Lab Assignment 1 – IoT Device Hardware Assembly

Introduction

In this assignment, you will assemble a basic IoT device. This device will contain several sensors that will continuously take measurements of the ambience temperature, humidity and particles.

You will learn how to connect these sensors to the Arduino board, to write code to read the sensor data, to configure the LoRaWAN parameters, and how to send the data to a LoRaWAN gateway. In next assignments you will also learn how to manage and visualize the data.

The following materials are required:

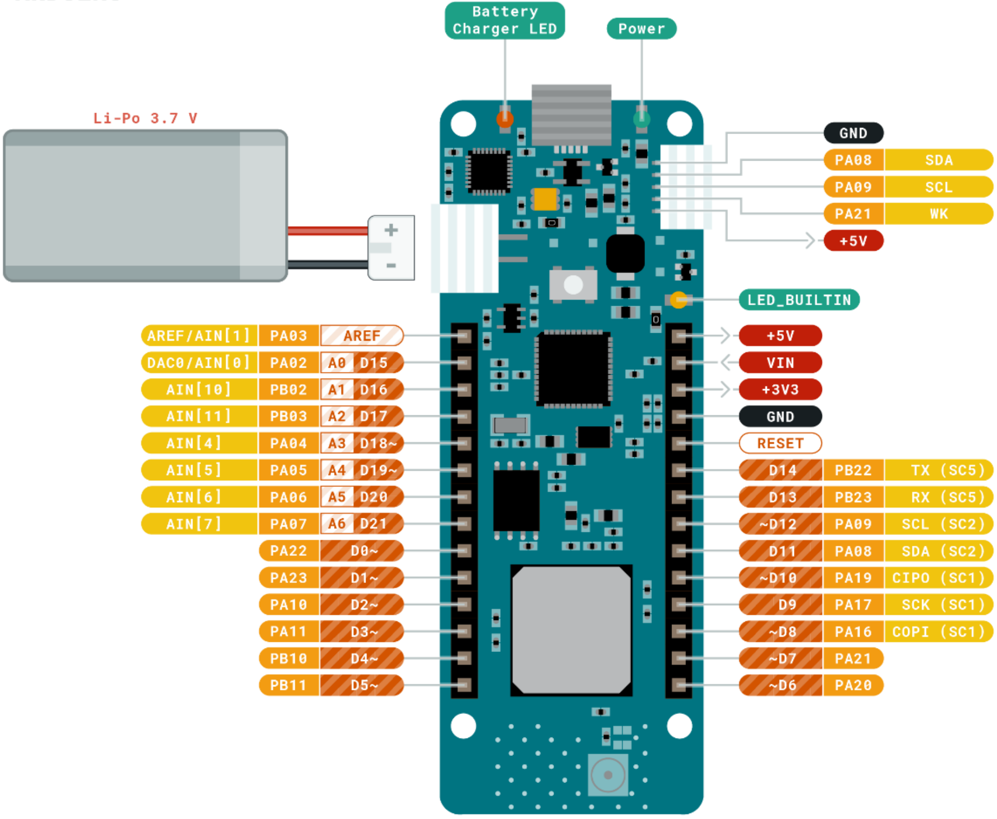
- Arduino MKRWAN 1310 board.
- HT21D-F temperature and humidity sensor.
- HPMA115S0 particle sensor.
- Breadboard.
- Resistors.
- Wire.

Knowing the hardware:

ARDUINO MKR WAN 1310

The Arduino MKRWAN 1310 is a development board that enables you to create and connect IoT devices using LoRaWAN technology. The board features a low-power microcontroller, a LoRa transceiver, a battery charger, and a connector carrier. The board can be programmed using the Arduino IDE, which provides a user-friendly environment to write and upload code. The board also supports various sensors and actuators that can be used to measure and control physical phenomena.

ARDUINO
MKR WAN 1310



■ Ground

🟥 Power

🟩 LED

■ Internal Pin

■ SWD Pin

□ Other Pin

▨ Digital Pin

▨ Analog Pin

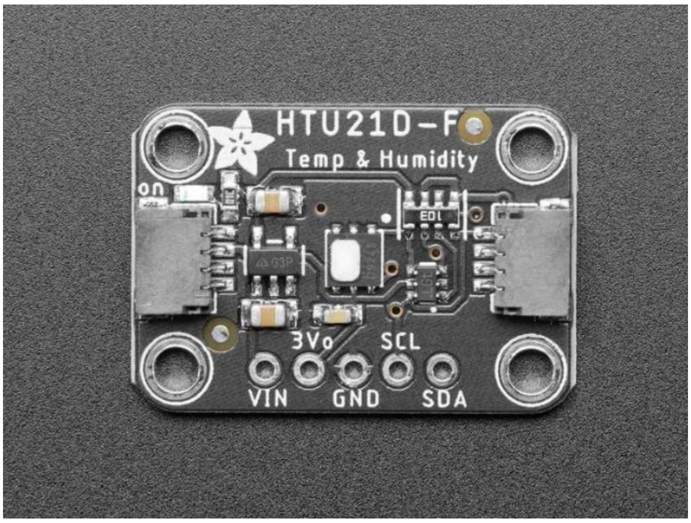
🟨 Default

🟧 Microcontroller's Port

ARDUINO.CC

HTU21D-F TEMPERATURE AND HUMIDITY SENSOR

The HTU21D-F is a I2C sensor. That means it uses the two I2C data/clock wires available on most microcontrollers and can share those pins with other sensors as long as they don't have an address collision. For future reference, the I2C address is 0x40 and you can't change it.



Pinout:

Vin - This is the power pin. Since the chip uses 3 VDC, a voltage regulator is included on board that will take 3-5VDC and safely convert it down. To power the board, give it the same power as the logic level of the microcontroller, in our case 5V from the Arduino board.

3Vo - This is the 3.3V output from the voltage regulator, we are not using this one.

GND - common ground for power and logic.

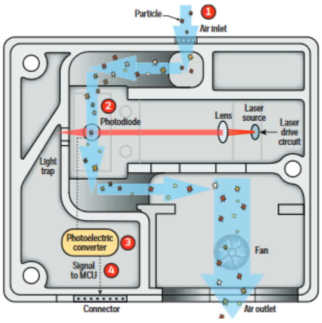
SCL - I2C data protocol clock pin, connect to your microcontrollers I2C clock line.

SDA - I2C data protocol data pin, connect to your microcontrollers I2C data line.

HPMA115S0 PARTICLE SENSOR

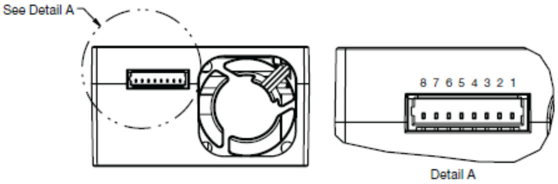
The Honeywell HPM Series Particulate Matter Sensor is a laser-based sensor which detects and counts particles using light scattering. The detection concentration range is 0 µg/m3 to 1,000 µg/m3. A laser light source illuminates a particle as it is pulled through the detection chamber. As particles pass through the laser beam, the light reflects off the particles and is recorded on the photo or light detector.

The light is then analyzed and converted to an electrical signal to calculate particle concentration. The Honeywell particle sensor provides information on the particle concentration for given particle concentration range.



1. Fan draws in air through inlet.
2. Air passes through the laser where the light reflected off the particles is captured by the photodiode.
3. The photodiode passes information to the photoelectric converter The photoelectric converter processes the signal from the particles into density.
4. Signal is transmitted to micro control unit where a proprietary algorithm processes the data and supplies outputs for the density of the particulate (μg/m^3).

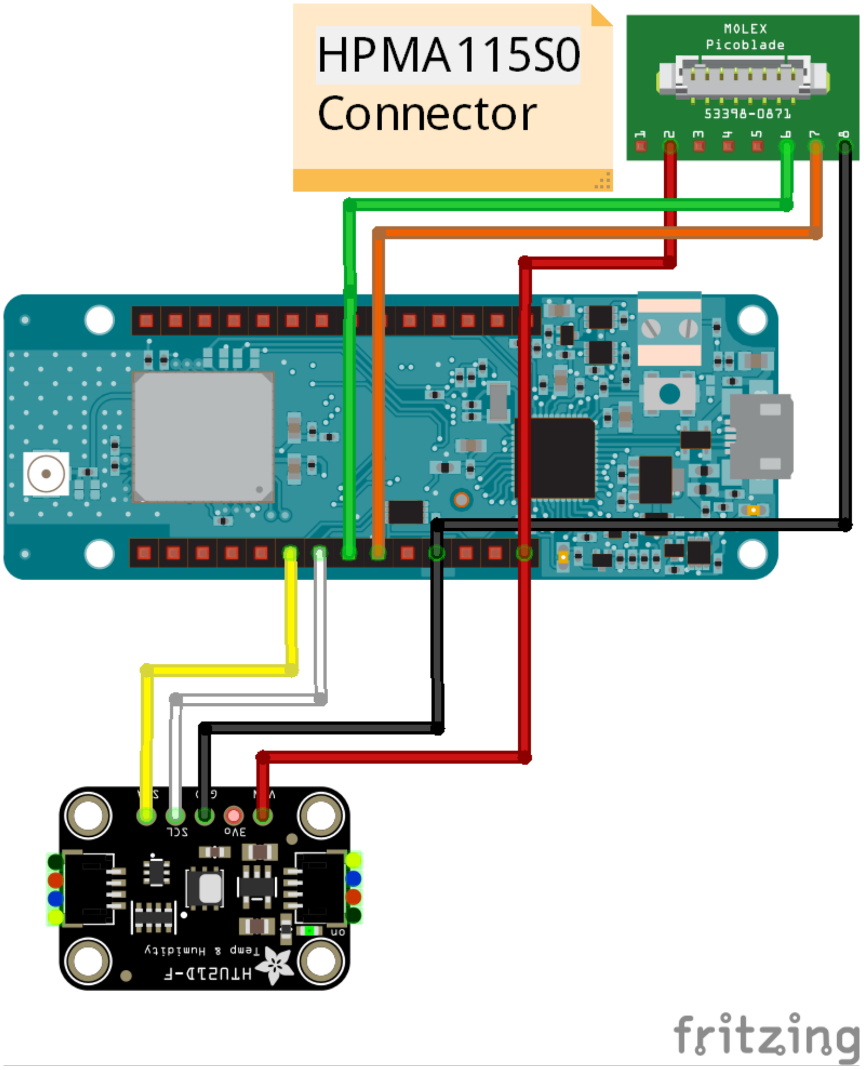
Table 2. Connector Pinout



Pin	Name	Description
1	+3.3 V	power output (+3.3 V/100 mA)
2	5 V	power input (5 V)
3	N/A	N/A
4	N/A	N/A
5	TEST	used for testing (NA)
6	TX	UART TX output (0 - 3.3 V)
7	RX	UART RX input (0 - 3.3 V)
8	GND	power input (ground terminal)

CIRCUIT ASSEMBLY

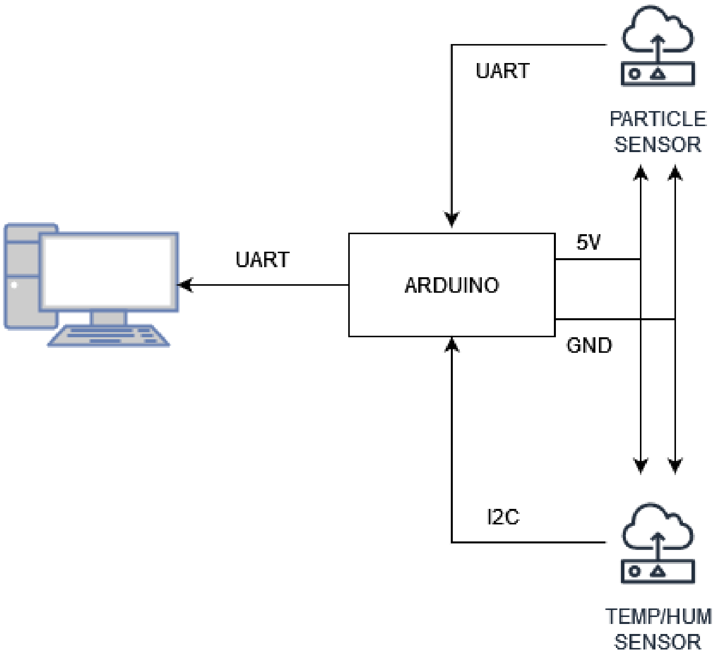
The circuit will be assembled on top of the breadboard, and you will be provided with all the necessary wire. You will have to check the documentation (provided at the bottom of this document) on both the Arduino and sensors pinouts to connect all elements successfully.



The steps to connect the components are as follows:

- Connect the **5V** pin of the Arduino MKRWAN 1310 board to the positive rail of the breadboard. Then connect the **Vin** pin of the HTU21D-F; and the **Pin 2** of the particles sensor to the positive rail (**red**).
- Connect the **GND** pin of the Arduino MKRWAN 1310 board to the negative rail of the breadboard. Then connect the **GND** pin of the HTU21D-F; and the **Pin 8** of the particles sensor to the negative rail (**black**).
- Connect the SCL pin of the Arduino MKRWAN 1310 board to the SCL pin of the HTU21D-F board (**white**).
- Connect the SDA pin of the Arduino MKRWAN 1310 board to the SDA pin of the HTU21D-F board (**yellow**).
- Connect the TX pin of the Arduino MKRWAN 1310 board to the RX pin (**pin 7**) of the HPMA115S0 sensor (**orange**).

- Connect the **RX** pin of the Arduino MKRWAN 1310 board to the TX pin (**Pin 6**) of the HPMA115S0 sensor (**green**).



SETTING UP THE ARDUINO IDE

Install the Arduino IDE on your machine and try to open and upload any of the example programs to ensure a proper connection between your board and the computer is done

First step in setting up the Arduino is to configure the IDE to use the MKRWAN 1310 board.

1. Open the Arduino IDE 2.

2. With the editor open, let's look at the left column. Here, we can see a couple of icons. Let's click the on the Arduino board icon, to the Board Manager menu.

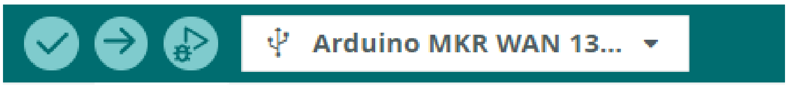
3. A list will now appear of all available cores. Enter the name “MKR” in the search field, and the “Arduino SAMD Boards” core will appear. Click on the "INSTALL" button.

4. This will begin an installation process, which usually only take a few moments.

5. When it is finished, we can look at the core in the boards manager column, where it should say "INSTALLED", as well as noting which version you have installed on your machine.

6. Let’s execute an example program. Go to File > Examples > 0.1Basics > Blink and click on it, a new window with the example program will open.

7. On the top tool bar select the Arduino MKR WAN 1310 option in the drop-down menu. Click on the ‘Tick’ icon to verify the program, and then click on the ‘arrow’ icon to upload it.



8. The Arduino board should blink an orange LED.

The program to develop must interface with both sensors and prepare a section of the code to send the received information over the LoRaWAN protocol to a server, which will be done in a future assignment.

You will need to install the following libraries in order to interact with the different sensors and capabilities. In the left toolbar click on the Arduino library manager icon, then proceed to install the following libraries:

MKRWAN by Arduino, provides APIs to communicate with LoRa and LoRaWAN networks.

Adafruit HTU21DF Library by Adafruit, designed specifically to work with the HTU21D-F in temperature-humidity sensor.

PMSensor-HPMA115 by Jed Parsons, library and example for receiving data from and interacting with Honeywell HPM series particulate matter sensors using the Arduino platform.

As the HPMA115 library only works with the "compact" version of the sensor, and we are working with the "standard" version, two more files must be included. Copy the files "HPMA115_Standard.h" and "HPMA115_Standard.cpp" from the GitHub repository (https://github.com/QartiaCube/BIP_course) and paste them in the directory "C:\Users\User\Documents\Arduino\libraries\PMSensor-HPMA115\src".

From the same repository, get the main program "IOT_practice_1.ino", execute it, and try to understand what the behavieour is of the different functions. We will explain this during the lesson.

DOCUMENTATION

Arduino

https://www.arduino.cc/reference/en/

https://www.arduino.cc/reference/en/language/functions/communication/serial/

https://docs.arduino.cc/learn/communication/wire

https://docs.arduino.cc/hardware/mkr-wan-1310

Temperature/Humidity sensor

https://www.adafruit.com/product/1899

Particle Sensor

https://www.mouser.es/datasheet/2/187/HWSC_S_A0012942921_1-3073234.pdf

Custom driver example

How To Write A Driver (STM32, I2C, Datasheet) - Phil's Lab #30