inspect the completed PCB
    1. Check that all parts are in the correct orientation.
    2. Check that all solder joints are shiny and smoothly connect the device pin with the PCB solder pad.
    3. Check that there are no bent pins or shorts between PCB solder pads or on the breakout boards.
    4. Check that the shunt jumper is in place on the LOCAL header pins.

Calibrate Voltage Sensor (Optional)

While the voltage calibration procedure is optional, it is recommended. The kit is supplied with quality components; however, normal parts variations may account for as much as ±10% deviation from accurate voltage measurement.

You will need a digital multimeter (DMM) for the calibration steps.

1. Connect the D1 Mini to your PC with the Micro USB cable. Make sure that the cable is connected to the D1 Mini and NOT the TP4056 charger.
2. Verify that the PROG/RUN switch on the PCB is in the PROG position.
3. Verify that the power switch on the PCB is in the ON position.
4. Verify that IDE menu `Tools > Port:` is showing the correct COM port.
5. Verify that IDE menu `Tools > Board:` is showing "WeMos D1 R2 and mini"
6. Use menu `File > Open` to open D1M-WX1_Calibration.ino.
7. Click on the IDE upload icon or use menu Sketch > Upload. The IDE will compile the sketch and begin to upload it to the D1 Mini. A small LED next to the ESP8266 with rapidly flash during the upload.
8. The D1 Mini will start running the firmware as soon as the upload is complete. The small LED should repeatedly blink one second on and one second off.

9. Voltage Calibration: Click on the IDE Serial Monitor icon or use menu `Tools > Serial Monitor` to open the serial monitor window. Change the baud rate in the dropdown box in the lower right of the window to 115200 baud. You will see voltage measurements appear every two seconds. This is the voltage measured by the ESP8266 analog to digital converter (ADC). It is normal for the voltage to change slightly between measurements. Write down a voltage that appears to be an average of the measurements.
10. Use your DMM to measure the voltage between the screws on the CELL terminal block. Write down the average voltage.
11. The ADC and DMM voltages will be used to calibrate the weather station software in a later step.

Configure & Upload the D1M-WX1_IoT.ino Firmware

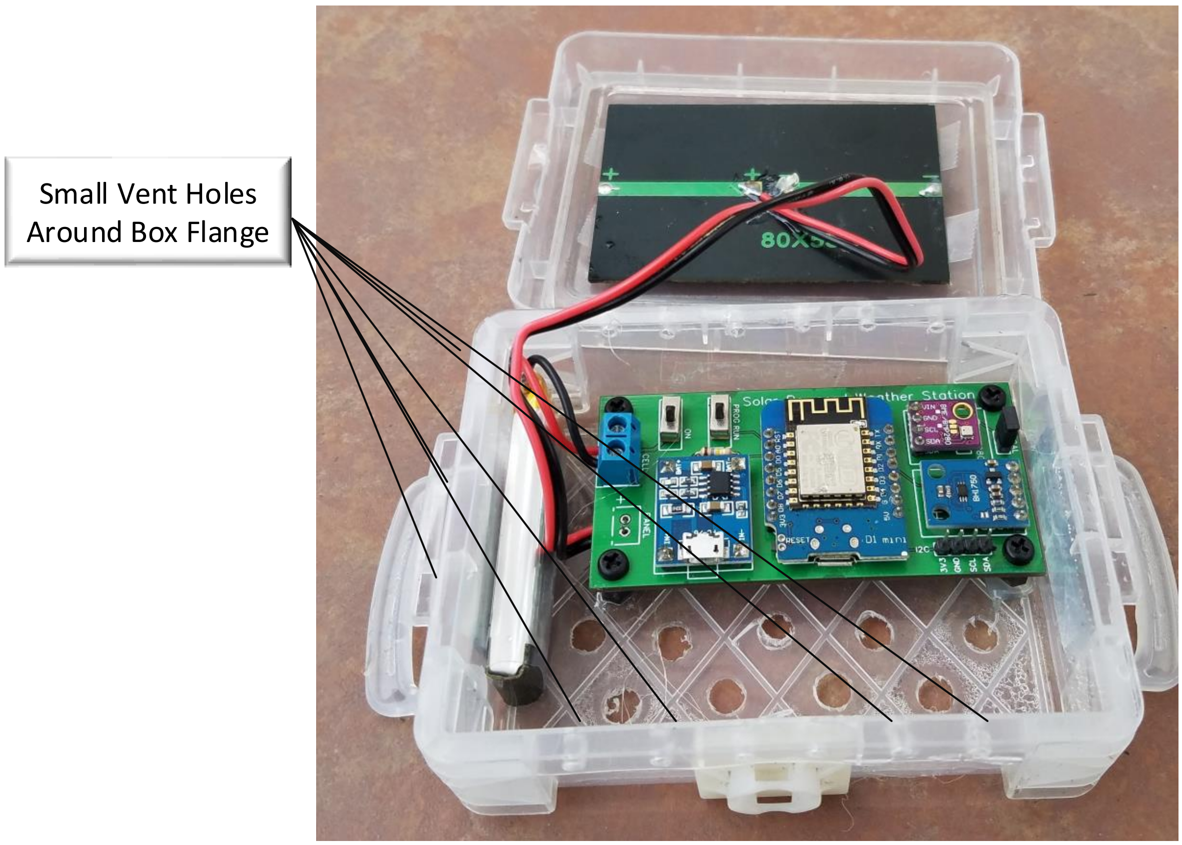
Follow the steps in APPENDIX A – ThingSpeak IoT Firmware to configure the ThingSpeak IoT firmware. You can view the weather station data on your ThingSpeak account and on the IDE serial monitor.

Prepare the enclosure

1. Refer to Figure 1.
2. Drill 1/8-inch (3 mm) holes in the box sides, at least twelve places. This will provide ventilation for the sensors. Drill from the outside through both layers of the box flange.
3. Place the PCB in the box as shown in Figure 1. Mark the location of the mounting spacers on the outside of the box with a permanent marker.
4. This step must be done quickly: Put a dab of hot melt glue in the bottom of the box at each location (4) marked. Insert the PCB with the spacers in the glue spots making sure the PCB box is square with the box. Hold until the glue is set - about 30-seconds.
5. Unscrew the PCB from the box.
6. Drill ¼-inch (6 mm) large ventilation holes in the bottom of the box. Make sure the holes do not interfere with the mounting spacers.

7. Place a bead of hot melt glue around the front surface of the solar panel. Press the panel onto the underside of the box lid. Allow the glue to set.
8. Make sure the box is clean and dry at the mounting location shown for the tie-wrap mount. Remove the paper backing from the tie-wrap mount and press against the box as shown. Hold for 30 seconds.

Figure 1 – Box Locations



Final assembly

1. Lithium polymer cell assembly and Installation
    1. Turn the Power switch to the **OFF** position.
    2. If the cell leads are provided with protective tape, carefully remove the tape from the cell leads.

WARNING: KEEP LEADS FROM TOUCHING EACH OTHER!

    3. If the cell is provided with a connector, carefully cut each lead from the connector as close to the connector as possible. Cut the leads **ONE BY ONE.**

WARNING: DO NOT CUT ACROSS BOTH WIRES AT THE SAME TIME!

    4. Strip about ¼-inch (6 mm) from each cell lead.
    5. Separately twist the strands in each lead and tin the leads by applying a minimum amount of solder.

    6. Insert the RED (positive) lead of the cell into the positive (+) screw terminal. Tighten the screw to hold the wire in place. **DO NOT OVERTIGHTEN.**
    7. Insert the BLACK (negative) wire in the other screw terminal. Tighten the screw to hold the wire in place. **DO NOT OVERTIGHTEN.**
    8. Gently tug the wires to check that they are secure in the screw terminals.
2. Charge the LiPo cell by plugging your Micro USB connector into the socket on the TP4056 board. Make sure the PCB power switch is **Off**. The orange light indicates charge. The blue light indicates full charge.
3. Solar Panel connection
    1. Locate the free end of the two-conductor wire attached to the XH plug. Note that the wires are marked with different patterns to identify each conductor. These markings are different for each pair of wires.

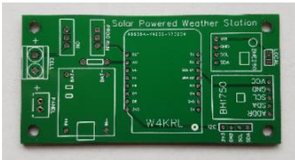
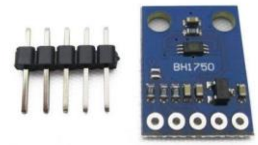
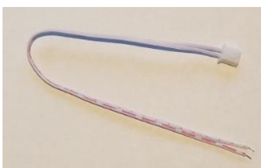
    WARNING: IT IS VERY IMPORTANT TO PROPERLY IDENTIFY THE POLARITY OF EACH WIRE!

    2. Place the plug into the connector and note the pattern on the wire that is closest to the + (positive) marking on the PCB. Write down the pattern on the positive wire. The other wire is the – (negative) wire. The next steps are easier if you remove the plug from the connector.
    3. Strip insulation from about ¼-inch (6 mm) of each lead. Separately twist the strands in each lead and tin the leads by applying a small amount of solder.
    4. Carefully solder the positive lead (+) to the positive (**+**) solder pad on the rear of the solar panel near its middle. DO NOT OVERHEAT THE SOLAR PANEL.
    5. Carefully solder the negative lead (-) to the negative (**-**) solder pad on the rear of the solar panel. This is the other pad near the center of the panel.
    6. Apply a dab of hot melt glue to the wire to hold it in place against the rear of the solar panel.
    7. This step will glue the solar panel to the underside of the box lid. The panel and lead will be contained within the box. Run a bead of hot melt glue around the top surface of the solar panel and press it into the underside of the lid. Hold until the glue sets.
    8. Plug the solar panel lead into the PANEL connector on the PCB.
    9. Reattach the PCB to the box spacers.
    10. Put the lid on the box with the wire routed inside the box.

Congratulations! You have completed the assembly of your Internet of Things weather station.

The next step is to mount the station. See page 25.

Illustrated Parts List for D1M-WX1 Single Board Kit

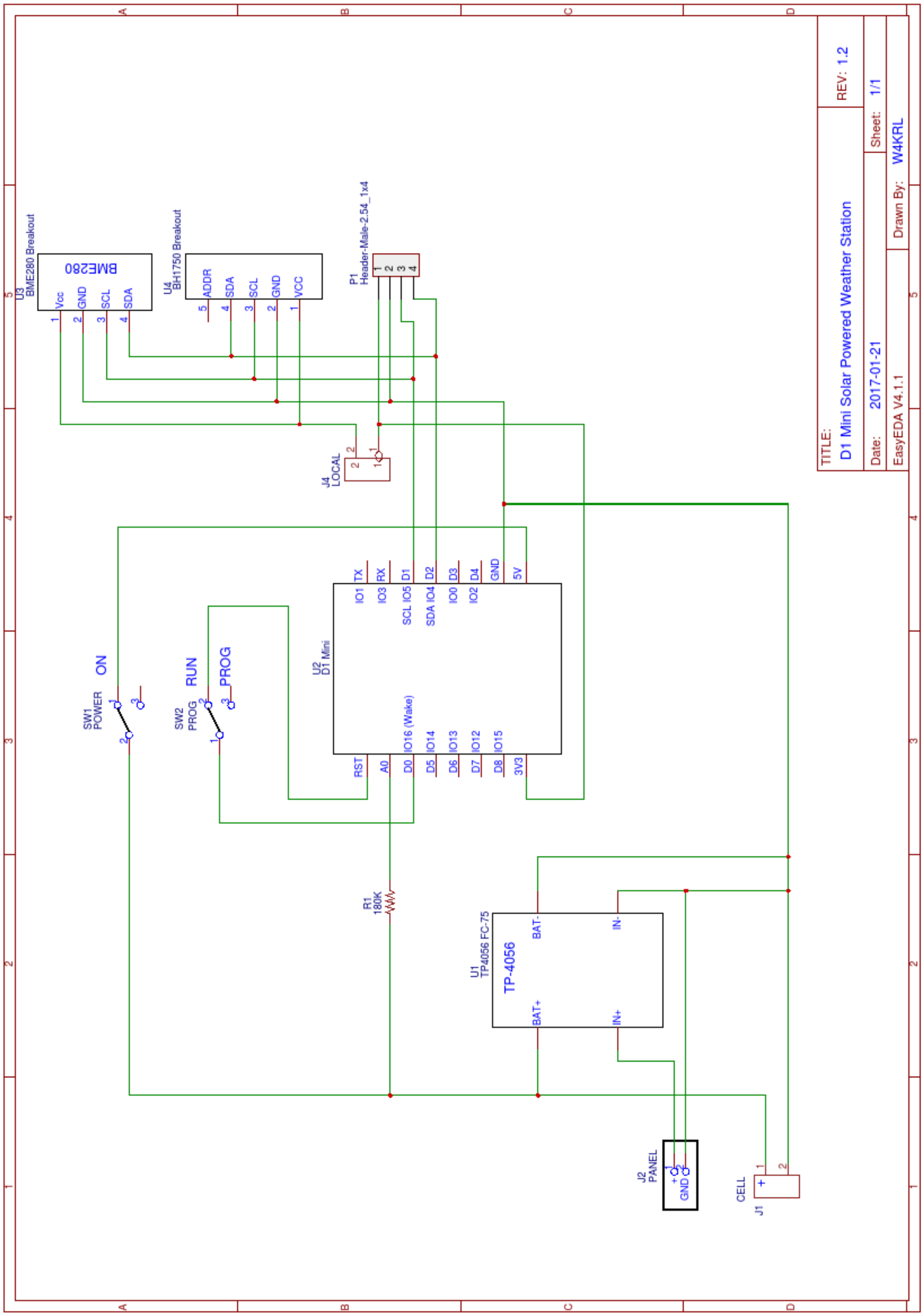
 Printed Circuit Board (PCB)	 Enclosure	 Solar Panel
 D1 Mini	 BME280 Breakout & Header	 BH1750 Breakout & Header
 XH Plug & Lead	 Lithium polymer Cell	Intentionally Blank

PARTS IN PLASTIC BAG

 Slide Switches (2)	 Header Stick	 Shunt Jumper
 Resistor (180 KΩ)	 Screw Terminal	 XH Socket
 TP4056 Breakout	 Tie-Wrap Mount	 Screws & Spacers (1 set)

NOTE: Parts may vary from photographs.

D1M-WX1 Schematic



Assembly of D1S-WX1 Shield Version Kit

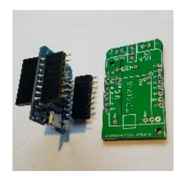
After completing the Arduino IDE installation on page 9, you can be confident that you will be able to program the microcontroller and that it operates correctly. You are now ready to assemble the printed circuit board shields. Find a good location to build the kit, collect your tools, and identify the parts. This is also a good time to read the ESD notice on page 3.

We will use the D1 Mini breakout board as a fixture to align the connectors on the charger and sensor boards. Read over this section and refer to the figures to make sure you understand the process. When placing the headers in any of the boards, first solder only one pin so that the header can be easily adjusted if it is not flat and square with the board.

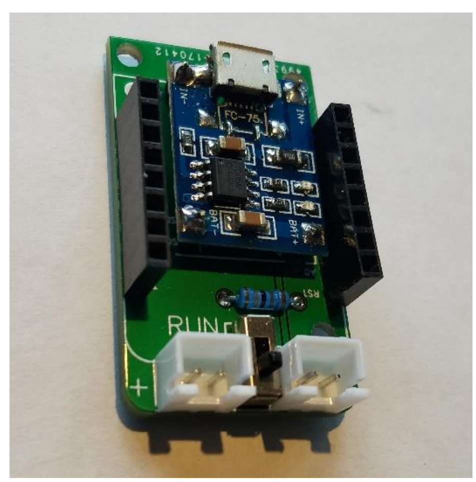
Assemble the D1 Mini Main Board

1. Identify the stackable headers. They are the headers with female sockets and long pins. Place two stackable headers on the D1 Mini PCB with the long tails of the pins through the PCB. The antenna and ESP8266 should face up.
2. Solder pins on the four corners on the underside of the D1 Mini being careful to keep the D1 Mini square with the PCB. The corner pins are RST, TX, 5V, and 3V3. Check that the headers are flat and square with the PCB.
3. Solder the remaining pins on the D1 Mini. (16 total)
4. Inspect the PCB.
5. Check that all solder joints are shiny and smoothly connect the header pins with the PCB solder pads.
6. Check that there are no shorts between PCB solder pads.

Assemble the D1S-LiSo Charger Shield

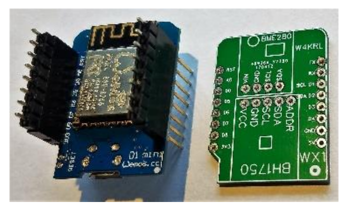


1. The general process is to first install the lowest profile components and then add progressively taller components.
2. Resistor Installation
    1. Place the 180 kilohm resistor in the resistor location (R1).
    2. Turn over the PCB and solder the two (2) resistor leads to the PCB.
    3. Trim the resistor leads with diagonal or flush cutters.
3. XH Connector Installation
    1. Install two XH sockets in locations marked PANEL and CELL. Make sure the notched opening in the connectors face outward from the PCB.
    2. Solder the pins on the underside of the board and trim the pins if necessary.
4. Slide Switch Installation
    1. Place one Slide Switch in the RUN location.
    2. Turn over the PCB and solder the three (3) Slide Switch pins to the PCB.
    3. Trim the pins.

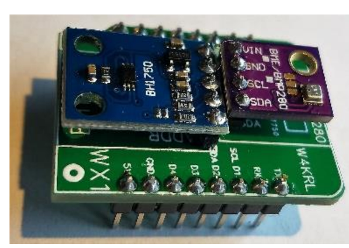


5. Install female headers.
    1. We will use the D1 Mini as a fixture to align the female headers on the D1S-LiSo board. Place the two female headers onto the male pins of the D1 Mini. Carefully engage the female header pins with the top side of the D1S-LiSo PCB.
    2. Solder just one pin on each header. Check that the headers are square and even with the PCB. Adjust, if necessary, by melting the solder on a pin and moving the header to a position square with the board.
    3. Solder all 8 pins of each header and inspect the joints. Sixteen (16) total.
6. TP-4056 Charger Assembly and Installation
    1. Cut four individual pins from the male header stock using diagonal or end cutters. Cut at the divisions between the pins. They do not take much force to cut. Be neat!
    2. Place the four pins on the PCB with the long tails through the PCB. Note that the board has pin locations for large and small versions of the TP4056 breakout board. Choose the holes that match the breakout board size.
    3. Carefully place the TP-4056 breakout board on the pins. Observe proper orientation of the pins, that is, BAT+ to BAT+, etc.
    4. Solder the pins to the TP4056 (4 places).
    5. Turn over the PCB. Keep the TP4056 flush on the PCB and solder the four TP4056 pins to the PCB.
    6. Trim the pins with diagonal or flush cutters.

Assemble the D1S-WX1 Sensor Shield

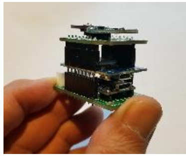


1. Assemble male headers
    1. Plug an 8-pin male header into each female header on the D1 Mini. This will keep the headers properly aligned.
    2. Set the D1S-WX1 board on the short pins of the male headers and solder 16 places.
    3. Inspect the solder joints.
2. BME280 Assembly and Installation
    1. Place the 4-pin header included with the BME280 on the PCB with the LONG tails through the PCB.
    2. Carefully place the BME280 breakout board on the pins.
    3. Solder the VIN pin to the BME280. Place the tip of your iron on the short side of the board to raise the board away from the PCB. Try to keep the board parallel to the PCB.
    4. Solder the remaining pins (3 places) to the BME280.
    5. Turn over the PCB and solder the four BME280 pins to the PCB.
    6. Trim the pins.
3. BH1750 Assembly and Installation



    1. Place the 5-pin header included with the BH1750 on the PCB with the LONG tails through the PCB.
    2. Carefully place the BH1750 on the pins.
    3. Solder the VCC pin to the BH1750. Place the tip of your iron on the short side of the board to raise the board away from the PCB. Try to keep the board parallel to the PCB.
    4. Solder the remaining pins (4 places) to the BH1750.

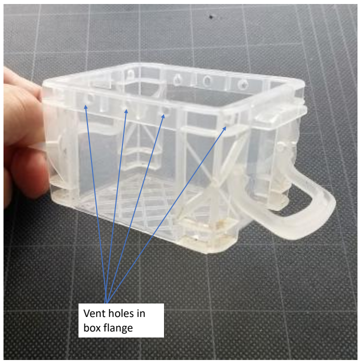
5. Turn over the PCB and solder the five BH1750 pins to the PCB.
6. Trim the pins and inspect all solder joints on all components.



Assemble the Shield Stack
- Plug the D1 Mini board into the D1S-LiSo charger board. Plug the D1S-WX1 Sensor board into the D1 Mini. This makes the shield stack.

Configure & Upload the D1M-WX1_IoT.ino Firmware
Follow the steps in APPENDIX A – ThingSpeak IoT Firmware to configure the ThingSpeak IoT firmware. You can view the weather station data on your ThingSpeak account and on the IDE serial monitor.

Final Assembly
1. Box Preparation
   1. Place the paper template into the box with the printed side down. Hold it against the bottom of the box with a finger. Use a permanent marker to mark the center of each hole on the outside of the box. Four places.



   2. Drill a 1/8-in (3 mm) hole in each marked spot. Clean the holes if necessary.
   3. Insert a screw through the bottom of the box and a spacer into the box. Hand tighten the hardware. Four places.
   4. Practice these steps first before gluing. The spacers will be glued to the underside of the board. Hot melt glue is nonconductive and can be carefully removed when cold if you make a mistake:
      1. Hold the shield stack on the tops of the spacers. Keep it square with the sides of the box.
      2. Remember the orientation of the stack and remove it from the box.
      3. Place a dab of hot melt glue on top of each spacer. If your glue gun cannot reach into the box, place the glue on the PCB where it touches the spacers.
      4. Return the stack to the box and hold it against the spacers until the glue sets.
   5. Unscrew the four screws and remove the stack from the box.
   6. Note that the bottom of the box has a grid of diamond shaped spaces. Drill ¼-in (6 mm) holes at multiple locations within these diamonds. Avoid drilling near the spacers.
   7. Drill 1/8-in (3 mm) holes in the flange at the top of the box. Ten (10) places.
   8. Affix the tie-wrap mount to one side of the box.
2. Solar panel
   1. Locate the free end of the two-conductor wire attached to one of the XH plugs. Note that the wires are marked with different patterns to identify each conductor.

WARNING: IT IS VERY IMPORTANT TO PROPERLY IDENTIFY THE POLARITY OF EACH WIRE!

   2. Place the plug into the PANEL connector and note the pattern on the wire that is closest to the - (negative) marking on the PCB. Write down the pattern on the negative wire. The other wire is the + (positive) wire. The next steps are easier if you remove the plug from the connector.

3. Strip insulation from about ¼-inch (6 mm) of each lead. Separately twist the strands in each lead and tin the leads by applying a small amount of solder.
4. Carefully solder the negative lead (-) to the negative (-) solder pad on the rear of the solar panel. This is the pad near the center of the panel. DO NOT OVERHEAT THE SOLAR PANEL.
5. Carefully solder the positive lead (+) to the positive (+) solder pad on the rear of the solar panel. This is the other pad near the center of the panel.
6. Apply a dab of hot melt glue to the wire to hold it in place against the solar panel.
7. This step glues the solar panel to the top of the box lid. The wire will exit through the glue under the panel on the edge of the lid. Run a thick bead of hot melt glue around the top on the box lid and quickly place the solar panel on the lid. Fill any gaps around the wire or under the panel by adding extra glue.
8. Plug the solar panel lead into the PANEL connector on the PCB.
9. Put the lid on the box with the wire routed under the lid.

3. LiPo Cell
    1. If the cell leads are provided with protective tape, carefully remove the tape from the cell leads.

    WARNING: KEEP LEADS FROM TOUCHING EACH OTHER!

    2. If the cell is provided with a connector, carefully cut each lead from the connector as close to the connector as possible. Cut the leads ONE BY ONE to prevent shorting the cell.

    WARNING: DO NOT CUT ACROSS BOTH WIRES AT THE SAME TIME!

    3. Strip about ¼-inch (6 mm) from each cell lead.
    4. Separately twist the strands in each lead and tin the leads by applying a minimum amount of solder.
    5. Locate the free end of the two-conductor wire attached the remaining XH plug. Note that the wires are marked with different patterns to identify each conductor.

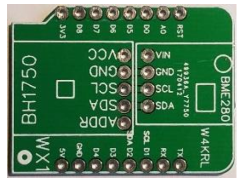
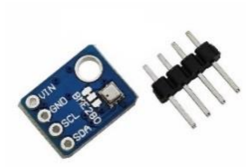
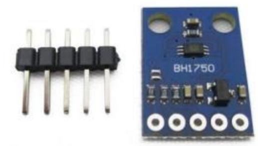
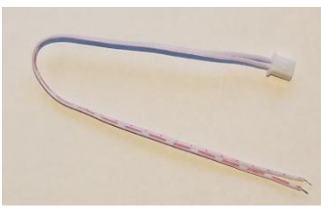
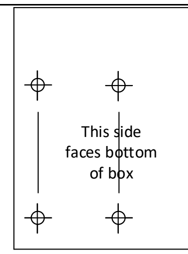
    WARNING: IT IS VERY IMPORTANT TO PROPERLY IDENTIFY THE POLARITY OF EACH WIRE!

    6. Place the plug into the CELL connector and note the pattern on the wire that is closest to the + (positive) marking on the PCB. Write down the pattern of the positive wire. The other wire is the - (negative) wire. The next steps are easier if you remove the plug from the connector.
    7. Strip insulation from about ¼-inch (6 mm) of each lead. Separately twist the strands in each lead and tin the leads by applying a small amount of solder.
    8. Carefully solder the positive lead (+) to the positive (RED) lead of the cell.
    9. Carefully solder the negative lead (-) to the negative (BLACK) lead of the cell.
    10. Carefully tape the individual joints with electrical tape. Wrap both joints with a final wrap.
    11. Plug the LiPo cell plug into the PANEL socket of the D1S-LiSo board.
    12. Set the RUN switch to the RUN position.
    13. Screw the shield stack into the box.
    14. Carefully place the LiPo cell in the box and close the box with the lid.

Congratulations! You have completed the assembly of your Internet of Things weather station.

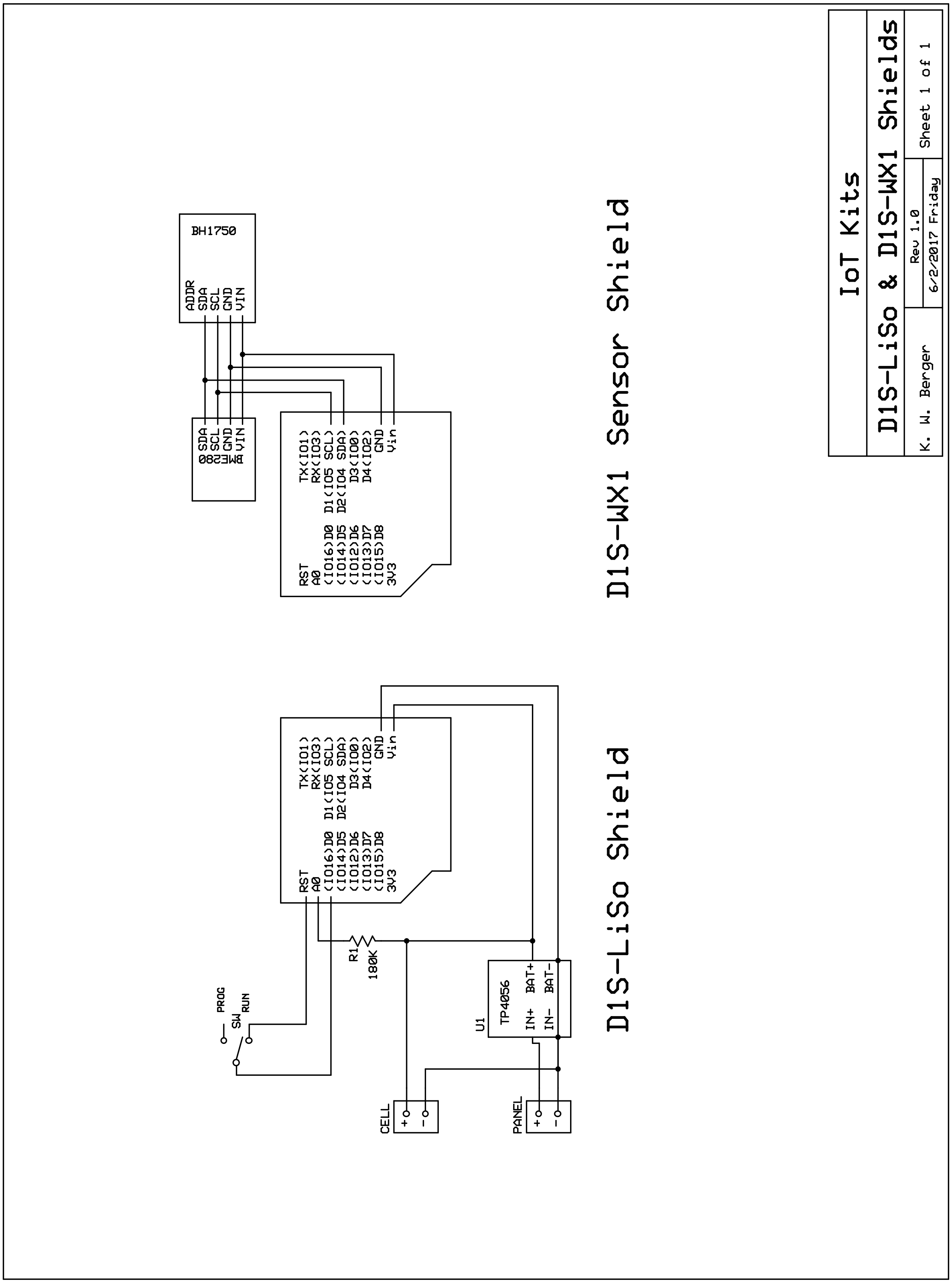
The next step is to mount the station. See page 25.

Illustrated Parts List for D1S-WX1 Shield Version Kit

 D1S-LiSo PCB	 D1S-WX1 PCB	 Box
 D1 Mini	 BME280 Breakout & Header	 BH1750 Breakout & Header
 XH Plug & Lead (2)	 Lithium polymer Cell	 Hole Template
 Slide Switch	 Header Stick	 Set of Headers
 Resistor (180 KΩ)	 Solar Panel	 XH Socket (2)
 TP4056 Breakout	 Tie-Wrap Mount	 Screws & Spacers (1 set)

NOTE: Parts may vary from photographs.

D1S-WX1 Shield Schematics



Mounting the Weather Station

The station should be mounted so that air can circulate around the box. It is best to mount the box from the side so that the vent holes in the bottom of the box are not blocked.

It can be attached to a deck rail or tree branch by looping tie wraps around the box. Do not rely on supporting the box from the tie wrap mount alone as the glue may loosen in weather. One or more tie-wraps connected in series will work on large objects. Duct tape can be useful.

Avoid a location exposed to direct sunlight. It will raise the temperature within the box affecting the accuracy of the temperature measurement.

Prior to mounting the station, you may wish to charge the LiPo cell by plugging your Micro USB connector into the socket on the TP4056 charge controller. The orange light indicates charge. The blue light indicates full charge.

Using the APRS Station

You must be a licensed amateur radio operator and have a valid callsign. Any license class may use the station as it does not require transmission on the amateur bands.

The config.h file for D1M-WX1_IoT-APRS.ino must be filled in with the information for your station. You must also enter the same information you provided for the IoT-only version of the firmware. See page 32 APPENDIX B – CONFIG.H as a reference.

Obtain an APRS passcode at www.george-smart.co.uk/aprs/aprs_callpass/ or n5dux.com/ham/aprs-passcode.

You will need to know your exact station location in decimal degrees and altitude in meters. To find your position, go to veloroutes.org, enter your address, and write down the position in decimal degrees and altitude in meters. Multiply the decimal portion of the latitude and longitude by 60. Rewrite your position in the APRS format of ddmm.hhN/dddmm.hhW. For example, the location of the Smithsonian Institute Udvar-Hazy Center is 38.9122679º/-77.4428585º. The decimal portion of the latitude is 0.9122679. Multiply by 60 to get 54.736074. Round this to two decimal places: 54.74. Prefix the latitude with the integer portion of the latitude and place an N as a suffix: 3854.74N. The suffix for southern latitudes is S. Do the same for longitude: 0.4428585 x 60 = 26.57W. The suffix is W for west (negative) longitudes and E for East. The final location is 3854.74N/07726.57W. Read this as 38 degrees 54.74 minutes north by 77 degrees 26.57 minutes west. Note that the integer portion of the latitude must contain two characters and the integer portion of the longitude must contain three characters. Pad the number with leading zeros if necessary. Make sure your location does not contain a minus sign.

Set the PROG/RUN switch to PROG and upload the new firmware. Set the switch back to RUN. The unit will transmit weather and telemetry every 10 minutes. Go to an APRS-IS website and search for your callsign and SSID to see your data:

- http://aprs.fi/ - recommended site
- http://www.wulfden.org/APRSQuery.shtml
- http://www.qsl.net/n9wtm/
- http://www.proaviator.com/findu/

Description of Operation

The D1M-WX1 and D1S-WX1 IoT Weather Stations use an ESP8266 System on Chip microcontroller to read and interpret weather data from sensors, format the data so that it is understandable by online services, and transmit the data to the Internet over Wi-Fi. The unit is completely solar powered with a lithium polymer cell to maintain operation in darkness.

Power Supply

Solar energy derived from sunlight is converted into electrical energy by a solar panel. The panel is composed of a series string of silicon photovoltaic cells. In bright sunlight, the panel produces 6 volts at up to 100 mA.

To continue operation at night and on dark days, a lithium polymer cell stores excess electrical energy when available. A TP-4056 charger regulates the solar-generated electrical power to maintain the charge without overcharging the cell.

The ESP8266 microcontroller operates when supplied with voltage between 3.0 and 3.5 V DC. A fully charged cell has a terminal voltage of about 4.2 V DC that gradually drops to 3.0 V DC as energy is consumed. The D1 Mini breakout board includes a voltage regulator that supplies the microcontroller with 3.0 to 3.3 V DC from an input voltage in the range of 3.1 to 10 V DC connected to the pin labelled VIN on the D1 Mini. When the lithium polymer cell in connected to this terminal, the cell will safely supply the D1 Mini. The 3.3 V DC output of the D1 Mini regulator is connected to the 3V3 pin. A limited amount of current may be drawn from the 3V3 pin to supply the sensors with electrical power.

Switch SW1 is the power switch for the station. When in the ON position, the cell is connected to the D1 Mini. In the OFF position, the cell can be charged with solar power or with power drawn from a USB supply connected to the micro USB jack on the TP4056 breakout board.

The D1 Mini can also be powered through its micro USB jack, however, a diode on the D1 Mini breakout board prevents power from flowing to the cell. The cell will not charge when power is supplied only to the D1 Mini board from a USB connection.

ESP8266 System on Chip

The ESP8266 is a powerful microcontroller that incorporates a Wi-Fi transceiver and flash memory. It is a surface mount device (SMD) that requires a few additional components to function in the weather station. Two important supporting components are the voltage regulator previously described and a USB adapter. These and other components are mounted to the back side of the breakout board.

The main function of the USB adapter is to convert signals to and from a PC to RS-232 (EIA-232) serial signals used by the ESP8266 for programming and communication. Additional components enable the USB chip to control the programming and operating state of the ESP8266.

The ESP8266 is supplied with a firmware LUA interpreter. LUA is an interesting computer language that does not yet have the wide support enjoyed by the Arduino flavor of C++. Fortunately, a dedicated band of programmers has developed an ESP8266 core. When the core is loaded into the Arduino Integrated Development Environment (IDE) it enables programming the ESP8266 exactly like it was an Arduino. This makes the huge Arduino ecosystem available to users of the ESP8266.

Deep Sleep

One of the many interesting features of the ESP8266 is its ability to go into a very low power mode called deep sleep. This is an important feature for the weather station since it relies upon the limited energy available from the lithium polymer cell during periods of darkness.

Measurements show that the D1 Mini and the weather sensors draw around 80 µA when in deep sleep; a very small amount of current. However, in the six to ten seconds the station is transmitting data to the Internet, it may draw up to 300 mA with an average of about 80 mA. If the sleep mode was not available, the station would require a much larger solar panel and storage cell.

When in deep sleep, the only active part of the ESP8266 is an internal real-time clock (RTC). The weather station firmware sets the RTC to wake up the ESP8266 after ten minutes of deep sleep. When the sleep interval has expired, the RTC pulls pin D0 to ground. To wake up the processor, the reset pin RST must be pulled to ground. When the RUN/PROG switch is in the RUN position, the reset signal from D0 is connected to the RST pin and the chip will wake itself up every ten minutes. Unfortunately, pin D0 is also used by the USB adapter to control the programming of the ESP8266 and would be affected by the RTC reset function. When the switch is in the PROG position, the connection between D0 and RST is opened allowing normal programming of the chip. It is important to return the RUN/PROG switch to the RUN position after programming.

I2C Interface Bus

Data and control information is exchanged between the sensors and the microprocessor over an I2C serial bus. I2C stands for Inter-Integrated Circuit and is pronounced "eye-squared cee". The I2C bus is ideally suited for the relatively low data rates needed for weather measurements and needs only two data pins on the D1 Mini. The software drivers for the I2C bus are built in to the Arduino software making it very easy to connect compatible I2C sensors to the microcontroller. Each I2C device must have a unique address. Some devices allow selection of the address and some do not. The sensors used in this weather station have fixed addresses. The driver library software for the BME280 and BH1750 have the addresses hard coded.

I2C Expansion Header

The D1M-WX1 has a 4-pin header marked I2C on the printed circuit board to provide for remote sensors or the addition of other I2C devices.

Each I2C device must have a unique address. In some cases, the address is selectable by the user. This is not the case with the BME280 and BH1750 sensors. The onboard sensors must be disabled if remote BH1750 and BME280 sensors are added. Address conflict can be avoided by removing power to the onboard sensors. This is easily done by removing the shunt jumper from the 2-pin LOCAL header.

NOTE: The I2C header pins are mislabeled on printed circuit boards marked Rev A. The labels for SDA and SCL are reversed. The proper labeling from left to right should be: VCC, GND, SDA, SCL

BME280 Environmental Sensor

The BME280 Environmental Sensor by Bosch Sensortech is a high accuracy sensor for pressure, temperature, and humidity. These are the primary weather parameters measured by the station.

The actual device is in a metal package 2.5 mm square and less than 1 mm thick. The breakout board requires just four wires: VIN and GND for power, and SCL and SDA for the I2C serial bus.

The measurements made by the BME280 are linear, floating point values. As with most sensors, the measurement units are metric and may be converted to US units by simple mathematic routines.

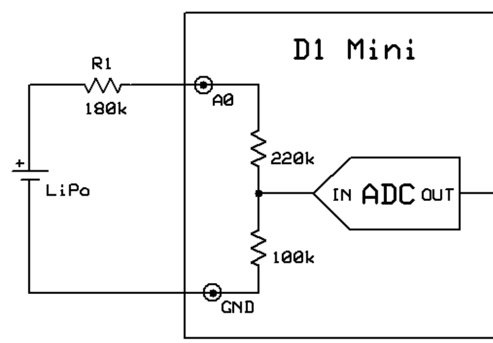
Generally, the placement of the weather station will not affect pressure and humidity measurements. Temperature measurements may be affected by solar heating of the weather station enclosure. If you notice unrealistically high temperatures during the day, you may wish to relocate your station so that direct sunlight does not fall on it. Good ventilation above and below the station is also important.

BH1750 Light Intensity Sensor

The BH1750FVI Ambient Light Sensor by ROHM Semiconductor is an I2C device with a spectral response very similar to the human eye. Unlike the eye, however, its amplitude response is linear rather than logarithmic. Its range of 1 to 65,535 lux responds to intensities from cloudy moonlight to medium bright sunlight. This is adequate to provide a relative measure of the brightness of cloudy and sunny days.

The light intensity measurement is strongly affected by the location of the weather station. Useful measurements can be obtained by a station not in direct sunlight. As with most light sensitive devices, the angle of incidence of the light strongly affects the measurement. This means that light from directly over the sensor will register a higher intensity than light coming from an angle. This is similar to the heating effect of sunlight on the earth and accounts for winter being colder than summer.

Charge Monitor



In addition to weather data, the station reports the lithium polymer cell voltage to provide an indication of the state of the charge. The voltage is measured using the analog to digital converter (ADC) built into the ESP8266. The ADC has a range of 0 to 1 Volt that is reported as a 10-bit digital value from 0 to 1023. Since the cell voltage may exceed 4.2 V DC at full charge, it is necessary to use a voltage divider to reduce the voltage applied to the ESP8266. The D1 Mini has a voltage divider composed of a 220 kilohm resistor and 100 kilohm resistor that increases the range to 0 to 3.2V DC. The station adds an external 180 kilohm resistor (R1) to further increase the range to 5 V DC.

While all three voltage divider resistors are precision devices, an optional calibration step is provided in the D1M-WX1 kit to ensure good accuracy.

Received Signal Strength Indicator

The Received Signal Strength Indicator (RSSI) is a function built into the Wi-Fi receiver of the ESP8266. RSSI is a measure of the power level of the Wi-Fi signal from your access point. There is no standard for the accuracy of the RSSI value, however it is common (whether accurate or not) to show the value as decibels referenced to a milliwatt written as dBm. Since the received power is very small the RSSI is a negative value when given in dBm. The closer the figure is to zero, the better. As a general example, a good signal would be -50 dBm, a reasonable strength would be -75 dBm, a bad one would be -90 dBm, and -100 dBm would provide no service at all.

The weather station reports the RSSI to help you judge how reliable the Wi-Fi signal is at the station location. It may also be an interesting experiment to see if there is a relationship between signal strength and weather conditions. For example, higher humidity may reduce signal strength.

Firmware

Weather station firmware is described in a separate document at w4krl.com.

APPENDIX A – ThingSpeak IoT Firmware

These instructions will install and configure the Internet of Things firmware. The station will report weather data to the free ThingSpeak IoT service. The IoT code is smaller and easier to use than the APRS-IS code. Do this installation before installing the APRS-IS firmware.

1. Find your station altitude and geographic location by entering its street address in veloroutes.org. Record the latitude and longitude in decimal degrees and altitude in meters. Having the correct altitude is important for the accuracy of the barometric pressure sensor. Add an estimate of the height above ground level if you are mounting it in a raised position. For example, the second floor of many homes is about 3 m above ground.
2. Open a free account at ThingSpeak:
    1. Open www.thingspeak.com in your browser.
    2. Click "Sign Up".
    3. Select and record your UserID.
    4. Change Time Zone to GMT-5:00 Eastern Time or the appropriate zone for the station's location.
    5. Select and record your Password.
    6. Agree to Terms.
    7. Click Create Account.
    8. On next screen click "New Channel".
    9. Choose and record the channel name.
    10. Enter field definitions and check the box beside each of the following:

        1 = Temperature
        2 = Humidity
        3 = Station Pressure
        4 = Sea Level Pressure
        5 = Light Intensity
        6 = Cell Voltage
        7 = RSSI
        8 is not used. Do not change the defaults.
    11. Check Public.
    12. Elevation = your elevation in meters.
    13. Check Show Location.
    14. Latitude = your latitude in decimal degrees. Positive for north.
    15. Longitude = your longitude in decimal degrees. Negative for west.
    16. Click on Save Channel.
    17. Record your API Keys and ChannelID.
3. Set the PROG/RUN switch on the PCB to PROG.
4. Using the Arduino IDE, open file D1M-WX1_IoT.ino. Note that the IDE window has a tab for D1M-WX1_IoT and config.h.
5. Select the config.h tab and enter the information indicated:
    1. dmmVoltage and adcVoltage determined in the calibration step for the single board version of the kit or use the defaults for the shield version.
    2. STATION_ELEVATION in meters.

    3. SLEEP_INTERVAL = 600; [For initial testing, try 15 but change it to 600 for final installation.]
    4. ssid[] = your Wi-Fi SSID in quotes. Use only 2.4 GHz networks, 5 GHz networks will not work!
    5. password[] = your Wi-Fi password in quotes.
    6. ChannelID = your ThingSpeak channel ID in quotes.
    7. apiWriteKey = your ThingSpeak API Write Key in quotes.
6. Use IDE menu `File > Save` or type `Control+S` to save your changes.
7. Compile and upload the firmware with the upload icon or menu `Sketch > Upload`.
8. Turn the PROG/RUN switch to RUN.
9. Log on to your ThingSpeak account to see your Internet of Things data.

APPENDIX B – CONFIG.H

This is the config.h file for the APRS firmware. Config.h for the IoT firmware omits the APRS portion. The items you must enter are highlighted in yellow.

```
// config.h

// This configuration file should reside in the same Arduino
// directory as the weather station file D1M-WX1_IoT-APRS.ino.

// ****************************************************
// **************** STATION FACTORS *********************
// ****************************************************

// If you have performed the voltage calibration steps, enter
// the voltage you measured on your digital multimeter and
// the voltage reported by the ADC on the serial monitor.
// If you have not performed the calibration, do not change
// the default values. They should be equal.

const float dmmVoltage = 4.45;
const float adcVoltage = 4.45;

// station altitude in meters
// https://www.freemaptools.com/elevation-finder.htm
const float STATION_ELEV = 90.0;

// update interval in seconds
// must be longer than 15 seconds
// suggest 30 seconds for testing, 600 or 900 for use
const long SLEEP_INTERVAL = 600;

// ****************************************************
// ********************* WIFI LOGON *********************
// ****************************************************

// ENTER YOUR WI-FI SSID
// 2.4GHz Wi-Fi only!
const char ssid[] = "yourWiFi_SSID";

// ENTER YOUR PASSWORD
const char password[] = "yourWiFi_PW";

// ****************************************************
// ******************** THINGSPEAK **********************
```

```
// *******************************************************

// ThingSpeak Channel ID
String ChannelID = "yourChannelID";

// API write key for your ThingSpeak channel
String apiWriteKey = "yourAPI_write_Key";

// ThingSpeak Server - do not change
const char IOT_SERVER[] = "api.thingspeak.com";

// *******************************************************
// ******************** APRS-IS *************************
// *******************************************************

// station location in APRS format
// NOTE: latitude must have 4 digits left of the decimal. Pad with 0s.
//       longitude must have 5 digits left of the decimal. Pad with 0s.
// Find your location at http://veloroutes.org/elevation/
const char APRSlocation[] = "ddmm.hhN/dddmm.hhWW";

// select an APRS-IS Tier 2 server close to your station
// or use the default
// for list of servers: http://www.aprs2.net/serverstats.php
const char APRS_SERVER[] = "iad.aprs2.net";
const int APRS_PORT = 14580;   // do not change

// define your callsign and passcode
// use callsign-13 for weather stations
// see http://www.aprs.org/aprs11/SSIDs.txt
const String APRScallsign = "callsign-ssid";

// for passcode www.george-smart.co.uk/aprs/aprs_callpass/
// or http://n5dux.com/ham/aprs-passcode/
const char APRSpasscode[] = "nnnn";
```

APPENDIX C – APRS Tier 2 Servers in the US & Canada

Complete global list at http://www.aprs2.net/serverstats.php

Abilene, TX	texas.aprs2.net
Acton, ON	ontario.aprs2.net
Augusta, ME	maine.aprs2.net
Baltimore, MD	bwi.aprs2.net
Chilliwack, BC	ycw.aprs2.net
Columbia, MO	missouri.aprs2.net
Columbus, OH	cmh.aprs2.net
Corvallis, OR	oregon.aprs2.net
Dulles, VA	iad.aprs2.net
Dulles, VA	iad2.aprs2.net
Edmonton, AB	edmonton.aprs2.net
Greenwood, AR	arkansas.aprs2.net
Kansas City, MO	central.aprs2.net
Los Angeles, CA	socal.aprs2.net
Mishawaka, IN	indiana.aprs2.net
Montreal, QC	raqi.aprs2.net
New York, NY	lga.aprs2.net
Providence, RI	newengland.aprs2.net
San Jose, CA	sjc.aprs2.net
San Juan, PR	puertorico.aprs2.net
Seattle, WA	northwest.aprs2.net
Toronto, ON	yyz.aprs2.net
Tucson, AZ	tus.aprs2.net
Vancouver, BC	bc.aprs2.net
Vancouver, BC	vancouver.aprs2.net

APPENDIX D - Troubleshooting

PROBLEM	CHECK
Unit transmits data only once.	• The PROG/RUN switch may be in wrong position. Set PROG/RUN switch to RUN. • Cell may not be charged. Connect a 5V USB source to the Micro USB socket on the TP4056 board until the blue LED lights. • The unit may be out of Wi-Fi range. Move it closer to the Wi-Fi access point.
Unit does not transmit data.	• Check that the Power switch is in the ON position. • Cell may not be charged. Connect a 5V USB source to the Micro USB socket on the TP4056 board until the blue LED lights. • Cell wires may not be properly engaged in screw terminals. Verify wires are stripped and in contact with terminals. • The unit may be out of Wi-Fi range. Move it closer to the Wi-Fi access point. • Verify that your Wi-Fi logon information is correct and that you are using a 2.4MHz network, not a 5GHz network.
IDE returns espcomm error	• Check that the unit is connected to your PC. • Check that the cable is connected to the D1 Mini – not the TP4056 charger board. • Check that proper COM port is selected. • Check that PROG/RUN switch is in the PROG position.
IDE returns error compiling for board	• Verify that menu Tools > Board is set for WeMos D1 R1 & mini.
APRS data does not appear	• Check your APRS credentials. • Check that your location information is in the correct format.

APPENDIX E – Useful Links

APRS by Bob Bruninga, WB4APR	http://aprs.org
APRS general article	http://www.arrl.org/automatic-packet-reporting-system-aprs
APRS Google Maps	http://aprs.fi
APRS Passcode (licensed amateurs only)	http://www.george-smart.co.uk/aprs/aprs_callpass/ http://n5dux.com/ham/aprs-passcode
APRS Protocol	http://www.aprs.net/vm/DOS/PROTOCOL.HTM
APRS Protocol Reference	http://www.aprs.org/doc/APRS101.PDF
APRS SSID Recommendations	http://www.aprs.org/aprs11/SSIDs.txt
APRS-IS connection	http://www.aprs-is.net/Connecting.aspx
Arduino C++ Reference	https://www.arduino.cc/en/Reference/HomePage
Arduino IDE	https://www.arduino.cc/en/Main/Software
Arduino Library Instructions	https://www.arduino.cc/en/guide/libraries
BH1750 Data Sheet	http://image.dfrobot.com/image/data/SEN0097/BH1750FVI.pdf
BME280 Data Sheet	https://www.bosch-sensortec.com/bst/products/all_products/bme280
Civilian Weather Observer Program	http://wxqa.com/
D1M-WX1 Firmware	http://w4krl.com/wp-content/uploads/D1M-WX1_Weather_Station.zip
D1 Mini Information	https://www.wemos.cc/product/d1-mini.html
ESP8266 Arduino core	https://github.com/esp8266/Arduino
ESP8266 Resources	http://www.esp8266.com/
Geographic Location & Altitude	http://mygeoposition.com/ https://www.freemaptools.com/elevation-finder.htm http://veloroutes.org/elevation/
IoT Kits at W4KRL	http://w4krl.com/iot-kits
IoT Kits firmware	https://github.com/W4KRL
ThingSpeak	http://www.thingspeak.com
ThingSpeak tutorials	http://community.thingspeak.com/tutorials/