



# Soil Moisture Sensor with Circuit Playground Express and MakeCode

Created by Anne Barela



<https://learn.adafruit.com/soil-moisture-sensor-with-circuit-playground-express>

Last updated on 2024-06-03 02:22:03 PM EDT

# Table of Contents

<b>Overview</b>	<b>3</b>
<ul style="list-style-type: none"><li>• How It Works</li><li>• Pick Your Plant</li></ul>	
<b>Build</b>	<b>5</b>
<b>Code</b>	<b>6</b>
<ul style="list-style-type: none"><li>• For Windows</li><li>• Manual Calibration, All Platforms</li><li>• Final Setup</li></ul>	
<b>Going Further</b>	<b>10</b>
<ul style="list-style-type: none"><li>• Assistive Technology</li></ul>	

---

# Overview



These days, we all get busy and soon a week (or month) has gone by. Then we remember: water the plants! Run over to the plants and the soil is like a desert and we apologize to the poor plant.

This situation can be averted through a simple project using the [Adafruit Circuit Playground Express](http://adafru.it/3333) (<http://adafru.it/3333>). The amount of moisture in the soil can be measured by the board and indicators used to show if the plant needs watering.

All you need is a nail and an alligator clip cable. Adafruit carries a kit with the materials or you can buy the pieces separately.



## Circuit Playground Express Soil Sensor Mini Kit

The thirst is real! Keep your flora (plants, not the board \*wink\*) happy and hydrated with the Circuit Playground...

<https://www.adafruit.com/product/4060>

Separately:

### 1 x [Circuit Playground Express](https://www.adafruit.com/product/3333)

<https://www.adafruit.com/product/3333>

Circuit Playground Express Board

---

### 1 x [3 x AA Battery Holder](https://www.adafruit.com/product/3287)

<https://www.adafruit.com/product/3287>

Battery holder with On/Off Switch and JST connector to CPX

---

### 1 x [Alkaline AA Batteries](https://www.adafruit.com/product/3521)

<https://www.adafruit.com/product/3521>

3 packaged together

---

### 1 x [Small Alligator Clip Test Lead](https://www.adafruit.com/product/4100)

<https://www.adafruit.com/product/4100>

Set of 6

---

### 1 x **Nail or other piece of conductive metal**

2 to 4 inches (5-10 centimeters)

---

A nail is a common piece of metal to use but it could be any metal that isn't too broad that can stick into the soil of the pot with a decent depth. The metal/nail does not need to reach the bottom of the pot, only be an inch or two (2.5-5 cm) down in the soil.

## How It Works

The Circuit Playground Express can detect the capacitance (the ability to store an electric charge) on an analog pin (in our case, analog pin **A1**). As the moisture around a nail in the plant's pot increases, the value increases (we're creating a bigger capacitor). Dry soil doesn't work as well for electricity (like most dry materials except metals) so the value read by the Circuit Playground Express will be lower.

## Pick Your Plant

This project works best on indoor plants in a pot or indoor window bed. Outdoor use would require waterproofing and the "water me" indication wouldn't be obvious if it was outdoors.

The plant in my example is a baby Maple tree in a smallish pot. As it grows, it'll need a bigger pot!

You will need to spend a few minutes with your chosen plant to get a good idea of when it is wet and dry to provide a good, accurate measurement. So pick out your plant and do not water it before starting.

You might want to look on the Internet as to what type of soil moisture makes the plant flourish best. You can adjust the moisture values in your program to "dial in" the best values for your particular plant.

---

## Build



Use the alligator clip wire combo to clip onto the Circuit Playground Express **A1** pad. The other end should clip onto your nail or piece of metal. The nail should be in the soil about 2 inches (5 centimeters).

For the Code portion of this build, you will want the Circuit Playground Express connected via a USB cable connected to a computer capable of running MakeCode. The USB cable will provide the Circuit Playground Express power for this portion of the build.



When your code is done on the next page, unplug the Circuit Playground Express from the USB cable and plug in your battery pack. Be sure your off/on switch on the battery pack is set to **on**.

---

## Code

This project uses Microsoft MakeCode for the code to read the soil sensor.

If you are new to Microsoft MakeCode, Adafruit has an excellent getting started tutorial at <https://learn.adafruit.com/makecode> (<https://adafru.it/wWd>).

The following is a short program that takes a moisture value we read as dry and gives a red neopixel reading if the reading falls below this dry amount.

Your MakeCode screen should have the following MakeCode program loaded.

```
on start
  set dry_value to 1500
  set brightness 45
  clear

forever
  set soil_reading to button pin A1 value
  console log value "soil reading" = soil_reading
  if soil_reading < dry_value then
    set all pixels to red
  else
    clear
  pause 2000 ms
```

The value read by the Circuit Playground Express is set to a level that might be a good dry soil value **1500**. This number can be adjusted to give a better accuracy.

[Load this program into the Circuit Playground Express via Microsoft MakeCode. \(https://adafru.it/D1f\)](https://adafru.it/D1f)

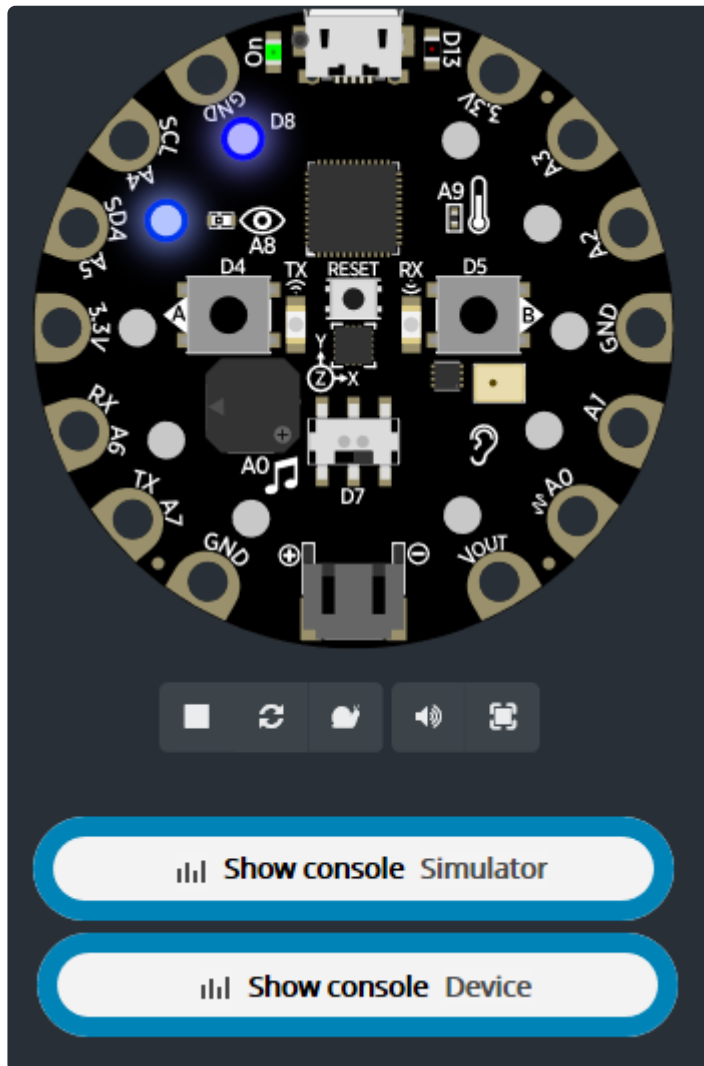
[Open this project in MakeCode](https://adafru.it/D1f)

<https://adafru.it/D1f>

Ensure you have your plant monitor hooked up as done on the Build page.

## For Windows

Click on the console output and see what reading is being printed out.



Click on the Show console Device button and look at the output values. With the moisture monitor hooked up, the reading should be somewhat greater than about **890**.

## Manual Calibration, All Platforms

Typical values: **890** or so with no alligator clip, **2160** for an alligator clip and 3 inch nail, **2600** in my dry(ish) pot with plant, You can set it lower, the Adafruit crew has **1500** for dry soil. Your numbers may vary. Be sure you have your Circuit Playground Express away from your body and on a dry, non-conductive surface like a countertop and keep your hands away while making settings to ensure things are not biased by the surroundings.

If your soil is dry, then the number you have with everything hooked up and no lights, should be good for your dry soil baseline. The larger the number, the wetter the soil. Note the number you are reading.

Change the value of the variable `dry_value` in the `on start` block to the value while the pot is dry and the lights are not on. You can then add water to the pot which will make the number go up and the light will go off when the project detects the soil is reading  $>$  `dry_value`. It may take a few tries to dial in the number perfect for your plant. This will be your dry soil benchmark.

Any time the Circuit Playground Express reads a value the value that is the dry value or lower, it will turn all the lights red, alerting you. If the read value is greater than `dry_value`, the NeoPixels will be turned off via the `clear` block.

Readings are taken every two seconds (2000 milliseconds) per the `pause` block on the bottom. If you want to increase the idle time between readings, add additional `pause` blocks. If you'd like a very long time between readings, see the trick at the bottom of the Going Further page.

Once you have your code "dialed in" with the final `dry_value`, you can download the final code via Microsoft MakeCode to the Circuit Playground Express.

Place fresh AA batteries into your battery pack.

You may now unplug the USB cable to your computer and plug in the battery pack into the black battery jack on the Circuit Playground Express. The green On light should be glowing next to the USB jack letting you know you have power.

## Final Setup

Set your plant in its final location. Place the Circuit Playground Express so you will notice the color of the NeoPixels as you walk by and the battery pack tucked away.

Go ahead and water your plant. The NeoPixels should be unlit. If they are still red, you'll need to go back and check your `dry_value`.

You're now set to have your Circuit Playground Express monitor your plant moisture!

---

# Going Further

The Circuit Playground Express can alert via other NeoPixel animations. The design constraint would be how much battery you use over time. You can adjust the NeoPixel brightness in the `set brightness` block - even a value of 25 is easily seen.

If you always have a green "good" display and then it turns into a flashing red display, that's alerting. But having the NeoPixels always on will drain battery. Using the animations defined in the LIGHT block group also gives an animated display at the expense of battery power.

You can have a sound alert but this might become annoying. See the first code below. You can always have fun and have it play a tune. In practice, it becomes annoying (and drains battery), so lights are best.

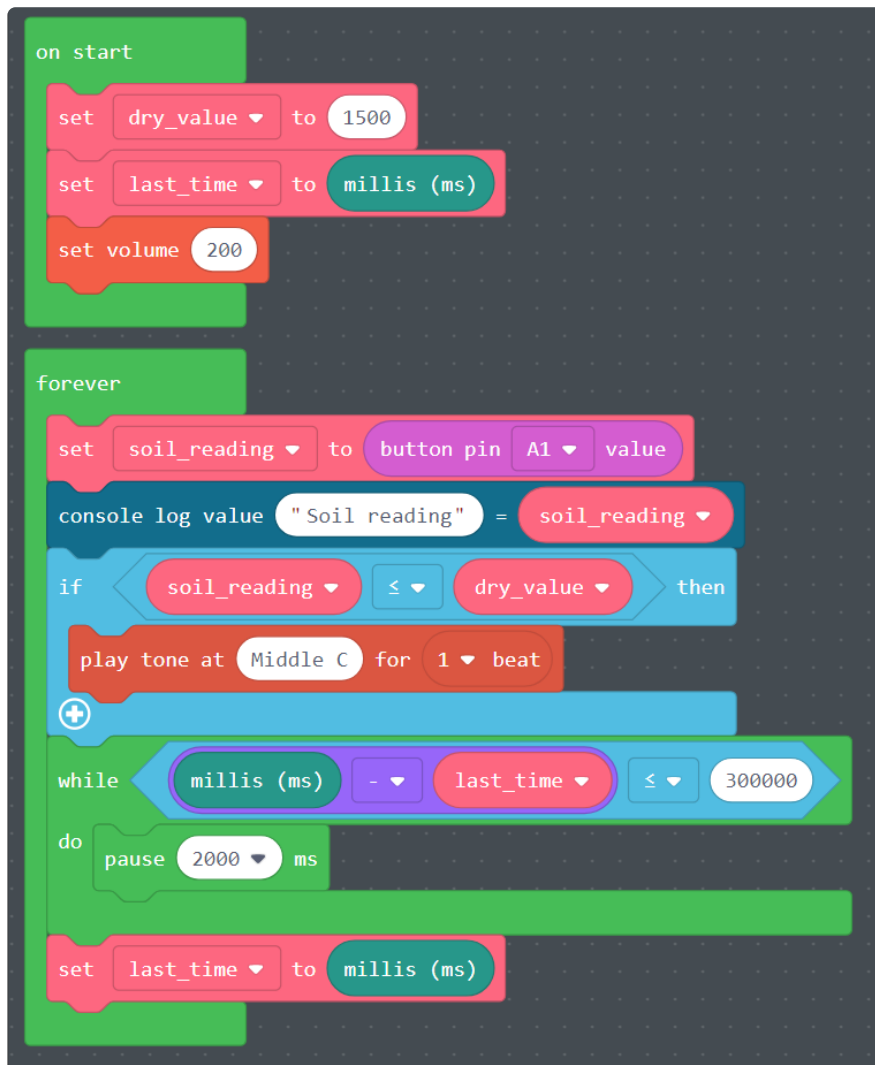
```
on start
  set dry_value to 1500
  set brightness 45
  clear

forever
  set soil_reading to button pin A1 value
  console log value "soil reading" = soil_reading
  if soil_reading < dry_value then
    set all pixels to red
    play tone at Middle C for 1/2 beat
  else
    clear
  pause 2000 ms
```

# Assistive Technology

For a person with limited sight, the sound might prove to be a better choice. The light code could be removed to save power and the sound stretched to a longer time.

The code below allows for much longer times without too many pause blocks. This program waits 5 minutes ( 300000 milliseconds) and makes a beep if the soil is dry. This may be better for someone who cannot see NeoPixels. Adjust the number of milliseconds in the `while` loop to increase or decrease from 5 minutes. Again, the sound should be such that it alerts but does not annoy.



```
on start
  set dry_value to 1500
  set last_time to millis (ms)
  set volume 200

forever
  set soil_reading to button pin A1 value
  console log value "Soil reading" = soil_reading
  if soil_reading ≤ dry_value then
    play tone at Middle C for 1 beat
  while millis (ms) - last_time ≤ 300000
  do
    pause 2000 ms
  set last_time to millis (ms)
```