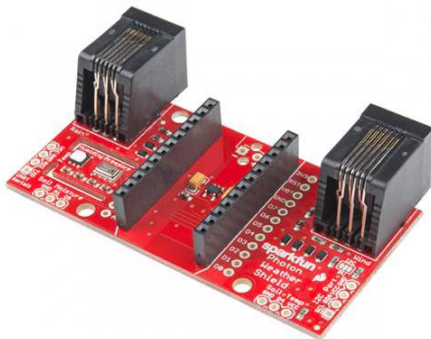




Photon Weather Shield Hookup Guide V11

Introduction

Have you ever wanted to have your own weather station? Or how about your own thermostat capable of controlling your home climate from the Web? Using the Photon from Particle coupled with the Photon Weather Shield from SparkFun, you can now connect your weather related project to the Internet of Things!



SparkFun Photon Weather Shield

© DEV-13674

Note: All SparkFun shields for the Photon are also compatible with the (now retired) Core from Particle. The WKP, DAC and VBT pins on the Photon will be labeled A7, A6 and 3V3*, respectively, on the Core, but will not alter the functionality of any of the Shields.

Required Materials

If you are looking to just measure **temperature**, **humidity**, and/or **barometric pressure**, you can integrate the Photon Weather Shield into your project right out of the box! No soldering necessary!

The shield also has numerous **optional** ports on which other sensors can be connected. Those optional devices will be discussed briefly in this tutorial, but examples showing how to use those external peripherals will be included in another tutorial, linked at the end of this one.

To follow along with this tutorial, you will need the following.

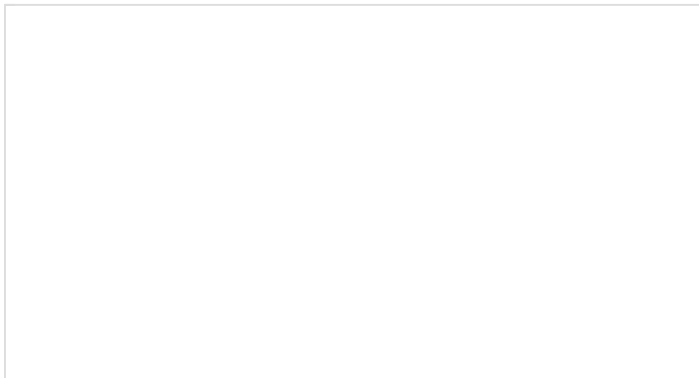
Photon Weather Shield Hookup SparkFun Wish List	
	<p>SparkFun Photon Weather Shield DEV-13674</p> <p>The SparkFun Photon Weather Shield is an easy-to-use add-on board...</p>
	<p>Particle Photon (Headers) WRL-13774</p> <p>Particle's IoT (Internet of Things) hardware development board, the P...</p>
	<p>USB microB Cable - 6 Foot CAB-10215</p> <p>USB 2.0 type A to micro USB 5-pin. This is a new, smaller connector f...</p>

Suggested Reading

The following are suggestions for other resources to read before getting started with your Photon Weather Shield.

- If you have never worked with the Photon or the Core before, we highly recommend visiting the getting started documentation available on the Particle site.
- We have also written a Photon Development Guide that goes over three different methods for developing code for the Photon.

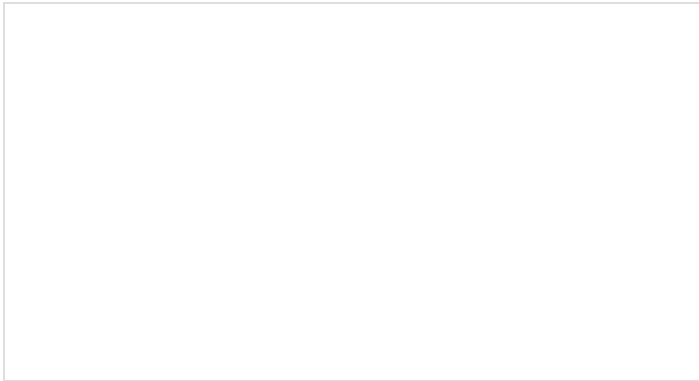
Check out the Hookup Guides for each of the sensors located on the shield for more information specific to that sensor.



Si7021 Humidity and Temperature Sensor Hookup Guide

MAY 5, 2016

The Si7021 humidity and temperature sensor is an easy to use, digital, low-cost sensor to aid you in all your environment sensing needs.

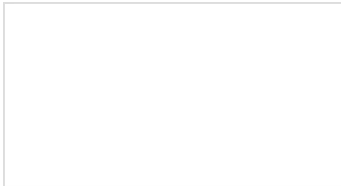


MPL3115A2 Pressure Sensor Hookup Guide

DECEMBER 5, 2013

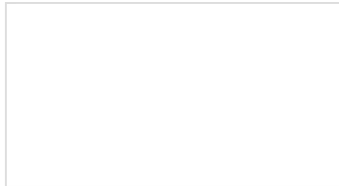
Getting started with the MPL3115A2 Pressure Sensor.

If you are unfamiliar with any of the concepts below, we suggest checking out those tutorials as well.



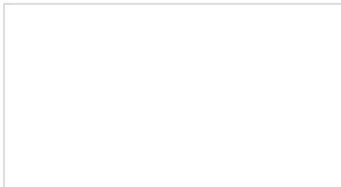
Serial Communication

Asynchronous serial communication concepts: packets, signal levels, baud rates, UARTs and more!



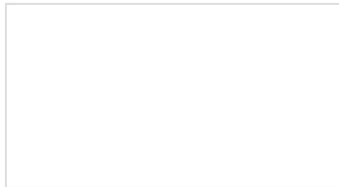
Using GitHub

How to use repositories and GitHub for version control.



I2C

An introduction to I2C, one of the main embedded communications protocols in use today.



Serial Terminal Basics

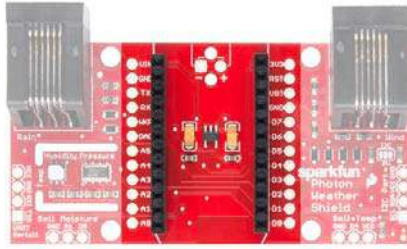
This tutorial will show you how to communicate with your serial devices using a variety of terminal emulator applications.

Hardware Overview

The Photon Weather Shield has a lot of functionality packed into one tiny package. Let's go over each section of the shield.

Photon Footprint

Both the Core and the Photon fit right onto the shield. Copper pours underneath the antenna were restricted so as not to interfere with wireless connections. Each pin is also broken out to the sides of the Photon for accessibility. When attaching a Photon, be sure to line up the beveled end of the Photon with the beveled silkscreen on the PCB.



Power

The simplest way to power the shield is to attach a Photon, and then power the Photon through the micro USB connector. This will power the Photon as well as all the components on the shield.



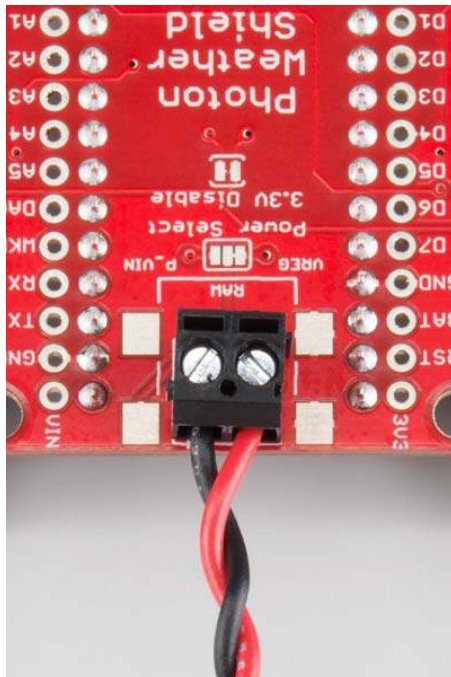
The downside to that power scheme is that micro USB connectors tend to be fragile when put under a lot of mechanical stress, and they can rip right off the PCB if pulled too hard. Thus, we provided a few other options for power connectors.

On the underside of the shield, you'll find a footprint for both an SMD Barrel Jack and a 2-pin, 3.5mm screw terminal. Either of these can be soldered to the PCB and used for alternate power inputs. The maximum voltage supplied on these alternate connectors should **not exceed 12V**. For a detailed explanation, read on.



Powered through the Barrel Jack.

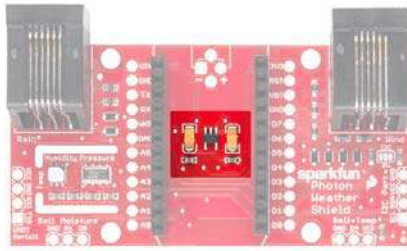
For the screw terminal, you can solder it to either side of the shield, since it fits underneath the Photon. Be sure to keep track of which pin is (+) and which is (-).



Powered through the Screw Terminal.

On-Board 3.3V Regulator and Power Solder Jumpers

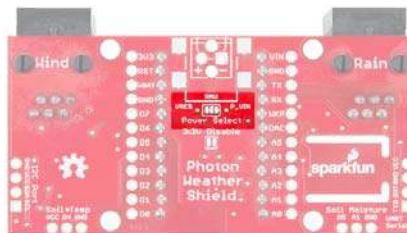
There is also a 3.3V regulator on the shield. If you are powering the Photon through the micro USB connector, this regulator is bypassed. Powering through one of the alternative power connectors mentioned above routes power **through the shield's 3.3V regulator**, which is then tied to the 3.3V rail on the Photon, powering it as well.



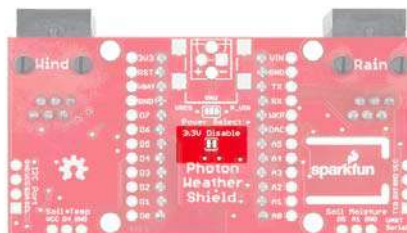
On-board 3.3V regulator and accompanying circuitry.

The main benefit to using the on-board regulator is that it has a higher maximum voltage rating than the regulator located on the Photon. As stated in the Photon datasheet, if power is supplied directly to the VIN pin on the Photon, the voltage should be regulated between **3.6VDC and 5.5VDC**. In contrast, the MIC5219 regulator is rated for **2.5VDC to 12VDC**, as per its datasheet. Being able to power this system at 12V is handy for solar-powered weather stations.

However, if you would rather have the alternative power source route power through the regulator on the Photon (for lower current consumption during sleep perhaps), simply cut the trace on the **Power Select** jumper (between the **VREG** and **RAW** pads), and add a blob of solder between **RAW** and **P_VIN** (Photon VIN) pads. Just be sure to **not exceed voltages of 5.5-6V** once this alteration has been made.



Warning! Never power the Photon/Shield through both the micro USB connector and the barrel jack or screw terminal at the same time! If you need both connected for whatever reason (programming in DFU mode over USB while powered via the barrel jack, for example), you'll need to cut the trace on the **3.3V Disable** jumper, pictured below. Only cut this trace if you **have not** altered the Power Select jumper



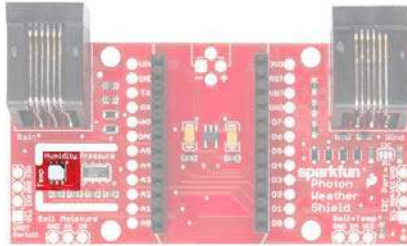
On-Board Sensors

The weather shield comes with two on-board sensors. Together, these two sensors can give you a lot of information about the environmental conditions around you or your project.

Si7021 or HTU21D Humidity and Temperature Sensor

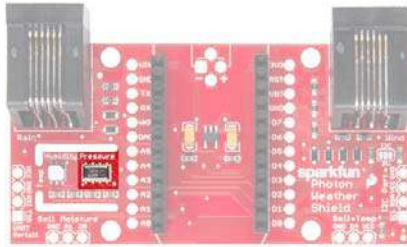
There are two versions of the Weather Shield, one containing the HTU21D and one containing the Si7021-A10. The HTU21D can be found on older versions of the Photon Weather Shield, while the Si7021 can be found on the latest version of the shield, V11.

The Si7021 is functionally identical to the HTU21D, and they both share the same register addresses and command codes. To make it easier on the user, the Photon Weather Shield library automatically detects which sensor is present on your board without needing a separate library. More on that later.



MPL3115A2 Barometric Pressure Sensor

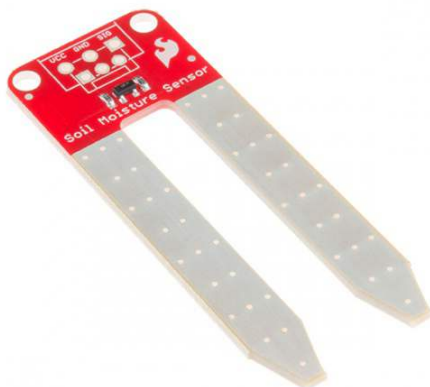
The MPL3115A2 is a MEMS pressure sensor that provides Altitude data to within 30cm. The sensor outputs are digitized by a high-resolution 24-bit ADC and transmitted over I²C. Pressure output can be resolved with output in fractions of a Pascal, and Altitude can be resolved in fractions of a meter. It provides 12-bit temperature measurements in degrees Celsius. This sensor also communicates over I²C and comes connected to the Photon's I²C bus by default.



Optional Peripherals

Soil Moisture Sensor

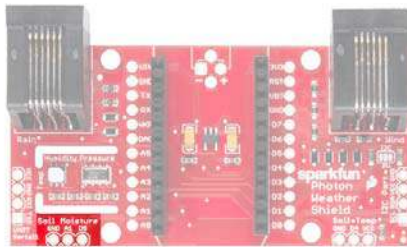
For those who want to keep track of their houseplant conditions or for those who want to know topsoil conditions in their gardens, the weather shield has an optional port for a Soil Moisture Sensor.



SparkFun Soil Moisture Sensor

© SEN-13322

Leaving the soil moisture sensor powered all the time leads to corrosion of the probes. To account for this, this port breaks out **Digital Pin D5 as the power pin** for the sensor, allowing the Photon to power the sensor, take a reading, and then disable power, giving the probes a longer lifespan. The moisture level can be read on **Analog Pin A1**.



More information about the Soil Moisture sensor can be found in the Hookup Guide.

Soil or Water Temperature Sensor

In addition to soil moisture, you can also add a waterproof temperature sensor. Whether you have a pond you'd like to monitor near your weather station or you want to know if the ground is still frozen or not, this can be a great addition to any weather station.

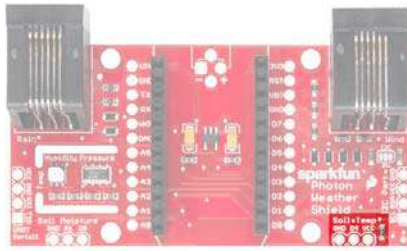
The waterproof version of the DS18B20 temperature sensor is a go-to option for many users looking to sense the temperature of various environments that may not be so kind to electronics.



Temperature Sensor - Waterproof (DS18B20)

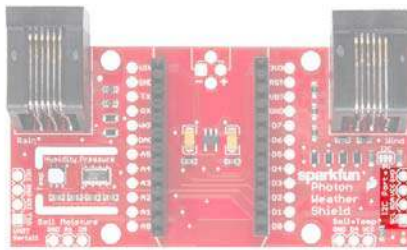
© SEN-11050

A port for this sensor is broken out along with the necessary 4.7K pull-up resistor located between the Vcc and Signal lines. The Signal line is connected to **Digital Pin D4**. Communicate with this sensor using the OneWire and Dallas Temperature libraries.



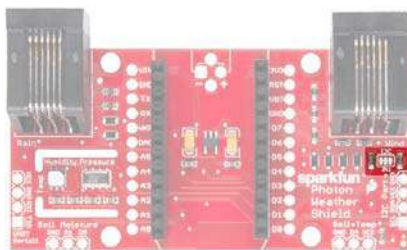
I²C Port

I²C is becoming increasingly popular as a means by which to communicate with sensors. Thus, it made sense to add an I²C port to allow users to add other sensors to their project. One great example would be adding an external light sensor. Since most weather stations need to be enclosed in a weatherproof housing, having a light sensor on-board didn't make much sense. However, adding the I²C port allows one to connect sensors such as the TSL2561 Luminosity Sensor or the ISL29125 RGB Light Sensor along with a whole slew of other I²C sensors.



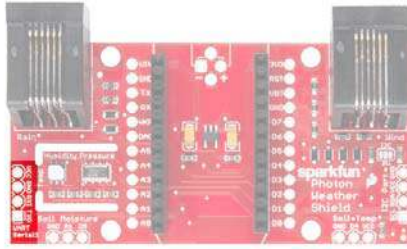
I²C Pull-ups

Many SparkFun I²C Breakouts have solder jumpers for disabling the pull-up resistors necessary on most I²C designs. The Photon Weather Shield is no different. If you already have a sensor that has pull-ups, you can disable them on the shield by cutting the traces with a hobby knife. Or, if you would rather leave the shield as is, you can disable the pull-ups on every I²C you add to the bus. The important thing to remember is that only one device on the bus needs to have its pull-ups active. All others should be disabled.



Serial 1 Port

There is another port broken out, this time for the USART Serial 1 port on the Photon. Serial ports are great for debugging. They also allow for various other wireless technologies to be attached for mesh networking or for user feedback via an LCD or other display. Let's say you wanted additional sensors around you home, well beyond the range of your WiFi signal. You could attach an Xbee Explorer to the Serial 1 port on the Weather shield and have your other node (consisting of a stand-alone XBee or an XBee Arduino combo, for example) out in the field, feeding data back to the Photon, which then sends the collective data to the web.



Another use case is wanting to have your weather station outside while still receiving data on a display indoors somewhere. This could easily be accomplished by attaching a BlueSMiRF Bluetooth module to the Serial 1 port and having another BlueSMiRF attached to a Serial Graphic LCD display installed in your home.



A SparkFun BlueSMiRF Silver connected to the Serial 1 Port using a 6-pin, right-angle female header with the outermost pins removed.

The pins on the shield match up with any of our BlueSMiRF modules. For other wireless solutions, you may have to wire each pin manually as they may not match up exactly.

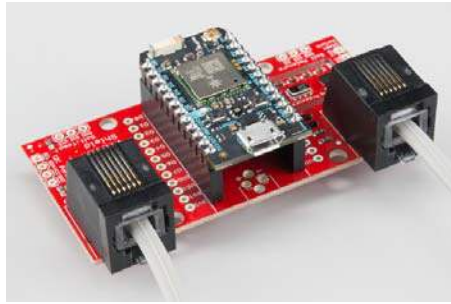
RJ-11 Weather Meter Connectors

If you would like to place your Weather Shield outdoors and create a fully-functional weather station, you can grab a set of Weather Meters and connect them to the RJ-11 connectors located on the Weather Shield. With these meters attached, you can measure **wind speed**, **wind direction** and **rainfall**.



Weather Meters

© SEN-08942



Throw in a stand, and you have yourself the makings for a very powerful weather station.



Particle IDE

If this is your first time using a Particle Photon, there are some extra steps you'll need to complete before you can follow the example in this tutorial.

First-time users will need to create a free account build.particle.io. Once your account is created, you'll need to claim your Photon. Detailed instructions on how to do this can be found on Particle's website.



Once your account is created and your Photon is claimed, head back to build.particle.io, and log in.

Collecting Weather Data

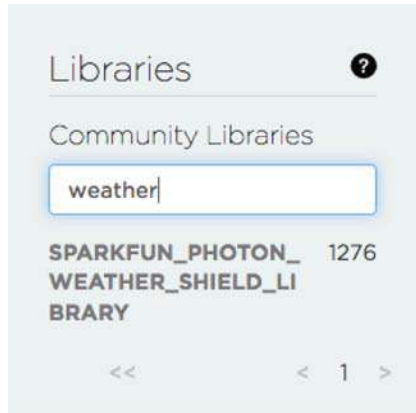
If you have not done so already, attach your Photon to the Weather Shield, and apply power through the Micro USB jack on the Photon, plugging the other end of the USB cable into your computer. If you opted for one of the alternate power schemes mentioned in the Hardware Overview section, power your Photon and shield accordingly.

Your Photon's RGB LED should be breathing cyan to indicate that you are connected to the Internet and are ready to upload code.

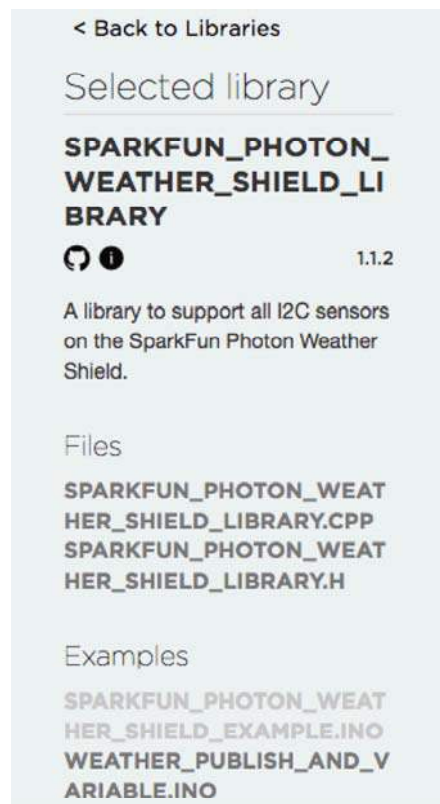
To see data from the Weather Shield, you need to upload a sketch to the Photon. Luckily, there's a library with an example sketch to get you started collecting weather data. To get to the example, click on the libraries button in the Particle IDE.



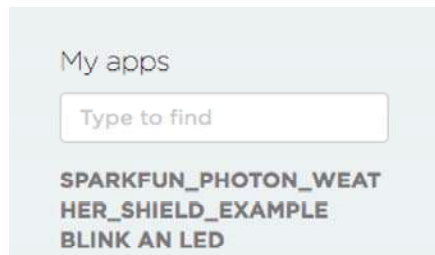
In the library search bar, type the word **weather**.



Click on the **SparkFun_Photon_Weather_Shield_Library**. You should see the library files (.cpp and .h) listed. Below the Examples heading, you should see a couple example .ino files. Click on the one that says **SparkFun_Photon_Weather_Shield_Example**.



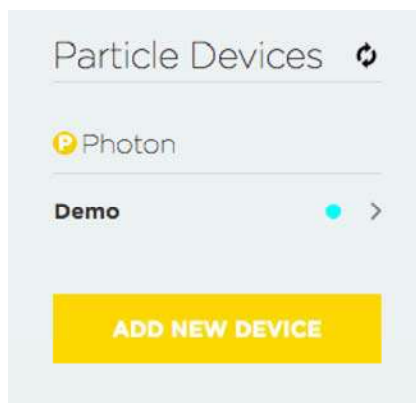
Once clicked, you should see a button that says "Use this Example". You should then see the sketch under the My apps heading.



We now need to select the device to which to upload code. Click the devices icon in the IDE.



If you have just one Photon claimed, it will be the only device to choose from. If you have more than one Photon, you will need to select the star next to the device you want to program.



With the example open, click the verify button to make sure the code compiles correctly.



Once verified, click the upload button to program the Photon.

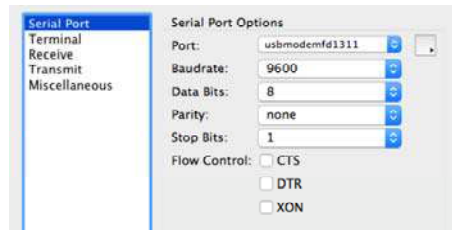


You should see the Photon's RGB LED blink magenta for a few seconds indicating it's being programmed.

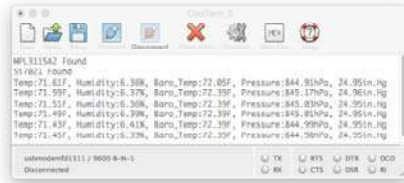
Firmware Updates: If this is the first time you've uploaded code to this Photon or if you haven't used this Photon in awhile, it may go through an automatic firmware update. These typically take less than a minute and are indicated by the RGB LED blinking magenta very sporadically. Once complete, your firmware should be uploaded. If your device has any issues during or after the update, hit reset on the Photon, and try again.

Once your code is uploaded, the RGB LED should go back to breathing cyan to indicate it's connected and running.

Now, open your favorite serial terminal program. Connect to the Photon at 9600 baud, and open a terminal.



Once the terminal is connected, press any key to start streaming weather data. You should now see weather data printing out!



From here, the sky is the limit! You can utilize the Photon's Internet connectivity to post your data to all kinds of data collection services. You can also connect any combination of optional peripherals to the Weather Shield to create a custom weather station suited to your needs.

Resources and Going Further

Here are a few links that should help with any further questions you may have about the Photon Weather Shield:

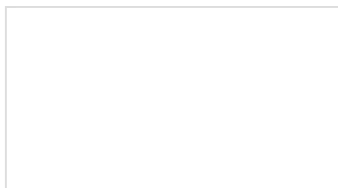
- [SparkFun Photon Weather Shield GitHub Repo](#) – this is where to go for all the example code and hardware files.
- [Photon Weather Shield Schematic](#) – Lots of juicy details found here.
- [Particle Documentation Pages](#) – go here to set up and configure your Photon (or other Particle devices)
- [Particle Community Forum](#) – anything that you couldn't find in the docs should be easily found in the community forum. If you are having trouble, search this forum first, as many of the answers are there already.
- If you would like to use the Photon and the Weather shield with Microsoft's cloud service, Azure, check out this great tutorial by Paul DeCarlo.
- There is a great tutorial on Hackster.io from Dan Fein that goes over how to completely enclose your weather shield and send your data to Wunderground.com.

Using some additional stackable headers, the Weather Shield pairs very well with any of our other Photon Shields. Check out our hookup guides for those shields:



Photon Battery Shield Hookup Guide

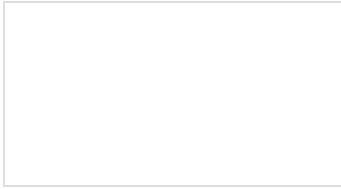
The Photon Battery Shield has everything your Photon needs to run off, charge, and monitor a LiPo



Photon Wearable Shield Hookup Guide

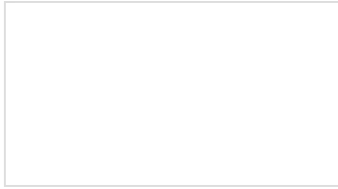
Learn how to use the Photon Wearable Shield for your next projects!

battery. Read through this hookup guide to get started using it.



Photon OLED Shield Hookup Guide

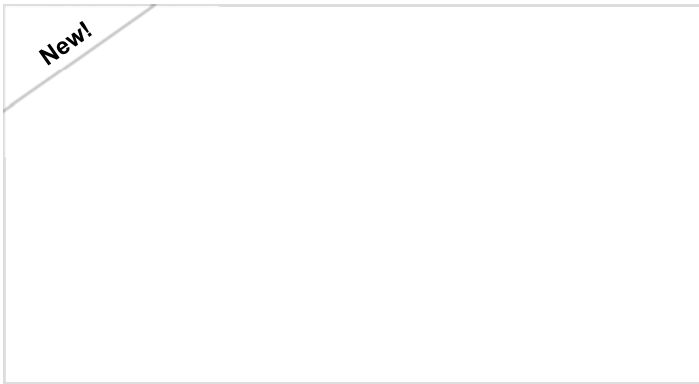
The Photon OLED Shield has everything you need to add a small yet crisp OLED screen to your Photon projects. This hookup guide will show you how to get started.



Photon IMU Shield Hookup Guide

Learn how to use the SparkFun Photon IMU Shield for your Photon device which houses an on-board LSM9DS1 system-in-a-chip that houses a 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer.

If you want to collect weather locally rather than sending it to the web, check out the Arduino Weather Shield Hookup Guide.



Arduino Weather Shield Hookup Guide V12

MARCH 2, 2017

Read humidity, pressure and luminosity quickly and easily. Add wind speed, direction and rain gauge for full weather station capabilities.