Instructions:

PLANT MONITOR - ARDUINO
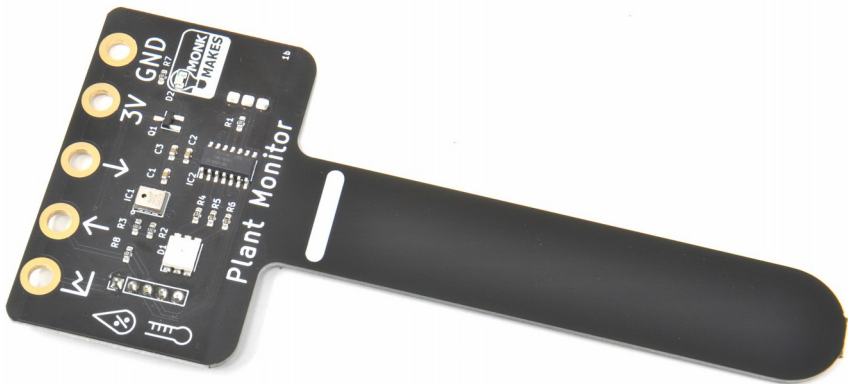
Instructions version 1b.

TABLE OF CONTENTS

WARNING

Only the prong of the Plant Monitor below the white line should be allowed to get wet. If the top of the board gets wet, disconnect it from everything, dry it using a paper towel and then leave it the throughly dry out before trying to use it again.

INTRODUCTION

The MonkMakes Plant Monitor measures soil moisture, temperature and relative humidity. This board is compatible with the BBC micro:bit, Raspberry Pi and most microcontroller boards.

- Superior capacitative sensor (no electrical contact with soil)

- Alligator / crocodile clip rings (for use with BBC micro:bit and Adafruit Clue etc.

- Ready soldered header pins for Arduino and other microcontroller boards.

- Easy to use UART serial interface

- Additional analog output for moisture only

- Built-in RGB LED (switchable)
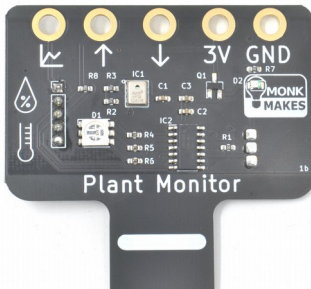
USING THE PLANT MONITOR

The plant monitor should be placed as shown below.

The front side of the prong should be as close to the edge of the pot as possible. The sensing all takes place from the far side of the prong.

The electronics should be facing out of the pot and the prong of the Plant Monitor pushed into the dirt as far as the white line (but no deeper).

It's a good idea to attach the wires you are going to use to connect to the Plant Monitor before positioning it in the plant pot.

Once powered up, the plant monitor will immediately start displaying the level of wetness using the builtin LED. Red means dry, green means wet. Before you put the Plant Monitor in the pot, try gripping the prong in your hand and the moisture of your body should be enough to alter the LED's color.
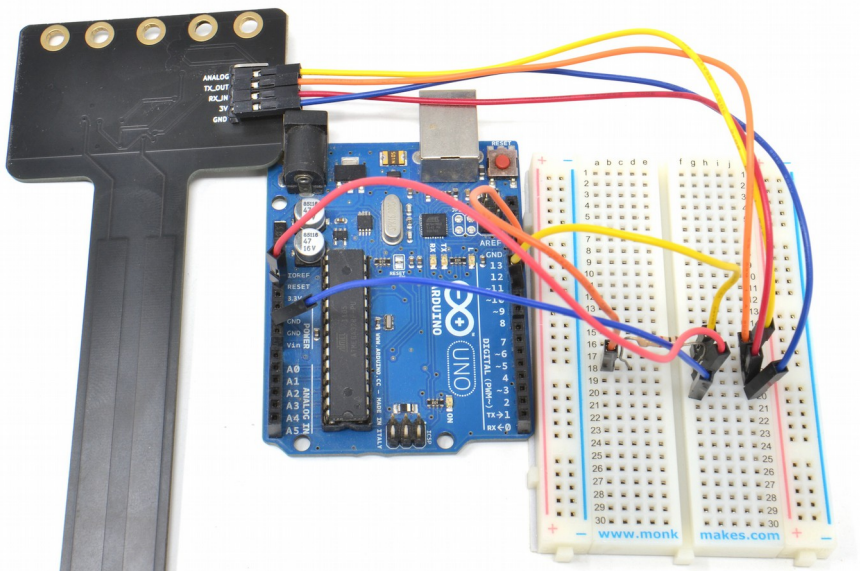
ARDUINO

Warning: The Plant Monitor is designed to operate at 3.3V, not the 5V that some Arduinos such as the Arduino Uno operate at. So, never power the Plant Monitor with 5V and make sure that none of it's input pins receive more than 3.3V.

To connect a 5V Arduino, such as the Arduino Uno or Leonardo you will need to use a level converter or a (as we have here) a 1kΩ resistor to limit the current flowing from the 5V Soft Serial transmit pin of the Arduino (pin 11) to the 3.3V RX_IN pin of the Plant Monitor.

Here's what this looks like, solderless breadboard is used to hold the resistor (in the middle of the breadboard), male to male jumper wires to connect the the Arduino to the breadboard and female to male jumper wires to connect the Plant Monitor to the breadboard. The connections are as follows:

• GND on the Arduino to GND on the Plant Monitor

• 3V on the Arduino to 3V on the Plant Monitor

• Pin 10 on the Arduino to TX_OUT on the Plant Monitor

• Pin 11 on the Arduino to RX_IN on the Plant Monitor **via a 1kΩ resistor**. Note that the resistor is not needed for a 3V Arduino.

Once its all connected, you can install the Arduino library for the PlantMonitor by going to https://github.com/monkmakes/mm_plant_monitor and the from the Code menu, select Download ZIP.
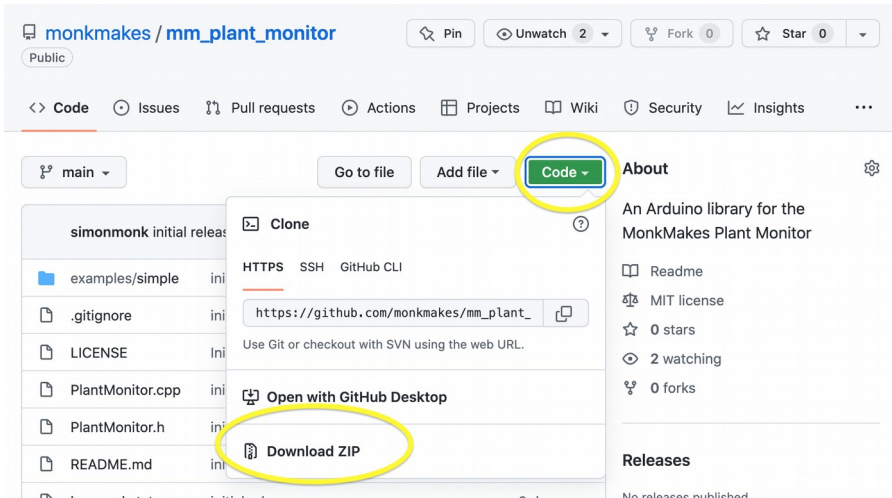

Now open the Arduino IDE and from the Sketch menu select the option to Add .ZIP Library and navigate to the ZIP file you just downloaded.
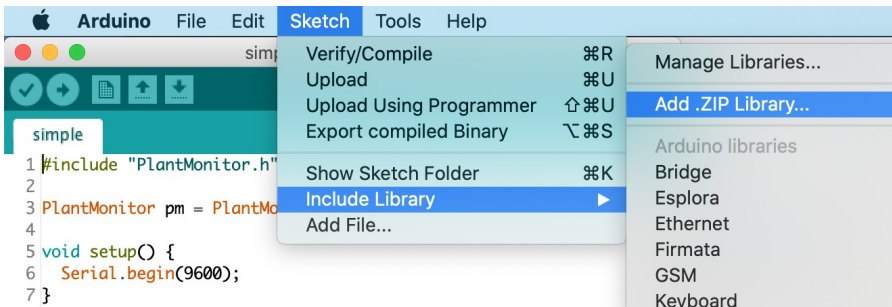
As well as installing the library, this will also fetch an example program that you will find in the Examples sub-menu of the File menu, under the category Examples from Custom Libraries.
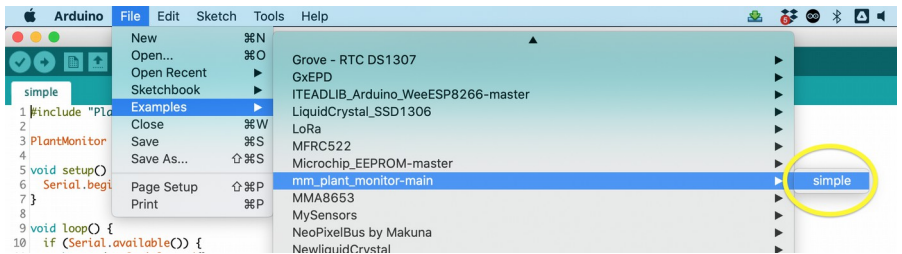
Upload the example called *Simple* to your Arduino and then open the Serial Monitor. Here, you will see a series of readings. You can also turn the Plant Monitor's LED on and off from the Serial Monitor by sending serial commands. Type L in the send area of the Serial Monitor and then press the Send button to turn the LED on, and l (lower-case L) to turn the LED off.
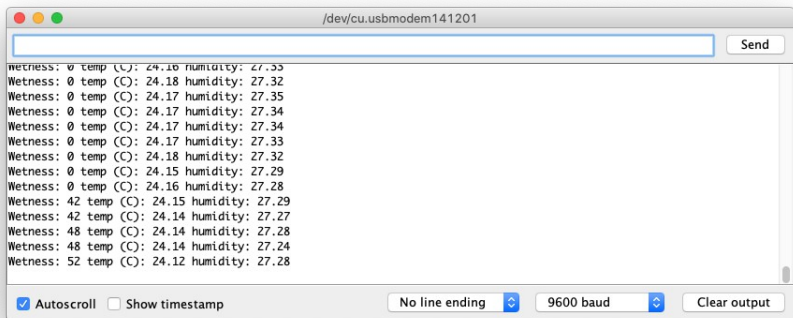
Here is the code for this example:

```
#include "PlantMonitor.h"

PlantMonitor pm = PlantMonitor(10, 11); // RX, TX

void setup() {
  Serial.begin(9600);
}

void loop() {
  if (Serial.available()) {
    char cmd = Serial.read();
    if (cmd == 'l') {
      pm.ledOff();
    }
```

```
    else if (cmd == 'L') {
       pm.ledOn();
    }
  }
  report();
  delay(1000);
}

void report() {
  Serial.print("Wetness: ");
  Serial.print(pm.getWater());
  Serial.print(" temp (C): ");
  Serial.print(pm.getTemp());
  Serial.print(" humidity: ");
  Serial.println(pm.getHumidity());
}
```

The library uses another Arduino library called `SoftSerial` to communicate with the Plant Monitor. This can carry out serial communication on any of the Arduino pins. So, when an instance of `PlantMonitor` called `pm` is created, the pins to be used to communicate to the Plant Monitor hardware are specified (in this case, 10 and 11). If you like, you can change 10 and 11 for other pins.

The main `loop` checks for incoming messages of *L* or *l* from you to turn the LED on or off respectively, using the `pm.ledOn` or `pm.ledOff` commands.

Getting readings from the PlantMonitor all takes place in the report function that writes out all the readings to the Arduino IDE's Serial Monitor.

TROUBLESHOOTING

Problem: When I first connect power to the PlantMonitor, the LED cycles through colors. Is this normal?

Solution: Yes, this is the Plant Monitor doing a self-test as it starts up.

Problem: The LED on the Plant Monitor does not light at all.

Solution: Check the power connections to the Plant Monitor. Alligator leads and jumper wires can become faulty. Try changing the leads.

Problem: I am connecting using the serial interface, and I get wetness readings, but the humidity and temperature readings are wrong and not changing.

Solution: You may have inadvertently powered your Plant Monitor from 5V rather than 3V. This may have destroyed the temperature and humidity sensor.

SUPPORT

You can find the Product's information page here: https://monkmakes.com/pmon including a datasheet for the product.

If you need further support, please email support@monkmakes.com.

MONKMAKES

As well as this kit, MonkMakes makes all sorts of kits and gadgets to help with your electronics projects. Find out more, as well as where to buy here: https://monkmakes.com you can also follow MonkMakes on Twitter @monkmakes.