

Gator:control ProtoSnap Hookup Guide

Introduction

The gator:control is one of a series of gator-clippable boards called gator:boards that SparkFun has created to interface with the micro:bit and gator:bit expansion for micro:bit. The gator:control contains two buttons, an on/off slide switch, and a reed switch, which is activated by a magnet. In this hookup guide, we'll go over how to hook up each of the individual boards, along with some examples involving all of the boards together.



SparkFun gator:control ProtoSnap
© COM-14968

Required Materials

For this activity, you'll of course need a micro:bit. You'll also need some alligator clips to connect everything together, and a micro-b USB cable to program your micro:bit. A magnet is also a good addition to the mix, as this is the only way to activate the reed switch. All of these things are shown below, so grab them if you haven't already. You can go ahead and grab a gator:bit as well to create some more robust projects, but you'll be able to get along fine with just a micro:bit.



micro:bit Board
○ DEV-14208



SparkFun gator:bit
● DEV-14484



SparkFun gator:color ProtoSnap
● COM-14890



Alligator Test Leads - Multicolored (10 Pack)
● PRT-12978



Magnet Disk - 0.709"



USB Micro-B Cable - 6"

Suggested Reading

If you decide to use the gator:bit and it's your first time using the board, check out the gator:bit Hookup Guide.



Gator:bit Hookup Guide

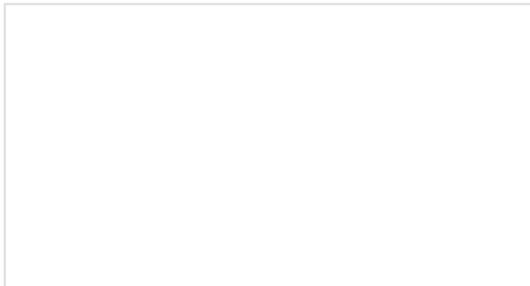
MAY 10, 2018

Gator:bit is a rugged development breakout board for the BBC micro:bit.

Gator:bit exposes almost every pin on the micro:bit to an alligator clippable pad.

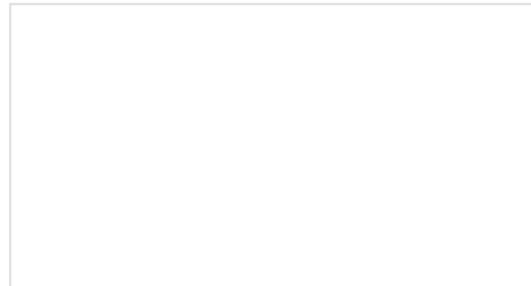
Gator:bit features over-voltage, over-current, and short circuit protection circuitry on every IO pin as well as built-in addressable LEDs, built-in buzzer, and a power management solution to give you sensible powering options.

Also, if you're starting out with electronics and aren't familiar with the following concepts, we recommend checking out these tutorials before continuing.



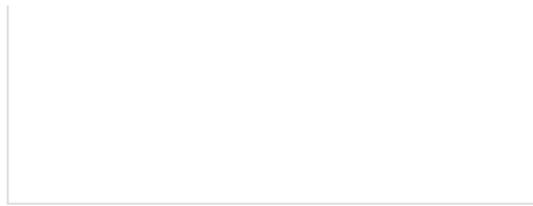
What is a Circuit?

Every electrical project starts with a circuit. Don't know what a circuit is? We're here to help.



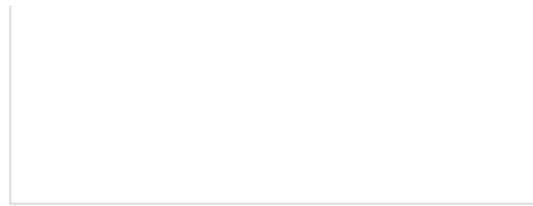
Voltage, Current, Resistance, and Ohm's Law
Learn about Ohm's Law, one of the most fundamental equations in all electrical engineering.





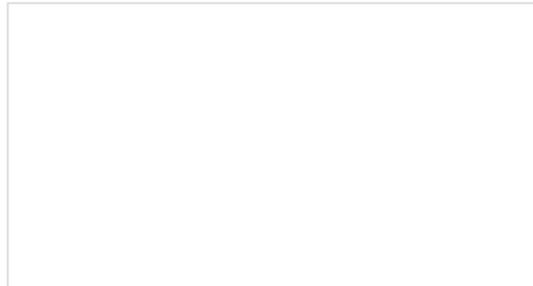
What is Electricity?

We can see electricity in action on our computers, lighting our houses, as lightning strikes in thunderstorms, but what is it? This is not an easy question, but this tutorial will shed some light on it!



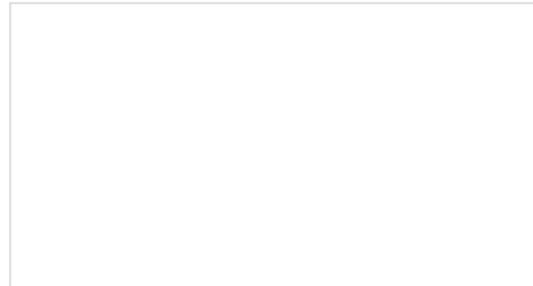
Light-Emitting Diodes (LEDs)

Learn the basics about LEDs as well as some more advanced topics to help you calculate requirements for projects containing many LEDs.



Analog vs. Digital

This tutorial covers the concept of analog and digital signals, as they relate to electronics.



Getting Started with the micro:bit

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

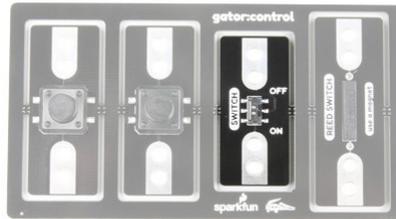
Hardware Overview

The gator:control contains four boards in the main assembly, all of which can be broken out of the main board for individual use. Buttons or switches allow you to create a point in a circuit that can either be closed (connected) or open (disconnected). Most buttons, including the ones on the gator:control, will be a closed circuit only when they are being pressed. These buttons on the gator:control are highlighted below.



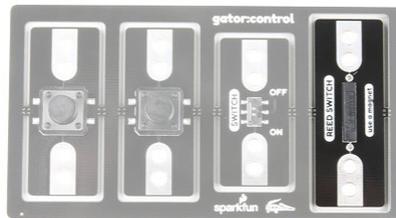
Buttons

The second type of control element on the gator:control is the slide switch. This will be a closed circuit when the switch is in the **ON** position and an open circuit when the switch is in the **OFF** position. The slide switch is highlighted on the gator:control below.



Slide Switch

The final type of control element is the reed switch, which is controlled by a magnet. The reed switch will act as a closed circuit when you bring a magnet sufficiently close to the switch. The reed switch is shown highlighted in the image below.



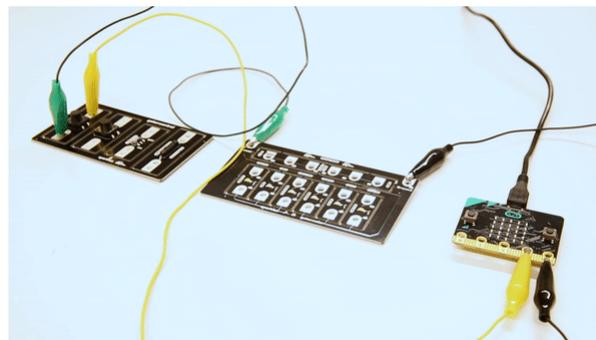
Reed Switch

Examples

There are two ways that you can use these buttons or switches in your circuit. One would be as a part of a simple circuit, allowing the button or switch to complete and close your circuit, the other would be to use the button or switch as a digital input to control an already completed and closed circuit. We'll demonstrate these two ways to use buttons and switches by turning on an LED. For this example, we'd recommend using a gator:LED from the gator:color because you won't have to worry about a resistor.

Simple Button

In the first circuit, we'll connect a button so that it is a part of the LED circuit. We will complete the circuit by pressing down on the button. This will close the circuit so that current will flow through our button and power the LED. As soon as you stop pressing on the button, the circuit is open causing the LEDs to not light up. To set this circuit up, connect the 3V pin on your micro:bit to one side of a button, connect the other side of the button to the + side of a gator:LED, then connect the - side of the gator:LED to the ground pin on the micro:bit. The circuit should look like the below image. The connections can be seen by matching the colors of the alligator clips.

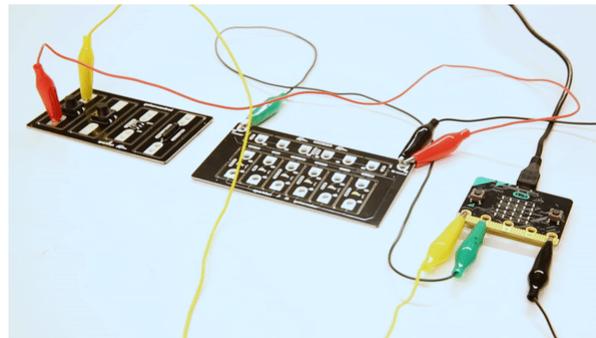


Simple Button Circuit

Clicking the button should turn the light on if you've set everything up properly. Instead of using the button, try using the slide switch to control the LEDs. Flipping the switch to the ON position is the same as pushing the button down. When we turn the switch to ON position, we complete the circuit by closing. Flipping the switch to the OFF position will open the circuit back up causing the LEDs to not light up. You should see the same effect with the reed switch in place of the button. To complete the circuit, place a magnet above the reed switch to close the circuit. Removing the magnet will open the circuit back up.

Digital Button

In the second circuit, we will connect our button to a digital input and our LED circuit to a digital output. Then, when we click the button, the micro:bit will see a digital signal, and it will toggle the pin for the LED circuit on. To connect this circuit, connect one side of a button or switch to the `ground` pin, and the other side of the button to pin `0`. Now connect pin `1` to the `+` end of the gator:LED and the `-` side of the gator:LED to the `ground` pin on the micro:bit. Your circuit should look something like the image below.



Digital Button Circuit

You may have noticed that we connect one end of the button to `ground`. This means that when we push the button, pin `0` will connect to ground and the micro:bit will read a digital `low` or `0` signal. However, when these digital pins are at rest, they're low, so we'll need to configure the micro:bit so that this pin rests in a high state. To do this with code, we add a pull-up resistor to pin `0` using the following block of code, located under **Pins** -> **More**

```
-> set pull pin P0 to up.
```

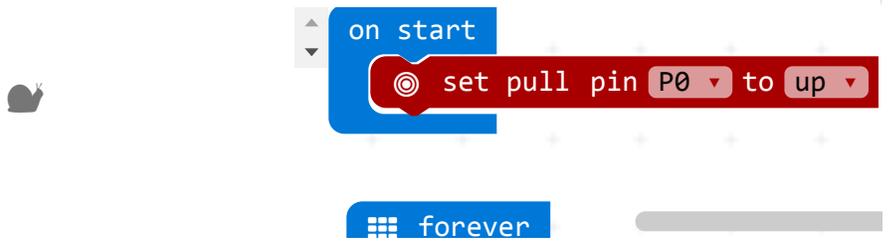


Set Pull-Up

This block will make it so that pin `0` will be a `high` logic level until we push our button, pulling the pin low. Now we need to have the micro:bit read the button pin to see if the button is pressed, then, using that information, decide on whether or not to turn on the light. To do this, we read the value of pin `P0` into a variable named `button_pin`. From the **Variables** and **Pin** category, set the blocks of code to `set button_pin to digital read pin P0`.

Then we use an `If...Else...` loop to check the value of `button_pin`. If the value is `0`, then that means the button is pressed, so we write pin `P1` `high`. Otherwise, we write a `low` value to `P1`. Re-create the following code into your MakeCode editor or download the example by clicking the download button to test it out!

 This site uses cookies for analytics, personalized content and ads. By continuing to browse this site, you agree to this use. [Learn more](#)



Note: You may need to disable your ad/pop-up blocker to interact with the MakeCode programming environment and simulated circuit!

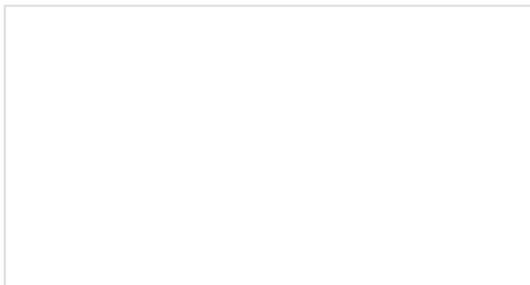
Go ahead and give your button a click. The light will still turn on, but unlike the previous example, we've done this action digitally! The example code will also work with the slide switch and reed switch. Try replacing the connection with either switch to control the LEDs.

Resources & Going Further

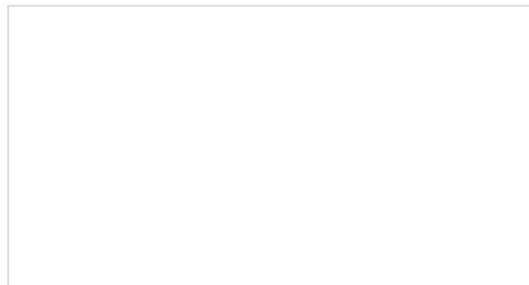
Now that you've successfully got your gator:control up and running, it's time to incorporate it into your own project! For more information about the gator:control board, check out the resources below:

- [Schematic \(PDF\)](#)
- [Eagle Files \(ZIP\)](#)
- [About micro:bit Page](#)
- [micro:bit Programmer](#)
- [GitHub Product Repo](#)
- [SFE Product Showcase](#)

Need some inspiration for your next project? Check out some of these related tutorials:



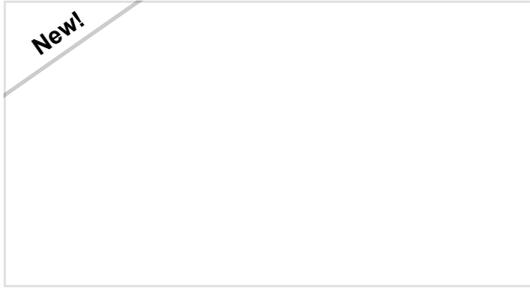
[micro:bit Educator Lab Pack Experiment Guide](#)



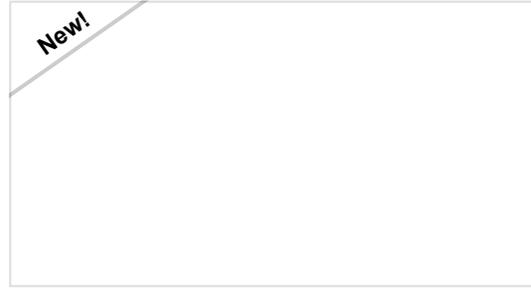
[micro:climate Kit Experiment Guide](#)

A quickstart guide for the micro:bit educator lab pack.

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



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



Gator:color ProtoSnap Hookup Guide
Clip some LED's onto your gator:bit with the gator:color.