Sensors and Communication

Welcome to Day 2 of our IoT Workshop! Today, we'll take a closer look at sensors, the backbone of IoT data collection. You'll gain insights into how sensors operate and discover key protocols like Analog, OneWire, and I2C that facilitate communication between devices and sensors. This foundational knowledge will empower you to effectively utilize sensors in your IoT projects.

1. Output - PWM

Now, we're stepping into the practical aspect of our workshop, focusing on Pulse Width Modulation (PWM) output with the ESP8266. PWM isn't just about adjusting the brightness of an LED; it's a versatile method of communication. Through this segment, you'll grasp how PWM operates as a form of control, allowing us to transmit information using varying pulse widths. This will come to life as we demonstrate PWM in action, using an attached LED as an illustrative example.

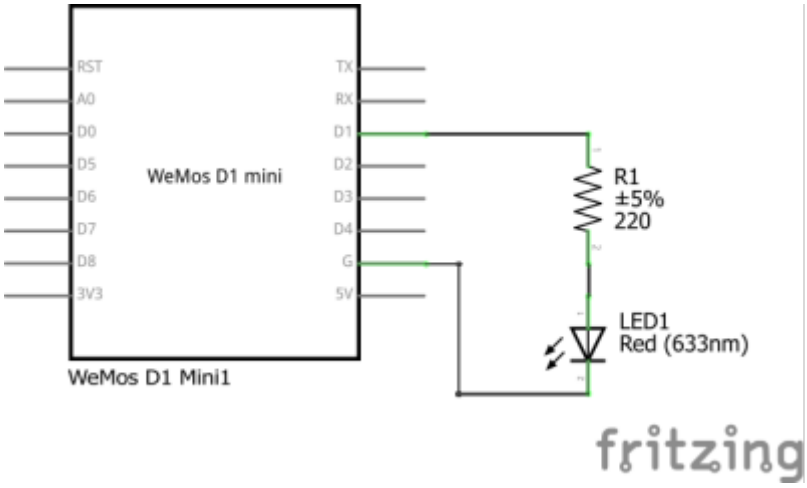


Fig. 1: ESP8266 with LED - Schematic

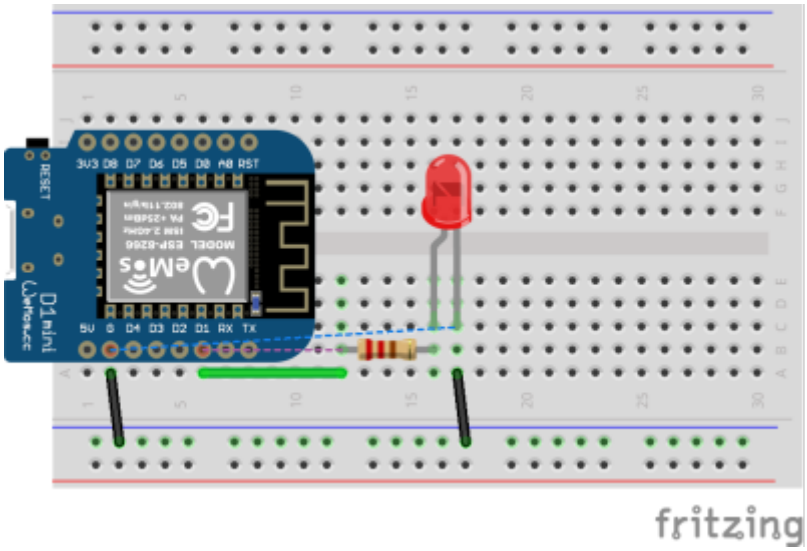


Fig. 2: ESP8266 with LED - Breadboard

2. Libraries

In the dynamic landscape of Arduino development, libraries play a pivotal role, especially when it comes to effective communication with sensors. These compact packages of pre-written code provide an invaluable resource, simplifying the process of interfacing with various sensors and components.

For IoT enthusiasts and makers, libraries serve as a bridge between complex hardware and user-friendly coding. By abstracting the intricate details of sensor communication, libraries empower you to focus on the application logic rather than low-level protocol intricacies.

Remember, in the Arduino world, libraries are your trusted companions, facilitating smooth communication and opening doors to limitless possibilities.

3. Inputs / Protocols

3.1. Analog sensors (ex. capacitive soil moisture sensor)

Through a special circuit, the sensor is able to translate from the soil moisture around it to an analog output voltage. The sensor consists of a PCB with a long thick trace on one end. If you want to learn more about this sensor you should read the following article: How Capacitive Soil Moisture Sensors Work by rbaron.

With these common sensors, we don´t need to reinvent the wheel. There are plenty of good tutorials out there on how to use this sensor. One we link here: Capacitive Soil Moisture Sensor - DFRobot Wiki There you will be able to learn how to integrate the sensor into your software as well as how to use it in the real world. Just keep in mind the electronics on top are not protected from the environment. Water shouldn't touch that part!

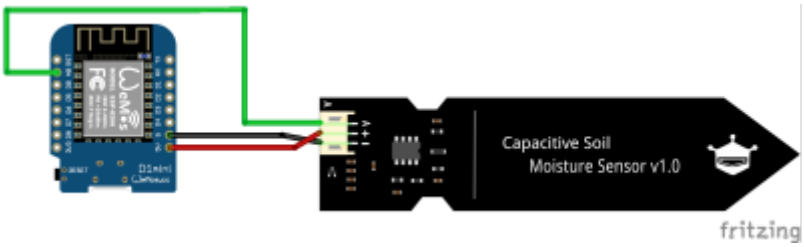


Fig. 3: Wiring of the analog sensor

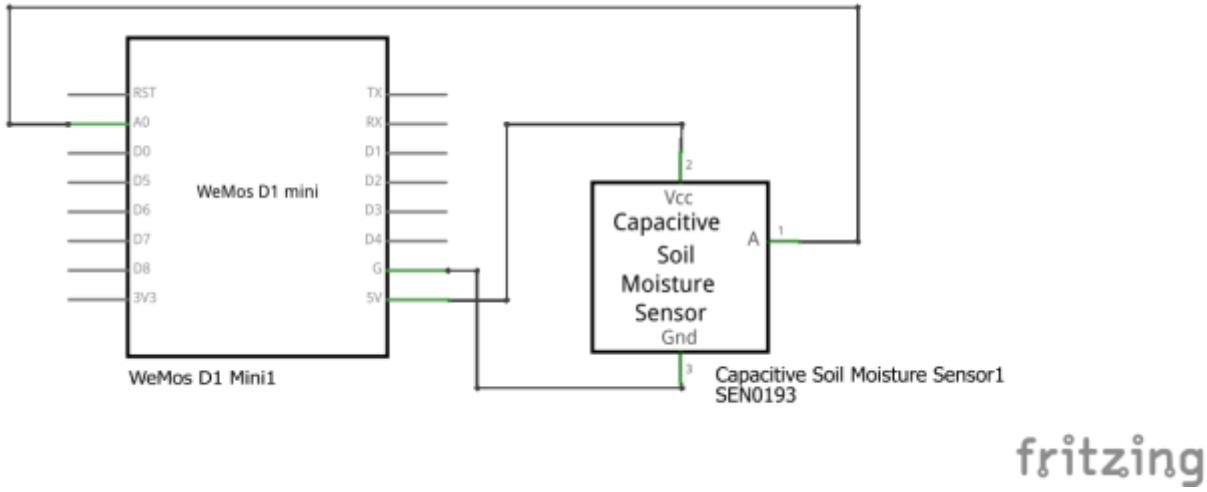


Fig. 4: Schematic

read-analog.ino

```
// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 115200 bits per second:
  Serial.begin(115200);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  // Convert the analog reading (which goes from 0 - 1023) to a voltage
(0 - 3.2V):
  float voltage = sensorValue * (3.2 / 1023.0);
  // print out the value you read:
  Serial.println("voltage: " + voltage);
  Serial.println("value: " + sensorValue);
```

3.2. 1-Wire (ex. temperature sensor)

One-Wire is a protocol that is heavily used by the company Maxim Integrated. One of the most popular sensors using this protocol is the (deprecated) DS18B20. It is an easy-to-use temperature sensor, that is available in multiple different form factors. In this workshop we use two different variants:

1. Water-Proof DS18B20
2. DS18B20 on a PCB

The DS18B20 needs in a minimal setup at least one resistor between its data line and the high potential. The first variant we mentioned doesn´t come with this included so it needs to be added on the breadboard. The second one has it already on its PCB included so it is a bit easier to integrate.

Here you can find some information that might be interesting for you, including a Datasheet for the

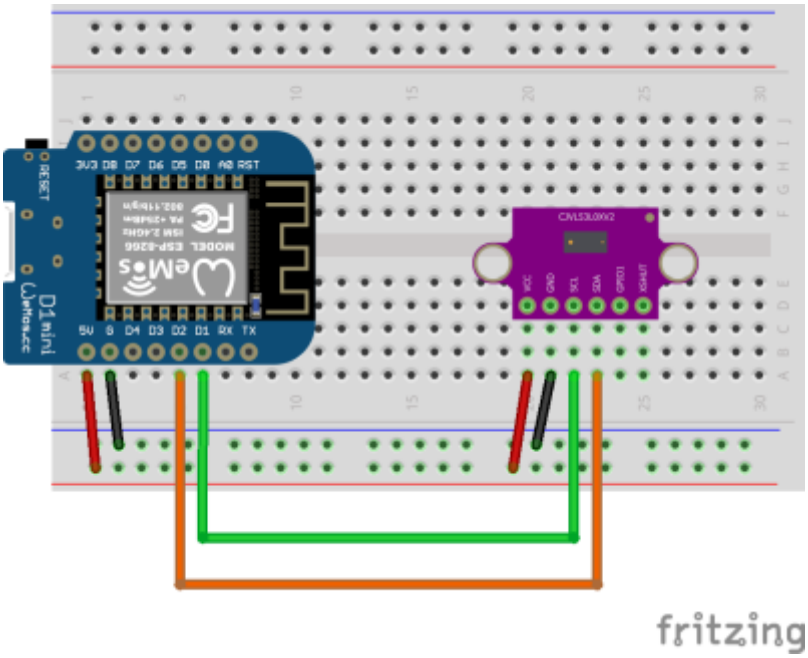
sensors (if you get a new sensor → READ THE DATASHEET), a more detailed look at the OneWire Protocol, and last but not least a Library that you can use to integrate the sensor in your system:

- DS18B20 Datasheet
- 1-Wire Technology Overview
- Library

Task: Try to get to research how to use the sensor with your MCU, regardless of which variant you have available.

3.3. I2C (ex. ToF sensor)

- VL53L0X Datasheet
- Here you can find a library for the sensor: Adafruit VL53L0X



3.3. UART

Recording

From:
https://wiki.eolab.de/ - **HSRW EOLab Wiki**

Permanent link:
https://wiki.eolab.de/doku.php?id=latinet:unicaes:workshops:sensors-23&rev=1693346283

Last update: **2023/08/29 23:58**