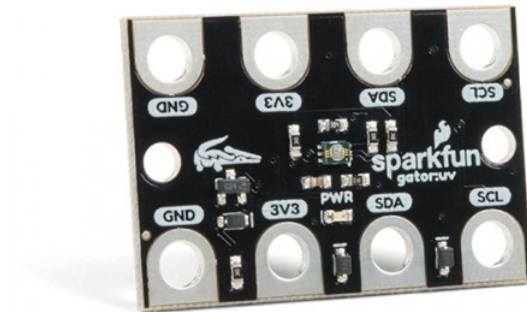


# SparkFun gator:UV Hookup Guide

## Introduction

Do you have a science experiment involving weather or UV light exposure?



## SparkFun gator:UV - micro:bit Accessory Board

© SEN-15273

The gator:UV is the perfect tool to monitor the UV exposure in your next experiment. With the VEML6070, the gator:UV can easily be used to measure the levels of UVA (320-400 nm) radiation. This tutorial will show you how to get started using this UV sensor with the gator:bit (v2) in the micro:bit development environment.

Product Showcase: SparkFun gator:log, gator:RTC, & gator:UV





## Required Materials

To get started, you'll need a micro:bit to control everything. You can purchase one individually or in a kit.



SparkFun Inventor's Kit for micro:bit

● KIT-15228



micro:bit Go Bundle

● DEV-14336



micro:bit Board

● DEV-14208

To easily use the gator board ecosystem, a gator:bit (v2) will help breakout the necessary pins and you will also need alligator and/or banana cables to connect the gator:bit to the gator:UV.



SparkFun gator:bit v2.0 - micro:bit Carrier Board

○ DEV-15162



Alligator Test Leads - Multicolored (10 Pack)

● PRT-12978



## Banana to Banana Cable - Right Angle

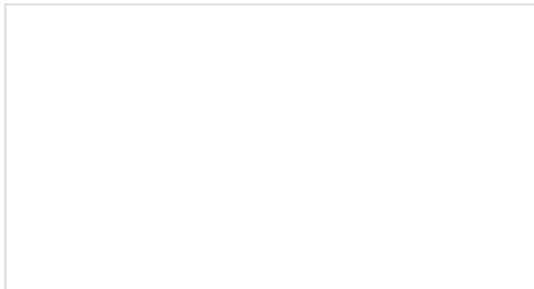
© CAB-15368

*(\*These banana cables have a special diameter on the attachment points designed specifically for use with the micro:bit ecosystem. They may or may not be compatible with the banana cables used on your test equipment.)*

You may already have some of these materials, so feel free to modify your cart as necessary.

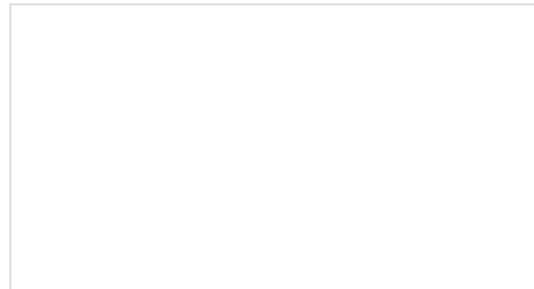
### Suggested Reading

The gator:UV sensor is pretty straight forward to use in application. However, you may find the following concepts useful along the way.



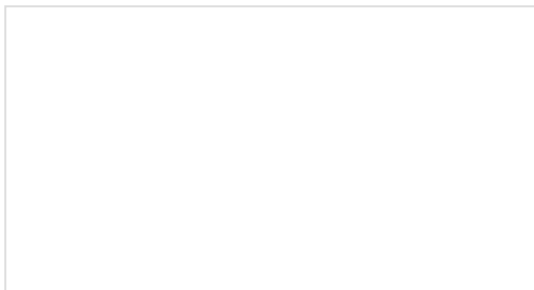
#### Light

Light is a useful tool for the electrical engineer. Understanding how light relates to electronics is a fundamental skill for many projects.



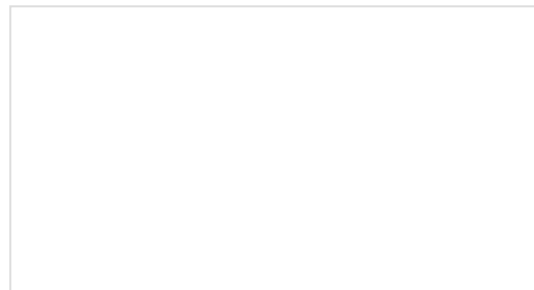
#### Logic Levels

Learn the difference between 3.3V and 5V devices and logic levels.



#### I2C

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



#### micro:climate Kit Experiment Guide

A weather station kit that is built on top of the inexpensive, easy-to-use micro:bit and Microsoft

MakeCode.

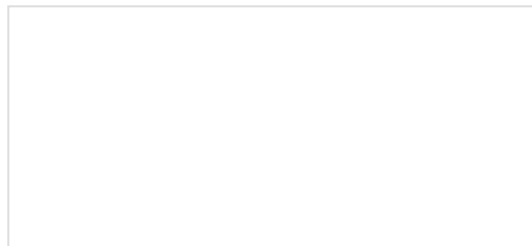
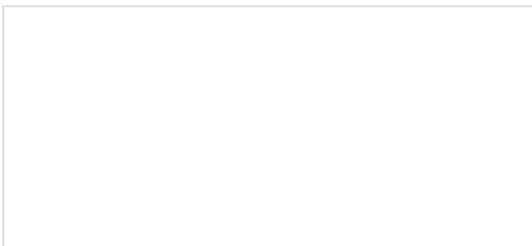
If you aren't familiar with the micro:bit, we recommend reading here for an overview.

## Product Showcase: micro:bit and SparkFun bits



*micro:bit Ecosystem*

We would also recommend taking a look at the following tutorials if you aren't familiar with them.

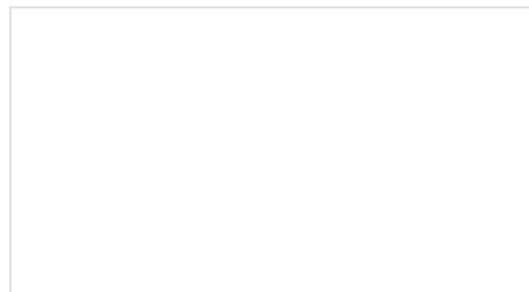
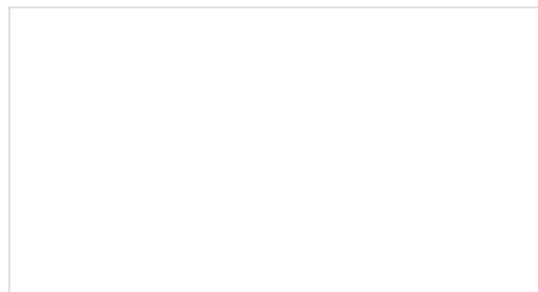


## Getting Started with the micro:bit

The BBC micro:bit is a compact, powerful programming tool that requires no software installation. Read on to learn how to use it YOUR way!

## Getting Started with MicroPython and the SparkFun Inventor's Kit for micro:bit

Learn MicroPython with the micro:bit.



## How to Load MicroPython on a Microcontroller Board

This tutorial will show you how to load the MicroPython interpreter onto a variety of development boards.

## SparkFun gator:bit v2 Hookup Guide

The gator:bit v2 is a breakout board for the BBC micro:bit. The gator:bit exposes almost every pin on the micro:bit to clippable pad with circuit protection. It also has as built-in addressable LEDs and a built-in buzzer.

## Hardware Overview

The gator:UV consists of 4 pads for power and data.

Contacts	Direction	Description
GND	N/A	Ground: Reference voltage and ground loop.
3.3V	In	Power: Provides <b>3.3V</b> to sensor.
SDA	Bi-directional	Data: Data and commands are transmitted between the sensor and microcontroller.
SCL	In	Clock: The microcontroller provides the timing needed to communicate on the data line.

### Power

The specified operating voltage for the VEML6070 is between **2.7V - 5.5V**. For use with the gator:bit (v2) and micro:bit, you should provide **3.3V** through the **3V3** and **GND** pads to keep the logic levels consistent.



*Power connection pads and power indication LED.*

## VEML6070

The VEML6070 is a UV sensor that communicates over an I<sup>2</sup>C bus. The VEML6070 detects UVA radiation up to **328 mW/cm<sup>2</sup>** with a resolution of **5 μW/cm<sup>2</sup>**. For the most part, users will only need to know I<sup>2</sup>C addresses to prevent address conflicts with other devices or sensors.



*VEML6070 UV sensor.*

Here are some of the characteristics of the VEML6070 sensor from the datasheet:

Characteristic	Range
Operating Voltage	<b>2.7V to 5.5V</b>
Supply Current	240 μA
UVA Detection	up to 328 mW/cm <sup>2</sup>
UVA Resolution	5 μW/cm <sup>2</sup> /step
I <sup>2</sup> C Address	0x70 - Write Commands 0x73 - Read MSB of UV Data 0x71 - Read LSB of UV Data

TYPICAL PERFORMANCE CHARACTERISTICS ( $T_{amb} = 25^{\circ}\text{C}$ , unless otherwise specified)

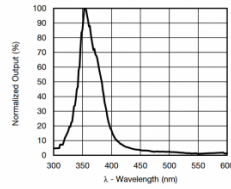


Fig. 4 - Normalized Spectral Response

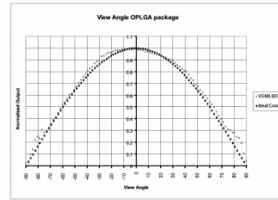


Fig. 5 - Normalized Output vs. View Angle

*Performance characteristics from the datasheet. Click to enlarge.*

## What is UVA Radiation?

UVA radiation is a specific part of the spectrum of the solar radiation from the Sun. UVA light has wavelengths between 320 nm and 400 nm. Approximately, 4.9% of all solar radiation that makes it to the surface of the Earth is UVA light. UVA rays are often associated with photoaging and the formation of skin cancer.

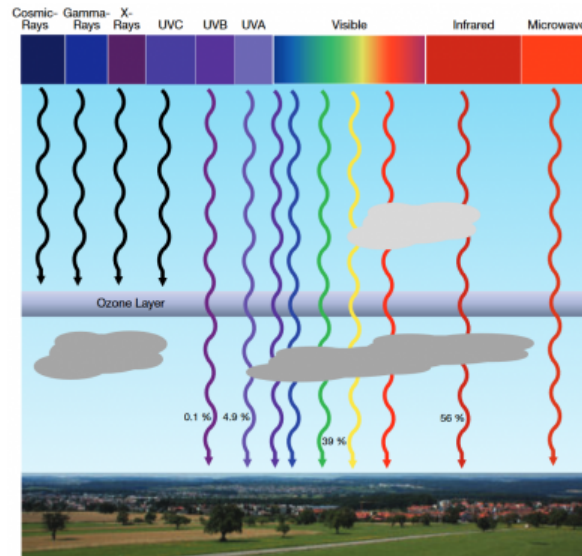


Fig. 5 - Radiation that Reaches the Earth's Surface

*Penetration of solar radiation through Earth's atmosphere (from VEML6070 App. Note). Click to enlarge.*

For more details on UV radiation, check out these online resources:

- NASA Science: Ultraviolet Waves
- Stanford Solar Center: UV Light

Some experiments and related topics for UVA radiation:

- Absorption and Exposure:
  - Sunblock
    - What Is the Most Effective Sunscreen?
    - U.S. Sunscreens Get Flunking Grade for UVA Protection (pub. 2010)
    - The truth about sunscreen (pub. 2014)
  - Photoexposure:
    - Sun Sensitive Paper
    - Ben from the Applied Science has a cool video on UV exposure.
- Effects on insects (like flies or bees):
  - UV Radiation as an Attractor for Insects
  - Insect Vision: Ultraviolet, Color, and LED Light
  - Image processing helps solve the age-old mystery of "Why do zebras have stripes?"

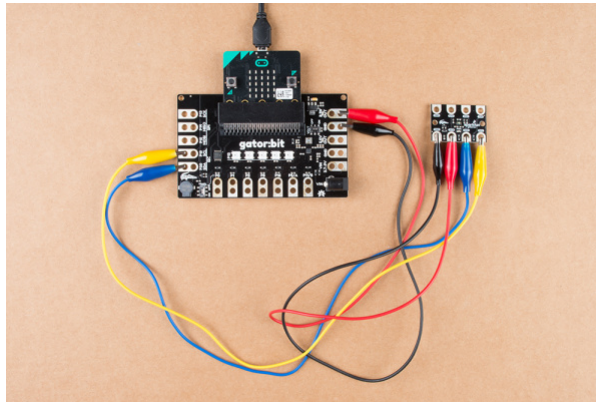
I<sup>2</sup>C is a communication protocol used to transfer data between *master* and *slave* devices. It is a 2-wire connection consisting of SDA (data) and SCL (clock). The protocol is pretty well defined so it can be shared with multiple devices. This is beneficial for daisy chaining multiple devices together on a single bus. The primary thing users will need to watch out for is address conflicts (you can't have devices with the same address).



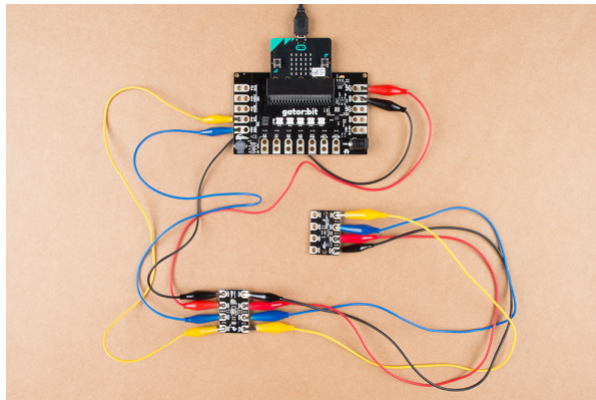
*I<sup>2</sup>C connection pads.*

## Hardware Assembly

Connecting your gator:UV to the to the gator:bit (v2) is simple. The board can also be daisy-chained with other I<sup>2</sup>C boards. This can easily be done with alligator cables or these *special* banana cables.



*Hardware assembly of connections used in example. [Click to enlarge.](#)*



*Example of daisy chaining multiple I<sup>2</sup>C gator boards. [Click to enlarge.](#)*

<b>gator:UV</b>	GND	3V3	SDA	SCL
<b>gator:bit (v2)</b>	GND	3.3V OUT	P20 (SDA)	P19 (SCL)

## Adding the MakeCode Extension



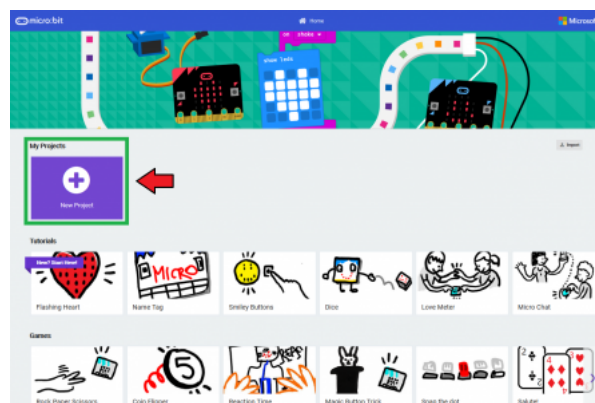
**Note:** This tutorial assumes you are familiar with MakeCode, the gato:bit (v2), and the micro:bit board.

- If you have not previously used **MakeCode**, please check out the Quick Start Guide fro micro:bit.
- If this is your first time using a **micro:bit**, please review our Getting Started with the micro:bit guide.
- If this is your first time using the **gato:bit (v2)**, please review our SparkFun gator:bit v2 Hookup Guide.

The easiest way to get started using the gator:UV is to use Microsoft MakeCode, a web-based block editor. This tutorial assumes you are familiar with the with MakeCode, the gato:bit (v2), and the micro:bit development board. If this is your first time check out this guides linked in the suggested reading section (above).

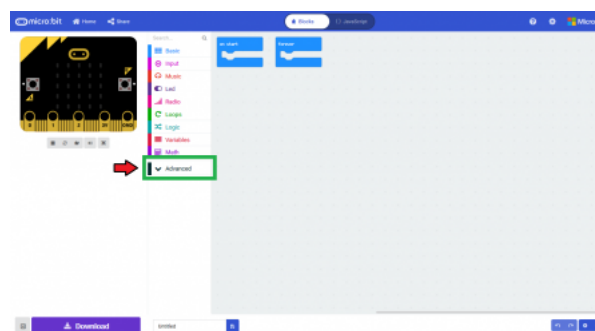
## Installing Extensions

To get started with using MakeCode with the micro:bit, head over to the MakeCode for micro:bit website by Microsoft. Then, click the **New Project** button to start a new project and open the *Editor*. (*\*Yes, all you only need to get start coding is a computer with internet access, an up-to-date web browser, and an available USB port!*)




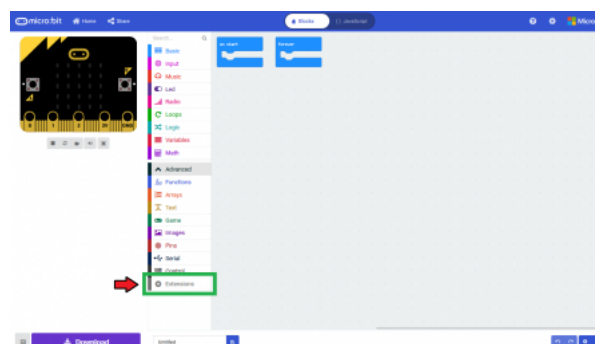
Click the **New Project** button to start a new project and open the Editor. Click on image to enlarge.

Once you have the editor up, click on the the **Advanced** block in the block library to reveal the drop down menu.



Click on the the **Advanced** block in the block library to reveal the drop down menu. Click on image to enlarge.

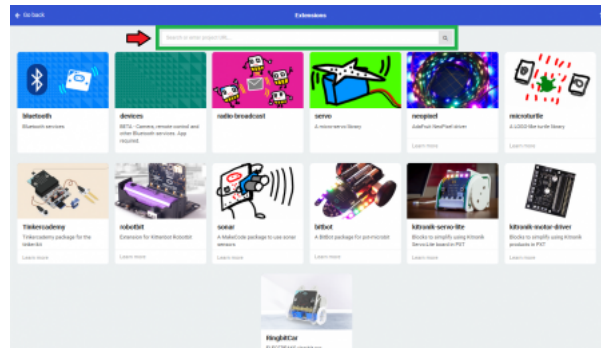
Finally, click on the **Extensions** block. This should bring up the extensions page. (*\*Alternatively, you could also click on the extensions link from the  settings menu.*)



Click on the the **Extensions** block to open the extensions page. Click on image to enlarge.

There are two methods for finding the gator:UV extension:

- Search for the **name** used on the extension<sup>1</sup>.
- Use the link to the GitHub repository for the **pxt-package** as the search term.



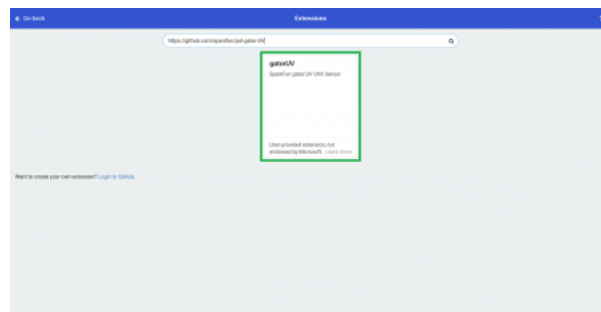
Use the search bar to find the extension you want to install. Click on image to enlarge.

**Note:** Unfortunately, it does take time to get an extension approved by the micro:bit Educational Foundation before it can be searchable by name. Part of the requirements for the approval process involves a [live](#) product page. Therefore, at the time of the launch of this product, the extension has not been approved yet and the only method of adding the extension is to use the link to the GitHub repository of the pxt-package. We will update this tutorial as soon as the extension has been approved.↔

## Search for the PXT-Package

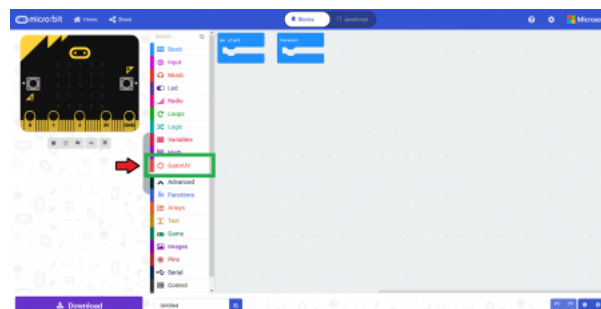
Search for the gator:UV extension using the GitHub repository link to the pxt-package:

<https://github.com/sparkfun/pxt-gator-UV>



Search for Extension. Then, click on the box to add it to the block library. Click on image to enlarge.

Then, click on the box for the extension to add it to the block library. The gator:UV extension should now appear in the block library.



Extension in the block library; click on the block to open menu options. Click on image to enlarge.

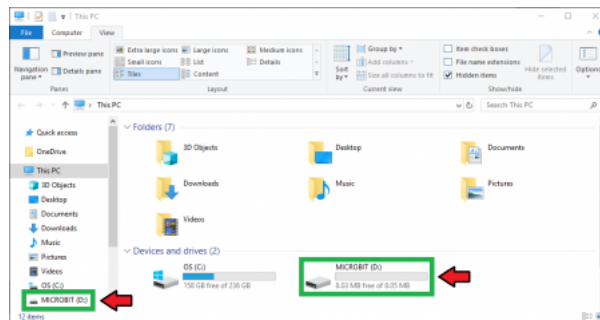
To verify that you have added the extension, click on the `gator:UV` block to make sure that your drop down menu options match the image below.



Available blocks for the `gator:UV` extension. Click on image to enlarge.

## MakeCode Examples

Now that you have added the `gator:particle` extension to the *Editor*, lets start with some example code. Plug the micro:bit into your computer using an USB micro-B cable after you have assembled your hardware with the instructions from the previous section. The micro:bit should appear on your computer as a removable storage device.



The micro:bit showing up as a USB drive on a Windows computer. Click to enlarge.

To upload new code, this is where you will be copying the downloaded `.hex` file to later.

## Block Functions

### **Initialize `gator:UV` sensor**

This block should be pretty self explanatory, it sets the sensor up to be used. You should use this within the `on start` block.

### **get UVA Reading**

This is a value block as indicated by the block's shape. The output of the block is value measured by the UV sensor.

- 0 to 65535 - A 16-bit value of the UV sensor reading that can be correlated to the UV-Index.

### **set refresh time on UV Sensor to \_\_\_\_\_**

This sets the integration time for the UV sensor. The refresh time should be adjusted for the range of values you expect to be measuring.

- half - Half of the normal integration time.
- one - **Normal integration time (default)**.
- two - Twice the normal integration time.
- four - Four times the normal integration time.

**Note:** \*Although not exactly correct, it can be interpreted as the time that the sensor spends collecting a data sample.

If you are barely measuring changes between measurement values, you should try to increase the refresh time, which allows the sensor to collect more light. The relation between measured values and the refresh time are linear (see table below).

## UV Index

The table below maps out UV index to the correlating sensor output values and their respective refresh time (or integration time) setting. The values were extrapolated from the UV index tables provided in the VEML6070 Application Note for an  $R_{SET} = 300k\Omega$ .

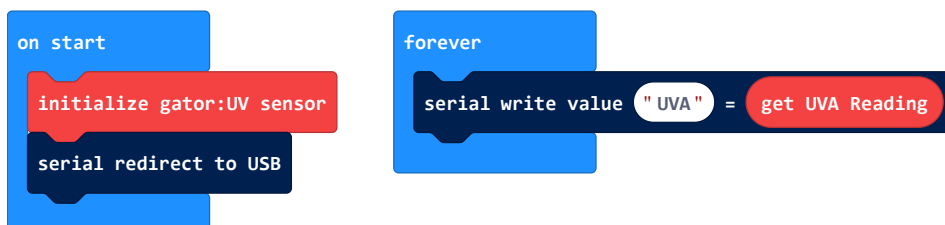
Int. Time = x0.5	Int. Time = x1	Int. Time = x2	Int. Time = x4	UVI (Value)	UV-INDEX
$\geq 1142$	$\geq 2283$	$\geq 4565$	$\geq 9130$	$\geq 11$	<b>Extreme</b>
831 to 1141	1661 to 2282	3321 to 4464	6641 to 9129	8 to 10	<b>Very High</b>
623 to 830	1246 to 1660	2491 to 3320	4981 to 6640	6,7	<b>High</b>
312 to 622	623 to 1245	1245 to 2490	2490 to 4980	3 to 5	<b>Moderate</b>
0 to 311	0 to 622	0 to 1244	0 to 2489	0 to 2	<b>Low</b>

## Basic Reading

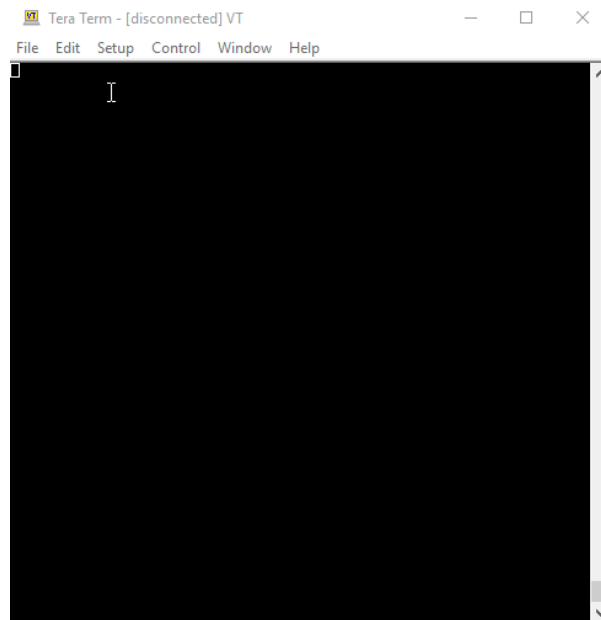
Below, is a simple example of how to take simple measurements from the sensor. To use this example, there are multiple options for accessing the .hex file:

- Replicate the block functions from the display below to copy the example project. Then download the .hex file.
- Expand the display widow to pull up the example project. Then download the .hex file.
- Use this link to pull up the example project. Then download the .hex file.
- Download the .hex file from the button below or the link on the bottom of the display.

**DOWNLOAD .HEX FILE FOR EXAMPLE**

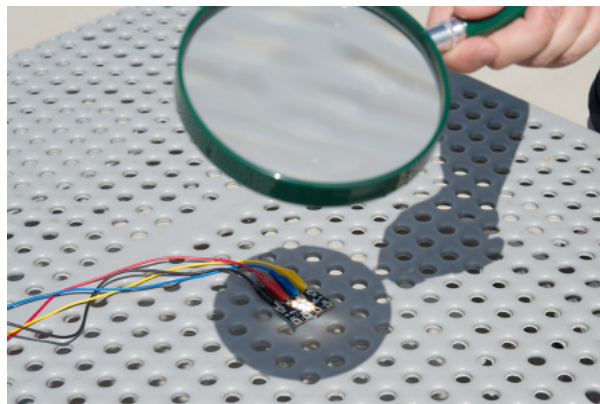


The output is redirected over the serial port to avoid conflicts on pins P0 and P1, which are also used for serial communication. To read the sensor values, pull up your favorite serial terminal emulator. Then, connect to the serial port that the micro:bit is on; the default baud rate is 115200 bps. Below, is an example of the sensor output for the example code (the sensor values are responding to the *flashlight* on my phone).



*Example of readings from the sensor.*

**Do Not Try This at Home:** In case you were thinking about it, do **NOT** try to replicate this image or attempt to max out your readings with a magnifying glass. **You will burn or damaged the board.** I had the magnifying glass focused on the board for less than a second for this picture and it started to burn and smolder.



## Troubleshooting Tips

### Power and Connections

Check that your power and data connections are correct and that the power `vOUT` switch is in the `ON` position on the `gator:bit` (v2).

## UV Filters

Make sure that if you are using a transparent cover over the sensor that it still passes UV light. More modern clear plastics and glass covers (especially for photo or documents) usually have a special UV coating to prevent pigment deterioration and yellowing of paper.

If you still have questions or issues with this product, please create a post on our forum under the `Gator Products` topic.

Additionally, if you are an **educator** and you have class or application based questions, please create a post on our forum under the `Educational Products` topic.

## Resources and Going Further

For more product information, check out the resources below:

- [Schematic \(PDF\)](#)
- [Eagle Files \(ZIP\)](#)
- [VEMML6070 Datasheet \(PDF\)](#)
- [VEMML6070 Application Note \(PDF\)](#)
- [gator:UV PXT Package](#)
- [GitHub Product Repo](#)
- [micro:bit Landing Page](#)
- [SFE Product Showcase](#)

Interested in the `micro:bit`? Check out some of these other `micro:bit` products:



SparkFun `micro:climate` kit for `micro:bit`  
○ KIT-15301



SparkFun `gator:bit` v2.0 - `micro:bit` Carrier Board  
○ DEV-15162

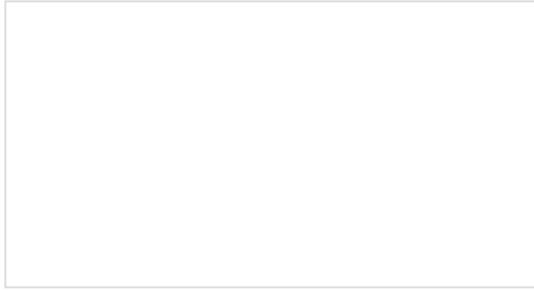


SparkFun Inventor's Kit for `micro:bit` Lab Pack



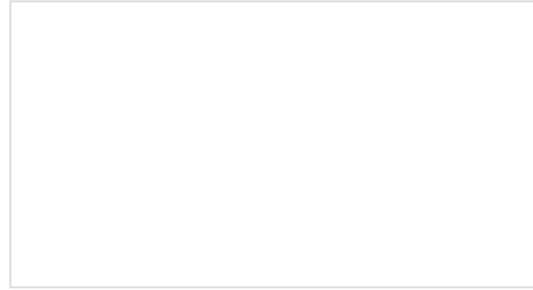
SparkFun Inventor's Kit for `micro:bit`

Need some inspiration for your next project? Check out some of these other micro:bit product tutorials:



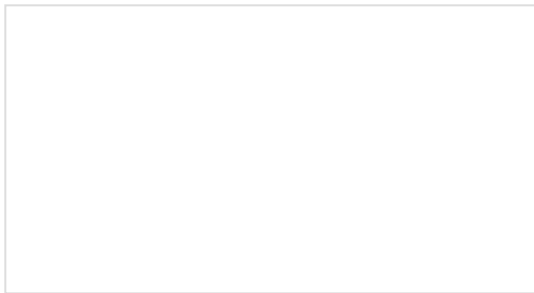
**Gator:starter ProtoSnap Hookup Guide**

Get started clipping sensors and lights to the micro:bit with the gator:starter gator:board from SparkFun!



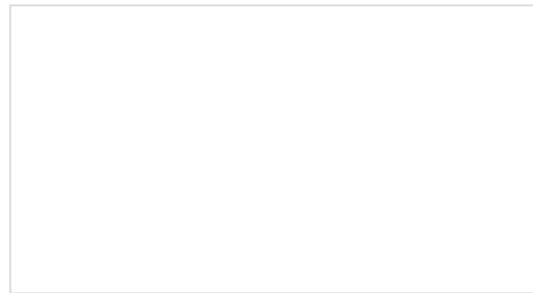
**Gator:control ProtoSnap Hookup Guide**

Buttons and switches and inputs oh my! Start adding more control to your gator:bit with the SparkFun gator:control.



**SparkFun gator:bit v2 Hookup Guide**

The gator:bit v2 is a breakout board for the BBC micro:bit. The gator:bit exposes almost every pin on the micro:bit to clippable pad with circuit protection. It also has as built-in addressable LEDs and a built-in buzzer.



**SparkFun gator:particle Hookup Guide**

The gator:particle is an I2C heart-rate monitor and pulse oximeter that can be used as a particle sensor. This tutorial will get you started using the gator:particle with the micro:bit platform.