



Node-RED: Case Study

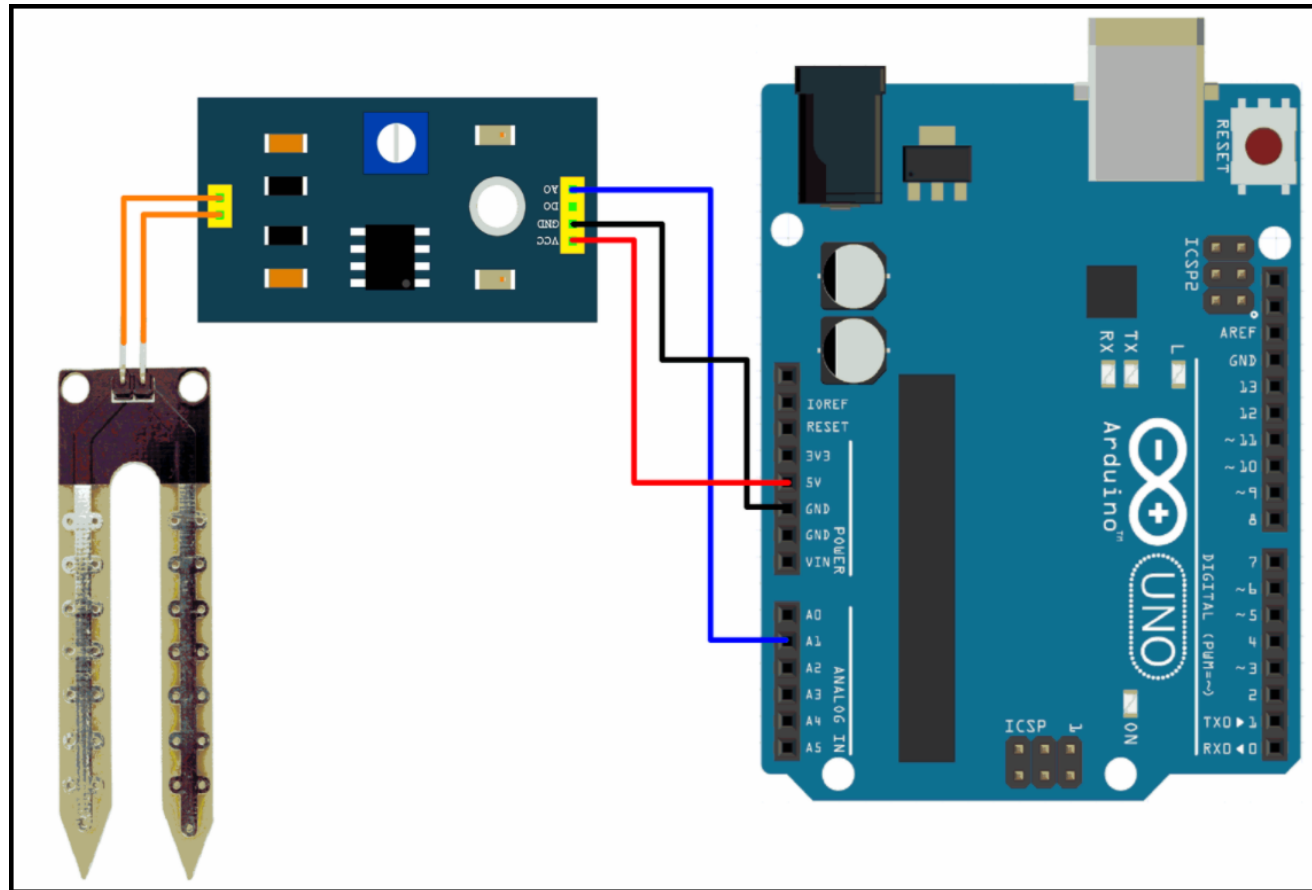


Requirement:

- i. Microcontroller x 1
- ii. Soil Moisture x 2
- iii. NodeRED

BDA CASE STUDY 1: (IoT Based Smart) Agriculture with Remote Monitoring System

The schematic diagram:



BDA CASE STUDY 1: (IoT Based Smart) Agriculture with Remote Monitoring System

The result:

When you run the sketch, you'll see the close to the following readings in the serial monitor:

- when the soil was dry (~850)
- when the soil was completely wet (~400)



Status: Dry
Test Reading: ~850



Status: Completely wet
Test Reading: ~400

<https://lastminuteengineers.com/soil-moisture-sensor-arduino-tutorial/>

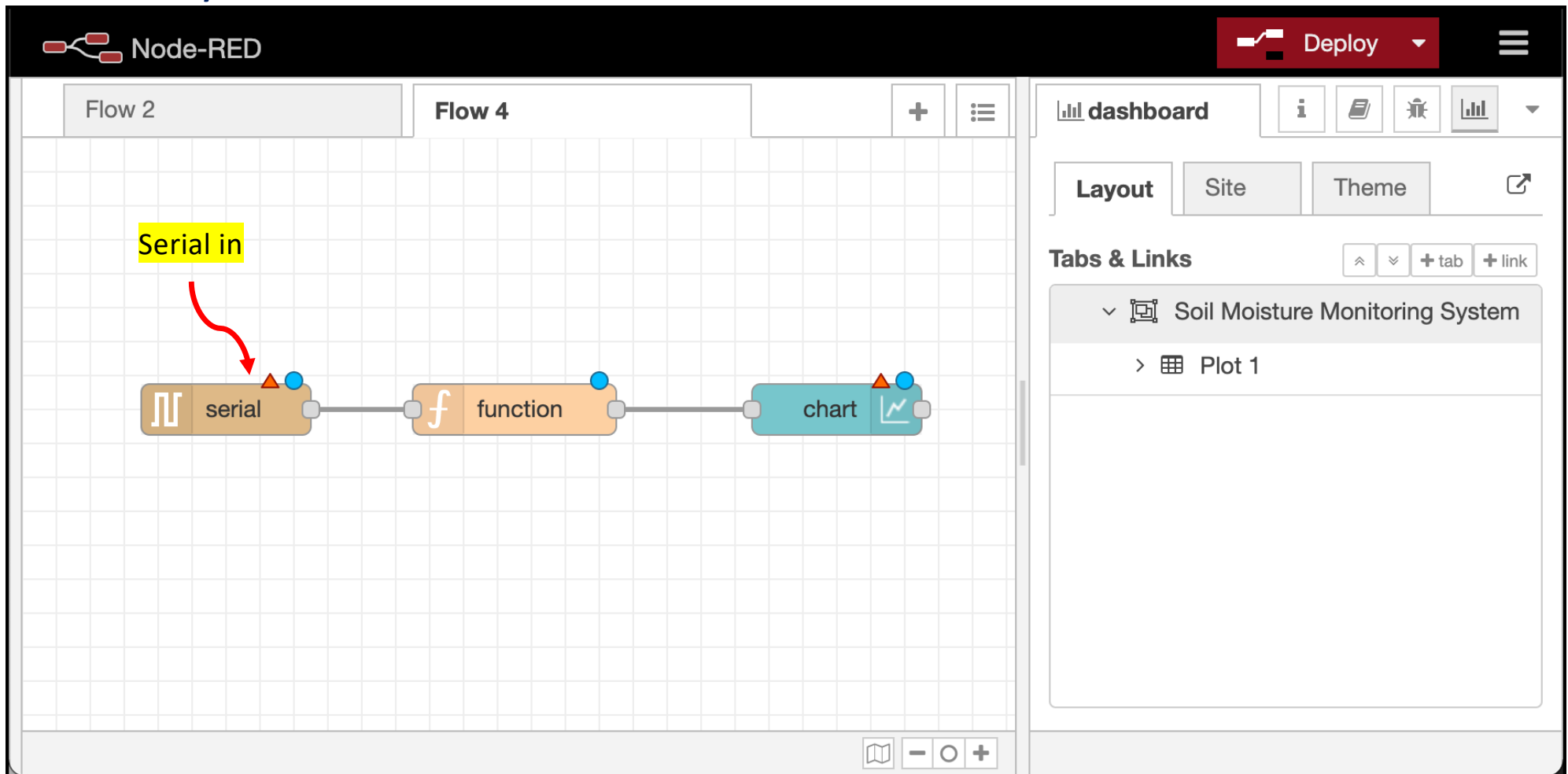
The sketch:

```
// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  String moist;
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  moist = string(sensorValue); // need to convert to string which will pickup by nodeRED
  // print out the value you read:
  Serial.print("Moisture Level: ");
  Serial.print(moist);
  Serial.println(","); //delimiter -> for nodeRED

  delay(1000);    // delay in between reads for stability
}
```

Node-RED layout:




BDA CASE STUDY 1: (IoT Based Smart) Agriculture with Remote Monitoring System

Node-RED properties:

Edit serial in node

Delete Cancel Done

Properties

Serial Port 

Name


☐ Enabled

v1-mar-21

Edit serial in node > Edit serial-port node

Delete Cancel Update

Properties

Serial Port 

Settings

| | | | |
|-----------|-----------|--------|-----------|
| Baud Rate | Data Bits | Parity | Stop Bits |
| 9600 | 8 | None | 1 |
| DTR | RTS | CTS | DSR |
| auto | auto | auto | auto |

Input

Optionally wait for a start character of , then

Split input

and deliver

Output

Add character to output messages

Request

Default response timeout ms

Tip: the "Split on" character is used to split the input into separate messages. Can accept chars (\$), escape codes (\n), or hex codes (0x03).

BDA CASE STUDY 1: (IoT Based Smart) Agriculture with Remote Monitoring System

Node-RED layout:

Edit function node

Delete Cancel Done

⚙ Properties

Name

Setup Function Close

```
1 var output = msg.payload.split(",");
2
3 var moist1 = parseInt(output[0]);
4
5 msg = {payload : moist1};
6
7 return msg;
```

⚙ Outputs 1

☐ Enabled

Edit chart node

Delete Cancel Done

⚙ Properties

Group

Size

Label

Type ☐ enlarge points

X-axis last hours OR points

X-axis Label ☐ as UTC

Y-axis min max

Legend Interpolate

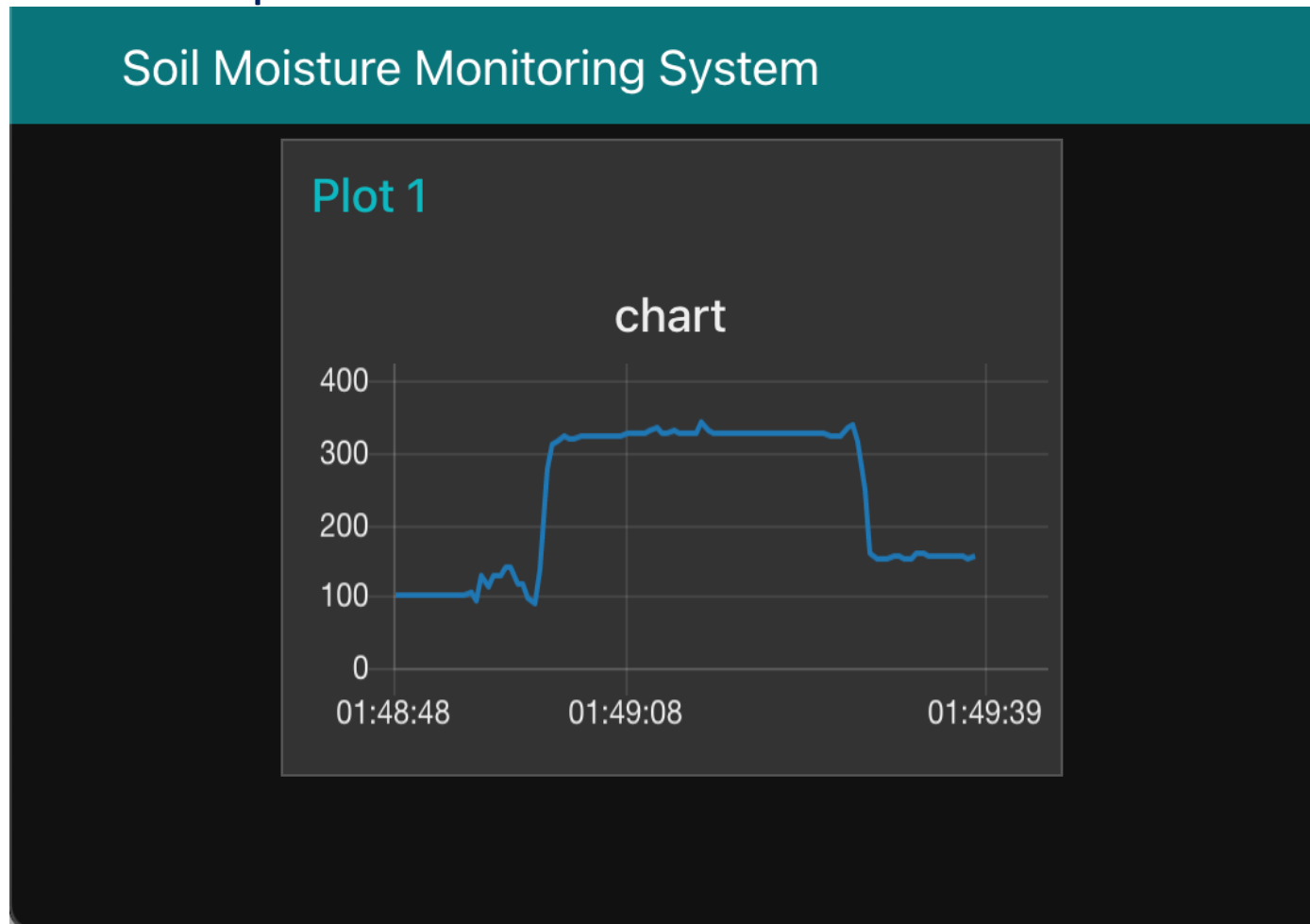
Series Colours

| | | |
|-------------|-------------|-------------|
| <div></div> | <div></div> | <div></div> |
| <div></div> | <div></div> | <div></div> |
| <div></div> | <div></div> | <div></div> |

Blank label

☐ Enabled

Node-RED expected output:



EXERCISES:

Add one more sensor, modify the function node.

QnA

END