



Using the SparkFun PicoBoard and Scratch

Introduction

Scratch is an amazing tool to teach kids how to program. Often, we focus on creating fun animations, games, presentations, and music videos in Scratch. But, imagine how cool it would be if you could add physical interactivity into your Scratch animation. For example, what if your Scratch character started to dance when you turned off the lights in your room or stopped as soon as the lights turned back on! With the PicoBoard, we can do all of this and more.



What is the PicoBoard?

The PicoBoard is a piece of hardware called a sensor board that can be combined with MIT's Scratch programming environment to allow your Scratch programs to react (and even respond) to events happening outside of the computer. (If you're unfamiliar with Scratch and would like to learn more about it, read the Science Buddies Introduction to Scratch page.) A sensor is a device that detects (senses) and measures the presence or absence of something. The PicoBoard is actually made up of several different types of sensors so it can detect many different "somethings," including sound and light.

The PicoBoard has several sensors that can be used directly with the Scratch programming environment:

Slider: Changes values from 0 to 100 based on the position of the slider.

Light Sensor: Changes values from 0 to 100 based on the amount of light on this sensor.

Button: The position or state of the button controls the button pressed value (true or false).

Sound Sensor: Changes values from 0 to 100 based on the amount of sound it detects.

Auxiliary Connections (A,B,C,& D): Alligator Clips: These are generic connections to any resistive sensor. As the resistance changes, so does the input value. This sensor can also be setup as a digital detector to detect whether the alligator clips are connected.



Download Scratch 1.4

Note: Currently, the PicoBoard is only compatible with Scratch 1.4.

To get started, you will need to install Scratch 1.4 for your computer. Visit scratch.mit.edu/scratch_1.4/ to download version 1.4 of Scratch.

At this moment, the interface between Scratch 2.0 and the PicoBoard is currently under development. We are still working out the kinks of this. We recommend using the older and stable (v1.4) version of Scratch for interfacing to the PicoBoard.

Installing Drivers

In addition to installing Scratch, you will need to install the FTDI drivers for your computer. The FTDI driver will allow the PicoBoard hardware to communicate with your computer. Follow our guide here on how to install and setup the drivers for your machine.

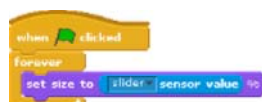
Connecting the PicoBoard

Using the USB cable, connect the PicoBoard to your computer, and open up Scratch. To test the functionality, let's play with the Scratch sprite's size feature using this block. Find these blocks and string them together. Move the (slider sensor value) block into the set size block to replace the size by the sensor value.

Find the "Green Flag" Hat block and forever block under the **Control palette**, the set size block under the **Looks palette**, and the sensor value block under the **Sensing palette**.



Assemble the blocks together so that it looks like this:



Click the green flag or the “hat” block to start your program. In a few seconds, you should see the red / green lights on your PicoBoard start flashing. Move the slider back and forth and watch what happens!



The red / green lights indicate data is being received (RX) and transmitted (TX) between the PicoBoard and your computer.

The Sensor value reporter block can be used in place of any value field (blank / number) in Scratch. Experiment with any of the other Scratch blocks such as *move () steps* or *go to x: () y: ()*. Any of these values can be replaced by the sensor value from the PicoBoard.

Find your favorite Scratch program and re-mix it using the Sensor blocks!

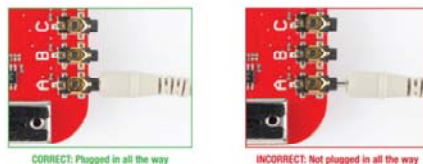
Using Additional Sensors (PicoBoard Add-on / LabPack)

The PicoBoard LabPack comes with several add-ons to attach the the auxiliary ports.

How to Add Sensors to the PicoBoard

The four additional ports (A-D) allow the PicoBoard to be connected to a number of other additional sensors. The PicoBoard measures the resistance across the two alligator clips. When plugging in the PicoBoard connectors, make sure that you push the connector all the way in.

- Plug an alligator clip into one of the PicoBoard end connectors. There are four end connectors. They are A, B, C, or D.



The PicoBoard measures the resistance between the two alligator clips of the auxiliary ports. You can use any resistive sensor with these clips. We’ve created an add-on kit of parts that you might find useful to play with. In the add-on kit, we have:

Button / Switches (Digital Sensors)

The most basic “sensor” is a simple switch. Switches allow us to electrically connect or disconnect a path for electricity between two metal connections. The PicoBoard senses whether or not the two alligator clips are connected.

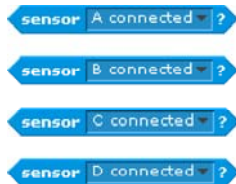
In the Add-on kit, we include two arcade switches.



These switches are pretty neat. They use a momentary limit switch that has three connections to it. Connect one of the alligator clips to the inner-most tab. Connect the second alligator clip to the outer-most connection tab.



The switches can be triggered in Scratch using the boolean sensor block:



These blocks can be used with control blocks such as:



Analog Sensors

All other resistive sensors are considered “analog” sensors. The PicoBoard will measure the resistance of the sensor and report a value that ranges from 0 to 100 in Scratch. These values can be accessed using any of these blocks for the auxiliary sensors on A, B, C, or D.



Turn / Rotary Potentiometer

A rotary potentiometer is basically the same as the slider – however, instead of moving the slider back and forth, you simply turn the knob on the potentiometer. Connect one of the alligator clips to the center tab and the second alligator clip to either one of the outside tabs. Note: The limits of this sensor will only vary from 0 to 50 (instead of 0 to 100).



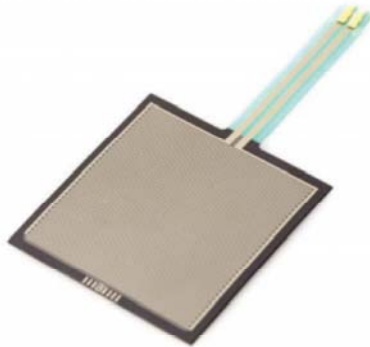
Motor

Traditionally, a motor is an output device, but when connected to the PicoBoard, you can use the motor as a generator. Spin the motor and see what happens to your input values. Simply connect the two wires of the motor to the two alligator clips.



Force Sensitive Resistor

This square sensor senses pressure applied to the surface. As you press down on the square, the resistance changes.



Attach the alligator clip extensions (regular alligator clip wires) onto the two metal prongs on the end of the Force Sensitive Resistor.



When clipping, make sure that the metal clips are fully seated and touching the metal part of the sensor. Also, make certain that the alligator clips do not touch each other.

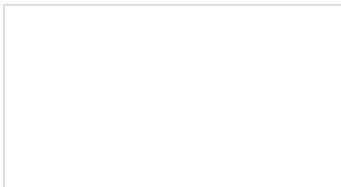


Thermistor

A thermistor is a special type of resistor whose resistance varies significantly with temperature. As the temperature increases, the resistance of a thermistor decreases. Technically, the thermistor requires the use of the Steinhart-hart equation to convert from the PicoBoard sensor values to a real temperature but, you can simply connect the thermistor up using the alligator clips.



If you're interested in a guide to calibrating a thermistor in Scratch, check out our blog post on using a standard turkey thermometer in Scratch.



**Turkey Scratch --
Thanksgiving, Programming,
and a 'lil Data**

Quick Reference Guide

Now that you have the PicoBoard set up correctly, you're ready to start playing with it. The sensor table describes each sensor, how you can use it, and the Scratch blocks that you could use to capture and use the sensor information (Click the image for a larger view).

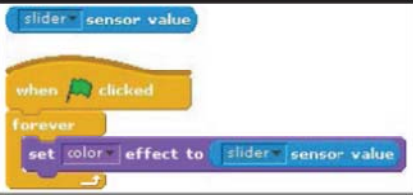
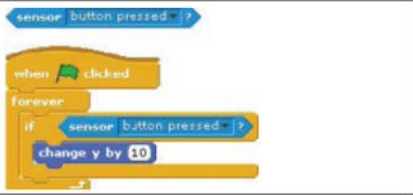
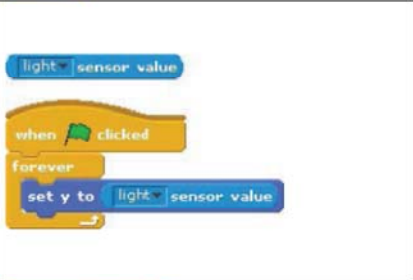
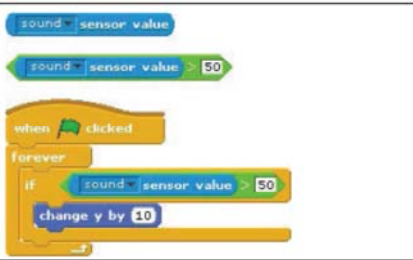
Sensor	Action	Scratch Blocks
Slider	You've already played around with this sensor. It is what engineers call a slide potentiometer! This sensor is also called a variable resistor. This sensor changes continuously on a scale from 0 to 100. Instead of changing Scratch's view by using the "set project effect block" - this is also under the Looks Palette.	
Button	The button can be in only one of two positions - either up or down. Notice that the shape of this sensor block is different! It's a light / button block. Your sprite can be programmed to react when you push the button. Make your sprite jump or change colors. How might you change this so that the sprite jumps when the button is pressed?	
Light	The light sensor has an "eye" icon next to it. Like the slider, this sensor value varies from 0 - 100 depending on the amount of light available. Just like with the slider, you can use this sensor block to replace any numeric value. For example, make your sprite react to light or shade using the data coming from this sensor, or use it as a "game controller" in a game like pong.	
Sound	Use this sensor to program your sprite to react when this sensor detects sound. This sensor also varies from 0 - 100. It picks up the vibrations in the air and converts it to an electrical signal. Can you make your sprite jump up to surprise if you yell into the sensor? Or, perhaps blink in size if you simply whisper into the sensor.	
A, B, C, D	Free for all. There are four connections on the end of the PicoBoard. Connect just about any variety of sensors using the alligator clips. These inputs monitor the resistance between the alligator clips. Connect these things up to flex sensors, temp sensors, or just connect them up to pieces of aluminum foil to make your own switch!! With this example code, be mindful of those around you - this can make a lot of noise -- Turn down your volume, and use headphones, if you have them.	

Resources and Going Further

You're not limited to just the sensors that we include in the add-on kit. Experiment around with seeing what things are conductive and what things are not! Because the alligator clips measure resistivity (conductance), you can use this as a sensor for water salinity or conductivity of solutions.

Check out these other educational SparkFun tutorials.

- DigitalSandbox PicoBoard
- Digital Sandbox Experiment Guide
- LogicBlocks & Digital Logic Introduction
- LogicBlocks Experiment Guide
- The Great Big Guide to Paper Circuits

Sensor	Action	Scratch Blocks
Slider	You've already played around with this sensor. It is what engineers call a slide potentiometer. This sensor is also called a variable resistor. This sensor changes continuously on a scale from 0 to 100. Instead of changing Scratch's size, try using the "set [color] effect block - this is also under the Looks Palette.	
Button	The button can be in only one of two positions - either up or down. Notice that the shape of this sensor block is different? It's a logic / boolean block. Your sprite can be programmed to react when you push this button. Make your sprite jump or change colors. How might you change this so that the sprite jumps when the button is pressed?	
Light	The light sensor has an "eye" icon next to it. Like the slider, this sensor value varies from 0 - 100 depending on the amount of light available. Just like with the slider, you can use this sensor block to replace any numeric value. For example, make your sprite react to light or shade using the data coming from this sensor, or use it as a "game controller" in a game like pong.	
Sound	Use this sensor to program your sprite to react when this sensor detects sound. This sensor also varies from 0 - 100. It picks up the vibrations in the air and converts it to an electrical signal. Can you make your sprite jump up in surprise if you yell into the sensor? Or, perhaps shrink in size if you simply whisper into the sensor.	
A, B, C, D	Free-for-all. There are four connections on the end of the PicoBoard. Connect just about any variety of sensors using the alligator clips. These inputs monitor the resistance between the alligator clips. Connect these things up to flex sensors, temp sensors, or just connect them up to pieces of aluminum foil to make your own switch! With this example code, be mindful of those around you - this can make a lot of noise -- Turn down your volume, and use headphones, if you have them.	