



SmartElex Soil Moisture Sensor

Have you ever wanted your plants to tell you when they need watered? Or know how saturated the soil in your garden is? With the [Soil Moisture Sensor](#), you can do just that! This tutorial will show you how to get started using the Soil Moisture Sensor.



Hardware Overview

Our original [Soil Moisture Sensor](#) is pretty straightforward when it comes to hookup. There are only three pins to connect: **VCC**, **GND**, and **SIG**.

You need to supply power to VCC and GND. We recommend not powering the sensor constantly to prevent corrosion of the probes (more on this in a bit). SIG provides an analog signal out that can be attached to the ADC pin on any microcontroller. The value read on SIG will vary depending on the voltage with which you power the sensor.

Theory of Operation

The two probes are acting as a variable resistor – more water in the soil means better conductivity and results in a lower resistance and a higher SIG out. Your analog readings will vary depending on what voltage you use for Vcc as well as the resolution of your ADC pins.

Powering the Sensor

we recommend powering the Soil Moisture Sensor with between **3.3V - 5V**. Please note that the analog value returned will vary depending on what voltage is provided for the sensor.

One commonly known issue with soil moisture sensors is their short lifespan when exposed to a moist environment. Another way to extend the lifespan of your sensor is to only power it when you take a reading. Using a digital pin set to HIGH on an Arduino, for example, is an easy way to accomplish this. If you wish to power the sensor with more than a digital pin on your microcontroller can provide, you could always use a transistor.

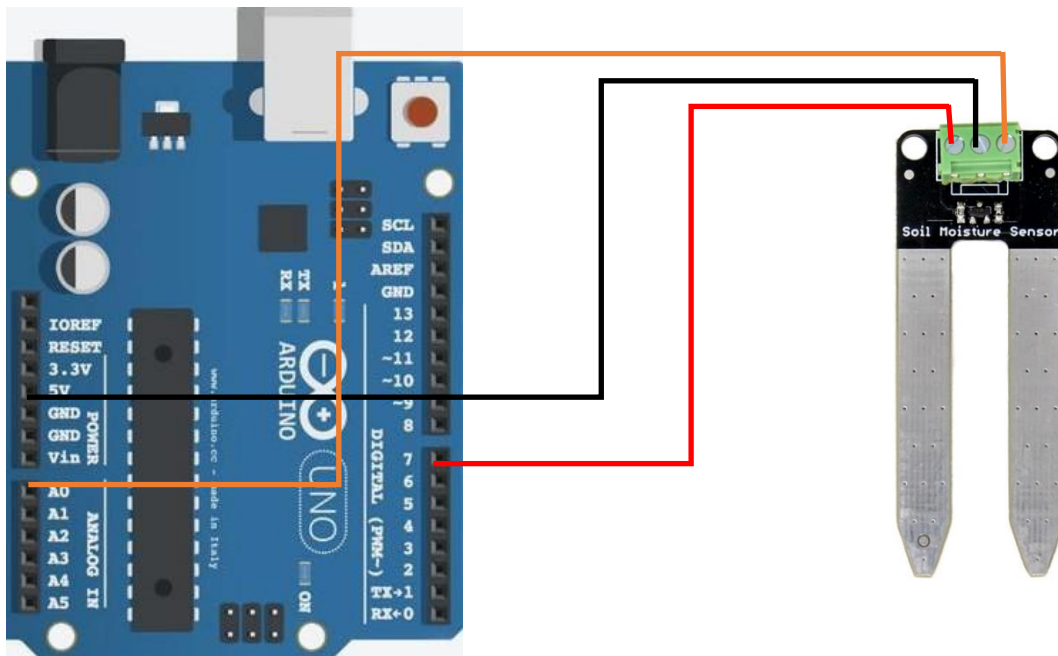
Weatherproofing

If you intend to use this sensor outdoors, we recommend adding a little protection to the PCB to extend its lifespan. You could always use good ol' fashioned hot glue. However, hot glue does not hold up well in the sun and is only recommended for projects that will not be exposed to high temperatures. For projects that need to be able to withstand *all* the elements, you could use a conformal coating to cover the SMD parts on the PCB as well as your solder connections.

Moisture Sensing Basic Example

For this example, we will connect the sensor to a Arduino UNO in a minimalist fashion to show its output over the serial terminal.

Connect your Soil Moisture Sensor to your board as shown in the diagram below.



Arduino	MOISTURE SENSOR
D7	VCC
GND	GND
A0	SIG

Once the circuit is built, upload the following code to your Arduino.

```
int val = 0; //value for storing moisture value
int soilPin = A0; //Declare a variable for the soil moisture sensor
int soilPower = 7; //Variable for Soil moisture Power

//Rather than powering the sensor through the 3.3V or 5V pins,
//we'll use a digital pin to power the sensor. This will
//prevent corrosion of the sensor as it sits in the soil.

void setup()
{
```

```

Serial.begin(9600); // open serial over USB

pinMode(soilPower, OUTPUT); // Set D7 as an OUTPUT
digitalWrite(soilPower, LOW); // Set to LOW so no power is flowing through the sensor
}

void loop()
{
Serial.print("Soil Moisture = ");
// get soil moisture value from the function below and print it
Serial.println(readSoil());

// This 1 second timeframe is used so you can test the sensor and see it change in real-
time.
// For in-plant applications, you will want to take readings much less frequently.
delay(1000); // take a reading every second
}
// This is a function used to get the soil moisture content
int readSoil()
{

digitalWrite(soilPower, HIGH); // turn D7 "On"
delay(10); // wait 10 milliseconds
val = analogRead(soilPin); // Read the SIG value from sensor
digitalWrite(soilPower, LOW); // turn D7 "Off"
return val; // send current moisture value
}
//////////////////////////////////////////////////////////////////END//////////////////////////////////////////////////////////////////

```

Note that in the code, we are powering the soil moisture sensor with a digital pin on the UNO, not directly from 3.3V or 5V. Since this is a 5V device, the digital IO pins have a HIGH voltage of ~5V.

Once the sketch is uploaded, open a **Serial Monitor** window to see the output from the UNO. You should see a value at or close to 0 when the sensor is not touching anything. To see it sense moisture, you can grab both probes with your hand. The moisture from your body will be enough for the sensor to detect.

Calibration

System Calibration

To get any sort of useful data out of your Soil Moisture Sensor, it is advised that you calibrate it to whatever soil you plan to monitor. Different types of soil can affect the sensor, and you may get different readings from one composition to the next. Before

you start storing moisture data or triggering events based on that value, you should see what values you are actually getting from your sensor. Using the sketch above, note what values your sensor outputs when the sensor is completely dry vs when the sensor is completely submerged in a shallow cup of water. Depending on what microcontroller you're using, the operating voltage of that microcontroller, and the resolution of its analog-to-digital converter, your results will vary.

For example, using the same circuit above, if I detach the VCC pin from D7 and attach it directly to the 5V supply on the UNO, you'll see the close to the following values in the serial monitor when the sensor is dry (~0) and when it is completely saturated with moisture (~880).

```
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 883
Soil Moisture = 881
Soil Moisture = 880
Soil Moisture = 879
Soil Moisture = 879
Soil Moisture = 878
Soil Moisture = 877
Soil Moisture = 877
Soil Moisture = 877
Soil Moisture = 876
```

But, if I take the VCC pin and connect it to the 3.3V supply on the UNO, the values change. As expected, they get lower since there is less resolution between 0V and 3.3V than there is between 0V and 5V.

```
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 0
Soil Moisture = 541
Soil Moisture = 540
Soil Moisture = 539
Soil Moisture = 539
Soil Moisture = 539
Soil Moisture = 539
Soil Moisture = 539
Soil Moisture = 539
```

Thus, it is difficult to write an example sketch that works for all platforms. It really depends on the operating voltage and ADC resolution of the board you're using.

Soil Calibration

Once you have an idea what values your sensor is outputting in completely dry and completely wet situations, it's time to calibrate your sensor for the specific soil you want to monitor. Do the same test above, only this time test your soil when it is as dry as possible, then measure it when the soil is completely saturated with moisture. Getting these values and comparing them to the ones from the previous calibration will give you the best insight into what values mean for your specific plant and soil. This test may take some trial and error and patience. Be careful not to over-water (or under-water) your plants during these tests.

Once you have a good handle on the values you can expect, you can use the `map()` function to adjust your code accordingly.