



# INTEL-IRRIS

Intelligent Irrigation System for Low-cost Autonomous Water Control  
in Small-scale Agriculture



# INTELLIGENT IRRIGATION SYSTEM FOR LOW-COST AUTONOMOUS WATER CONTROL IN SMALL-SCALE AGRICULTURE



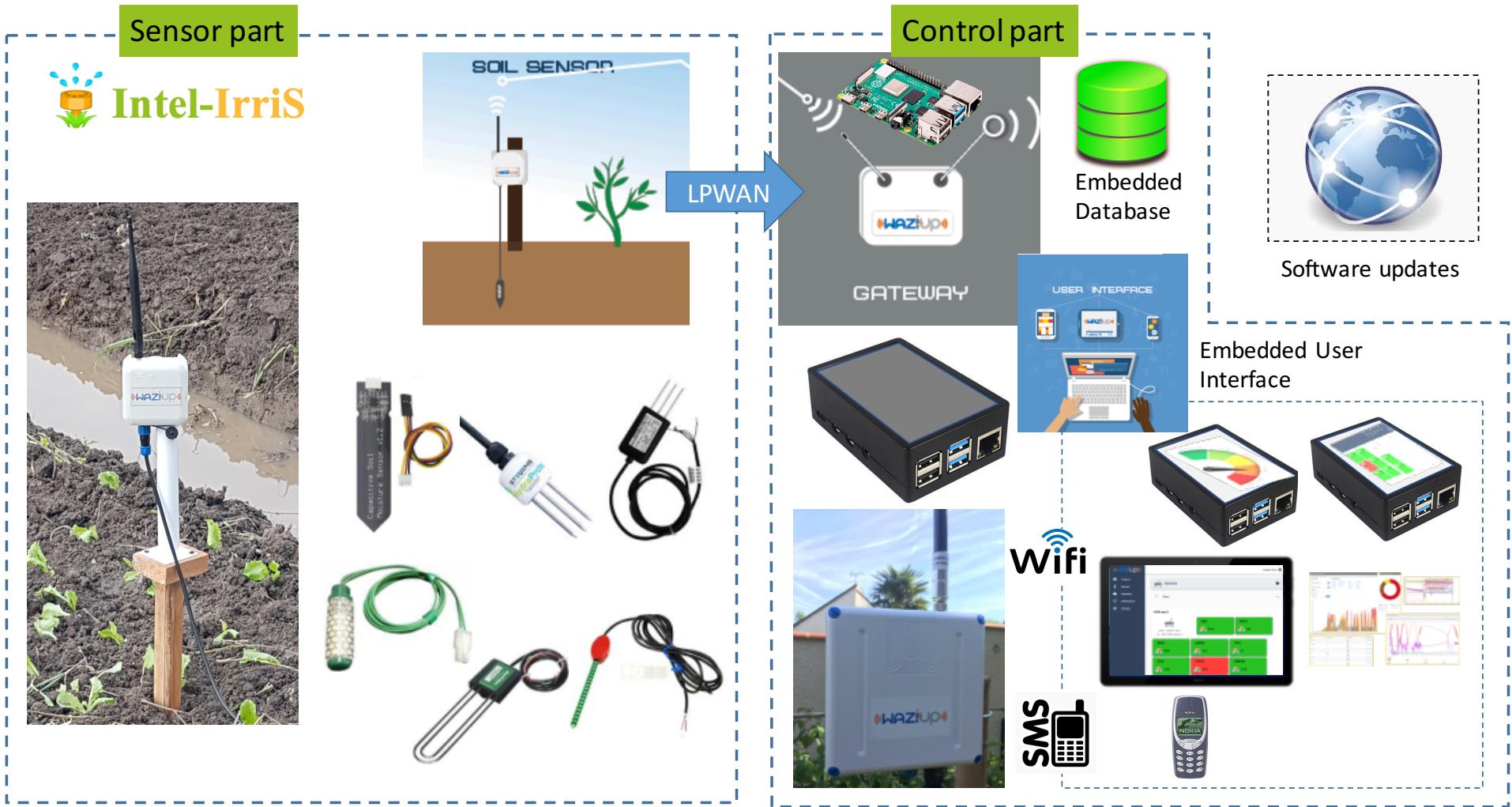
## Building the Intel-IrriS LoRa IoT platform Part 1: soil sensor device



Prof. Congduc Pham  
<http://www.univ-pau.fr/~cpham>  
Université de Pau, France



# Review: Technology components



# Review: Low-cost sensors



- Build on low-cost, low-power IoT expertise
- Increase accuracy of low-cost sensors by automatic and remotely controlled procedures for advanced calibration
- Enable deployment of several complementary low-cost sensors
- Include agricultural models / knowledge with corrective & predictive analytics

# Review: Smart embedded control

- Build on low-cost embedded & open IoT gateway expertise
- Implement the “Intelligent Irrigation in-the-box” with "plug-&-sense" approach
- Model complex water-soil-plant interaction
- Embed Decision Support System (DSS) and disruptive Artificial Intelligence (AI)
- Integration of various knowledge streams
- Fully autonomous

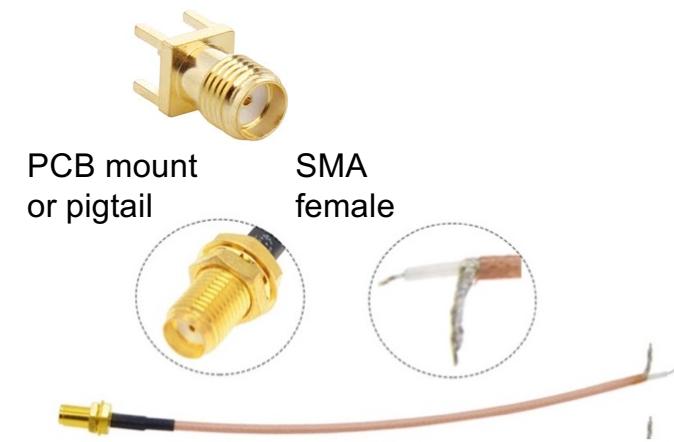
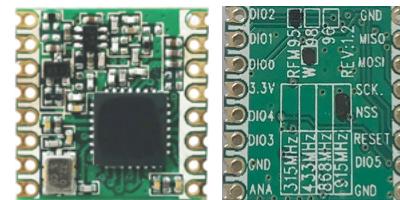
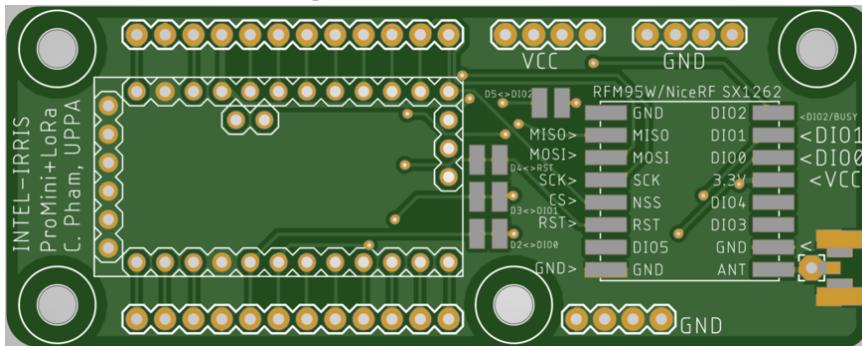


# Review: Starter-kits

- "Intelligent Irrigation in-the-box", "plug-&-sense"
- At least 100 starter-kit will be distributed

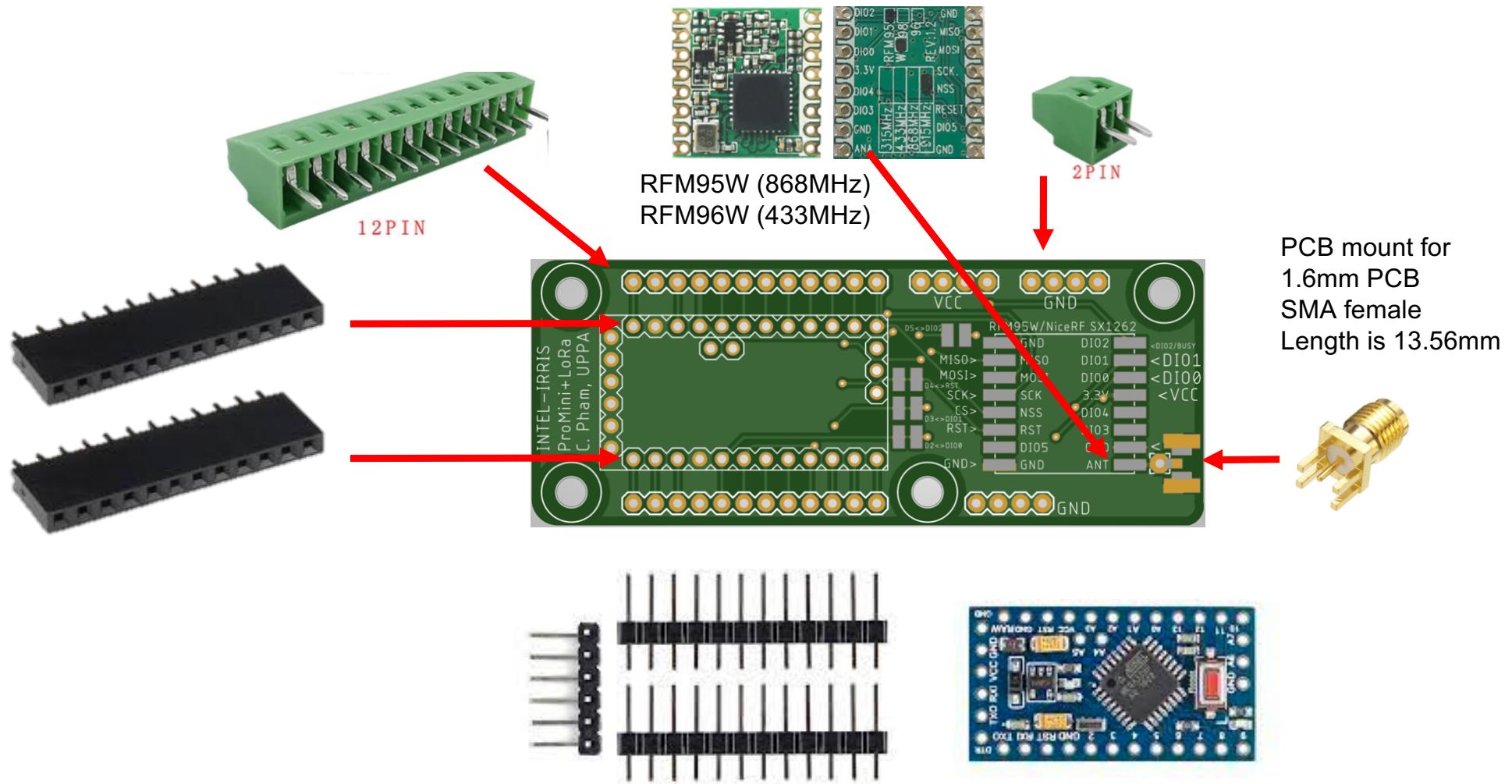


# Soil sensor: electronic parts complete version

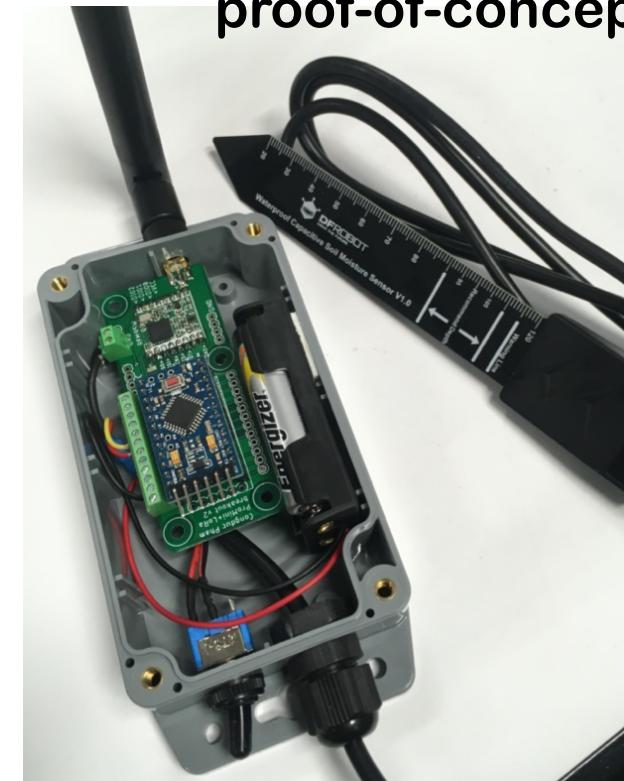


SMA  
male

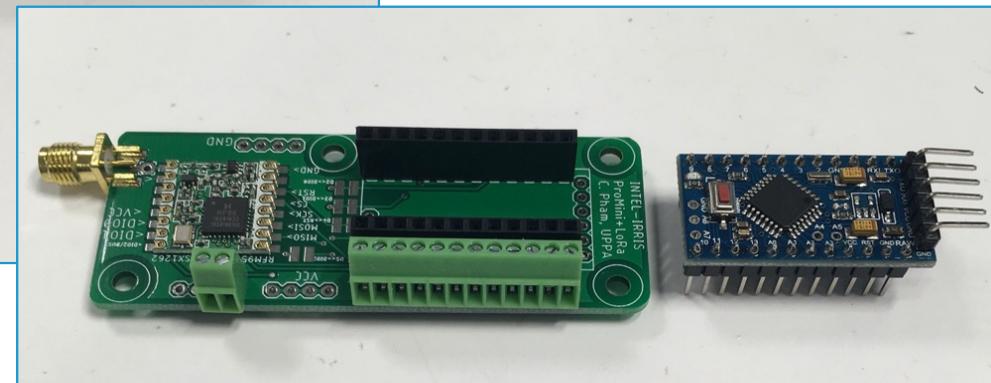
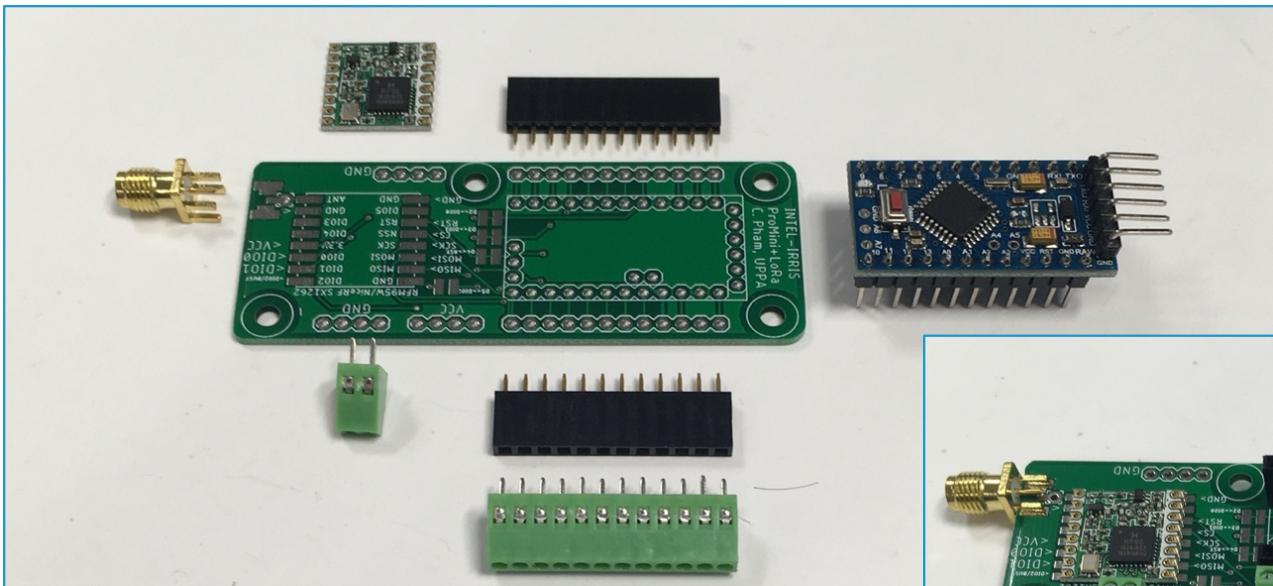
# Soil sensor: electronic parts starter-kit version



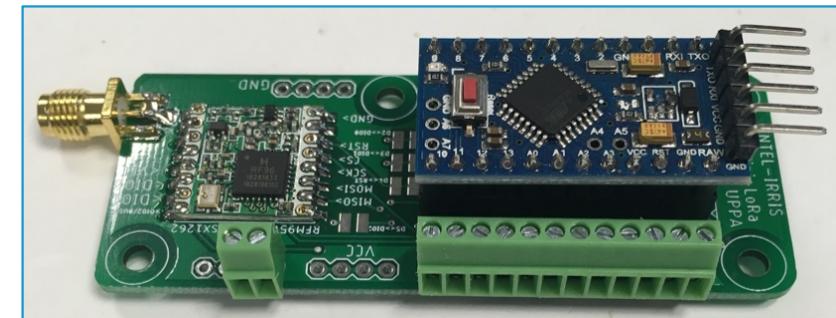
# Soil sensor: enclosure & integration



# Assembling the PCB board

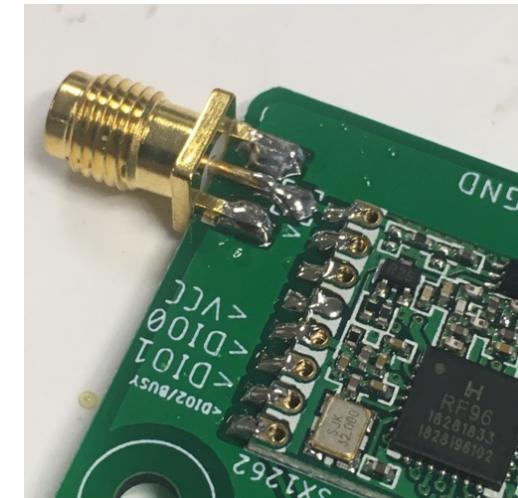
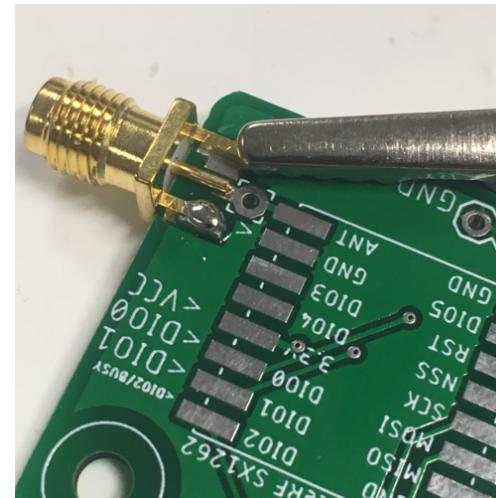
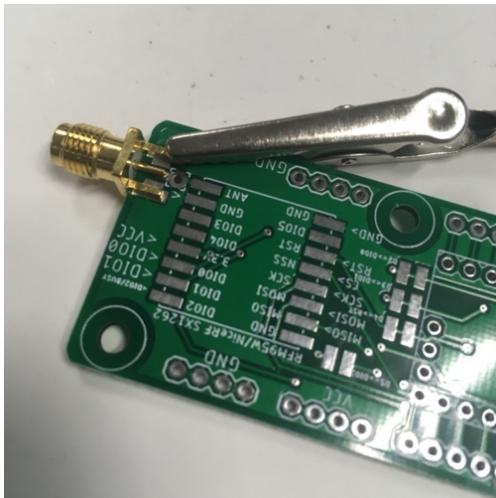


This dedicated video will show  
how to solder all the components  
Video n°1: <https://youtu.be/3jdQ0Uo0phQ>



# Updates

- Here are some updates on Video n°1
- Clip the SMA connector to the PCB to correctly align the connector before soldering one of the GND pad
- Remove clip and finish soldering
- Make sure that the GND pads and the central pin are not touching each other, in doubt, do a continuity check





# Good soldering

- It is important to have good soldering to avoid any issue when deploying the devices
- Check this great soldering tutorial from Collin's lab
  - <https://youtu.be/RB4P1HTmwLc>

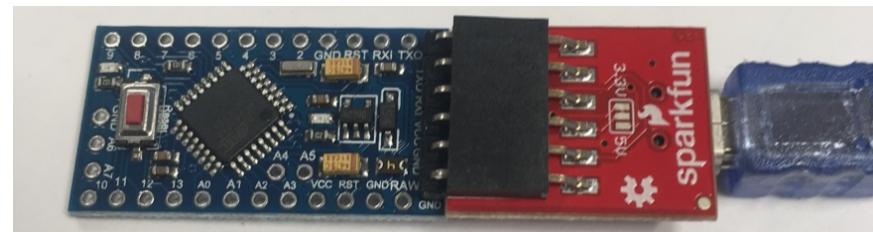
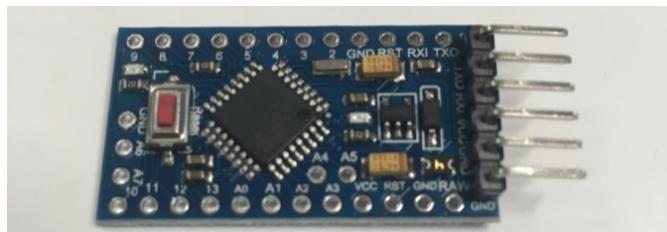


- Just put the required amount of lead, too much lead is not good!
- It is not a bad idea to train on soldering before making the device and not waste electronic components

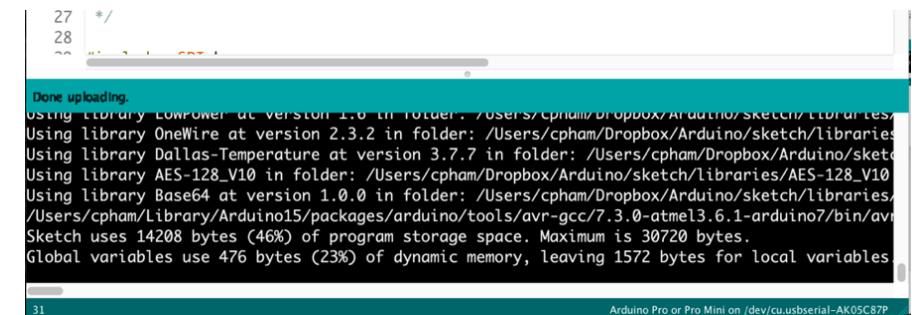


# Check the Arduino

- First, just solder the programming header



- To flash the Arduino board with the INTEL-IRRIS code
- See slides from "Getting the software: Arduino IDE" to "Uploading to your board"
- If uploading is successful, then you can continue with soldering the remaining header pins, otherwise, take another board as this one may have hardware issue



```
27 */  
28  
Done uploading.  
Using library LowPower at version 1.0 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/  
Using library OneWire at version 2.3.2 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/  
Using library Dallas-Temperature at version 3.7.7 in folder: /Users/cpham/Dropbox/Arduino/sketch/  
Using library AES-128_V10 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/AES-128_V10  
Using library Base64 at version 1.0.0 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/  
/Users/cpham/Library/Arduino15/packages/arduino/tools/avr-gcc/7.3.0-atmel3.6.1-arduino7/bin/avr  
Sketch uses 14208 bytes (46%) of program storage space. Maximum is 30720 bytes.  
Global variables use 476 bytes (23%) of dynamic memory, leaving 1572 bytes for local variables  
31  
Arduino Pro or Pro Mini on /dev/cu.usbserial-AK05C87P
```

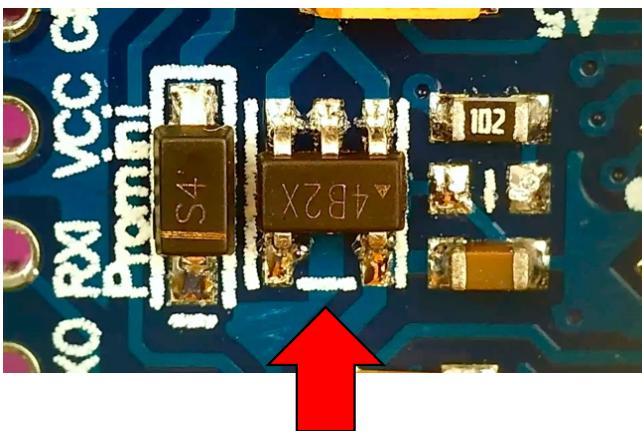
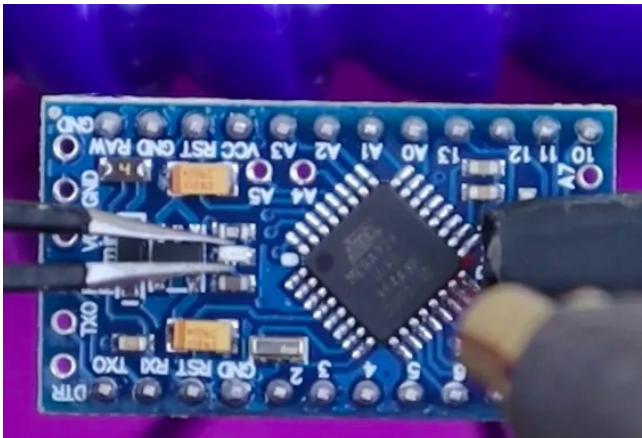


# Never transmit without antenna

- NEVER, NEVER transmit without an antenna
- Doing so can damage the radio module
- If your board is already connected to the radio module and you need to flash the board, connect the antenna
- If you need to update the existing code and your device already run a code that transmit data, connect the antenna
- It is safer when programming the device to remove the Arduino board from the female header and program it disconnected from the radio module
- If you deploy a device, make sure that the antenna is correctly connected before powering on the device and realizing any transmission test

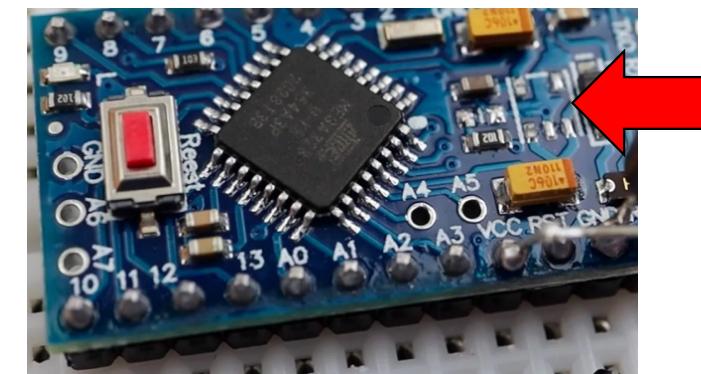


# Reduce power consumption



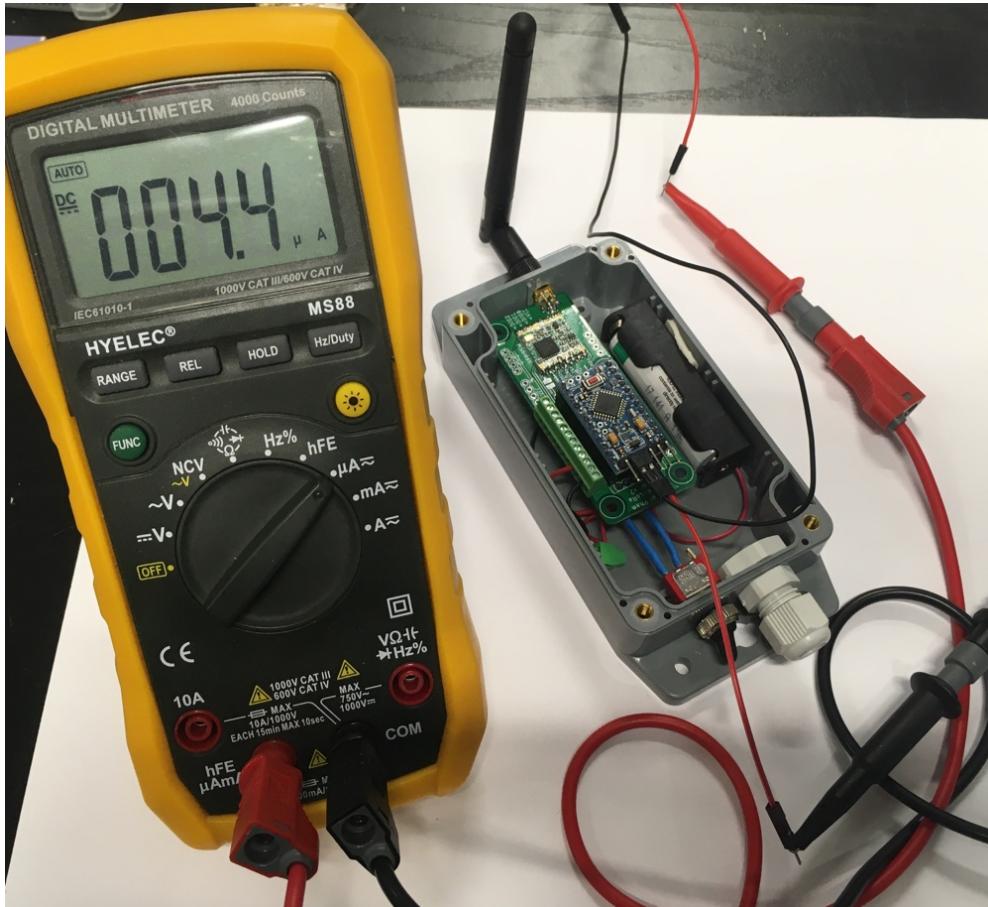
**Do not forget** to remove the power LED by just clipping it off with some wire cutters

**Do not forget** to remove the voltage regulator with a small plier



**Only inject up to 3.6V** through the VCC pin

# Power consumption in deep sleep



Measured below 5 $\mu$ A in deep sleep, between 2 active periods with transmissions

Expected autonomy with 1 transmission / hour is over 2 years with either 2 AA batteries or 1 AA 3.6V Lithium battery

# Get an enclosure for outdoor usage



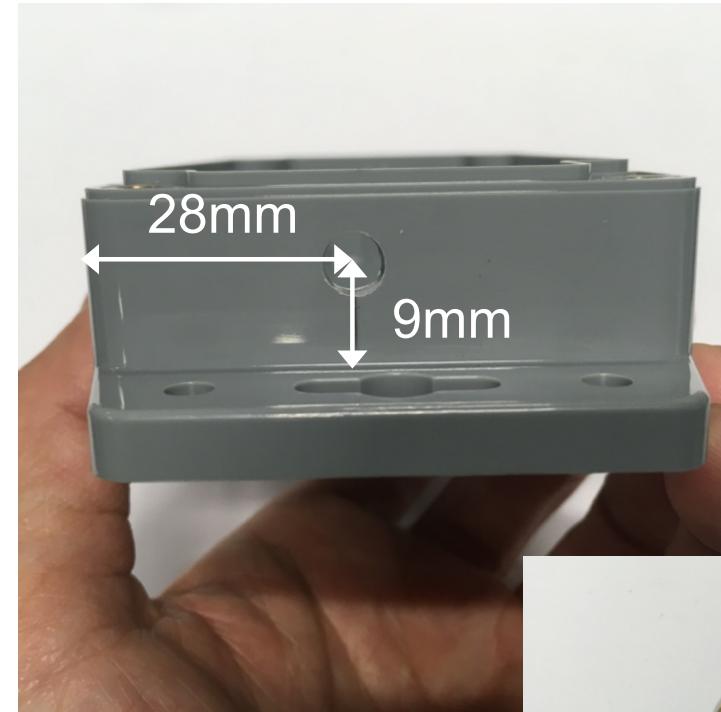
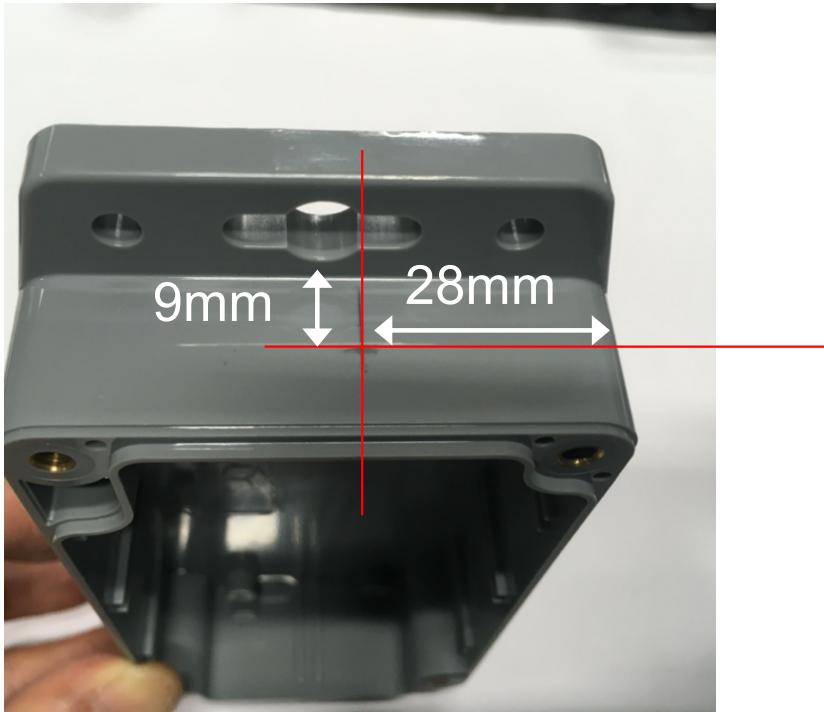
Here, it is an IP65 box which dimension is 115 x 65 x 40mm  
<https://www.gotronic.fr/art-boitier-abs-etanche-g304m-17977.htm>

# Drilling machine and drilling bits



At least a simple cordless drilling machine is necessary  
If you have a (small) bench drilling machine it is of course better  
Then you need an assortment of drilling bits for **metal**, not for  
wood nor concrete! Here, you will mainly need 7mm and 13mm bits  
It is also interesting to have step drill bits

# Drill a hole for the SMA connector



**28mm for the right edge:**

- measure from the flat side as the corner is round

**9mm from the outside bottom**

**use a 7mm drill bit for metal, not for wood nor concrete!**

# Remove unwanted plastic part



we need to remove the plastic reinforcement part for this particular enclosure  
use a flat cutter for instance to remove and smooth the inside part  
(a small plier can be used first to remove most of the part)

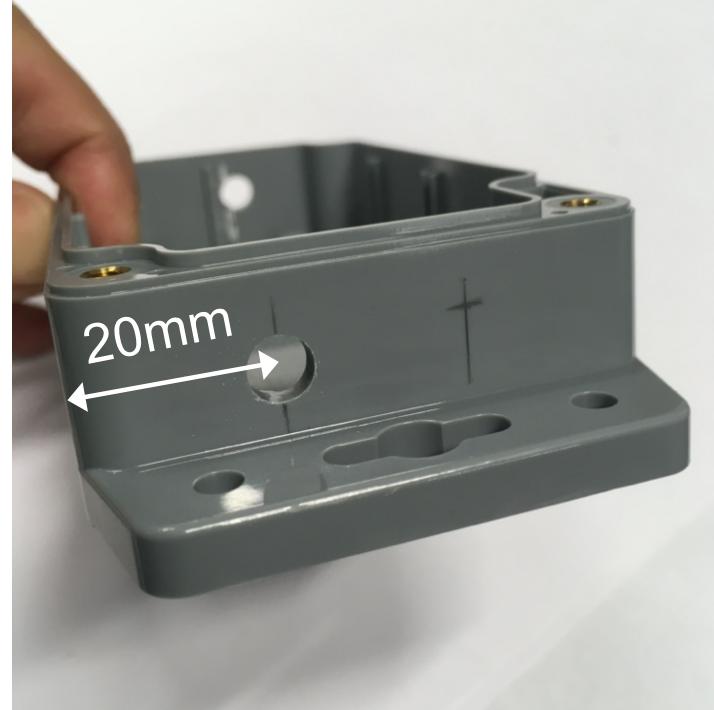
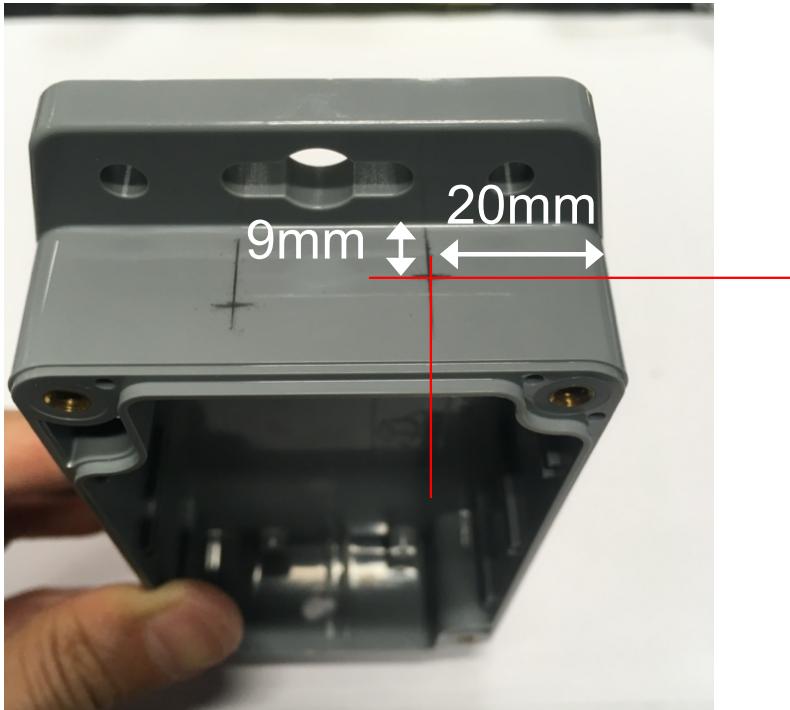
# Placing the PCB board



the PCB board can be placed, with the SMA connector going through the hole

both 2-AA (left) and 1-AA (right) battery holder can be used

# Drill a hole for the external switch



**20mm for the right edge:**

- measure from the flat side as the corner is round

**9mm from the outside bottom**

**use a 7mm drill bit for metal**

# Drill a hole for the cable gland



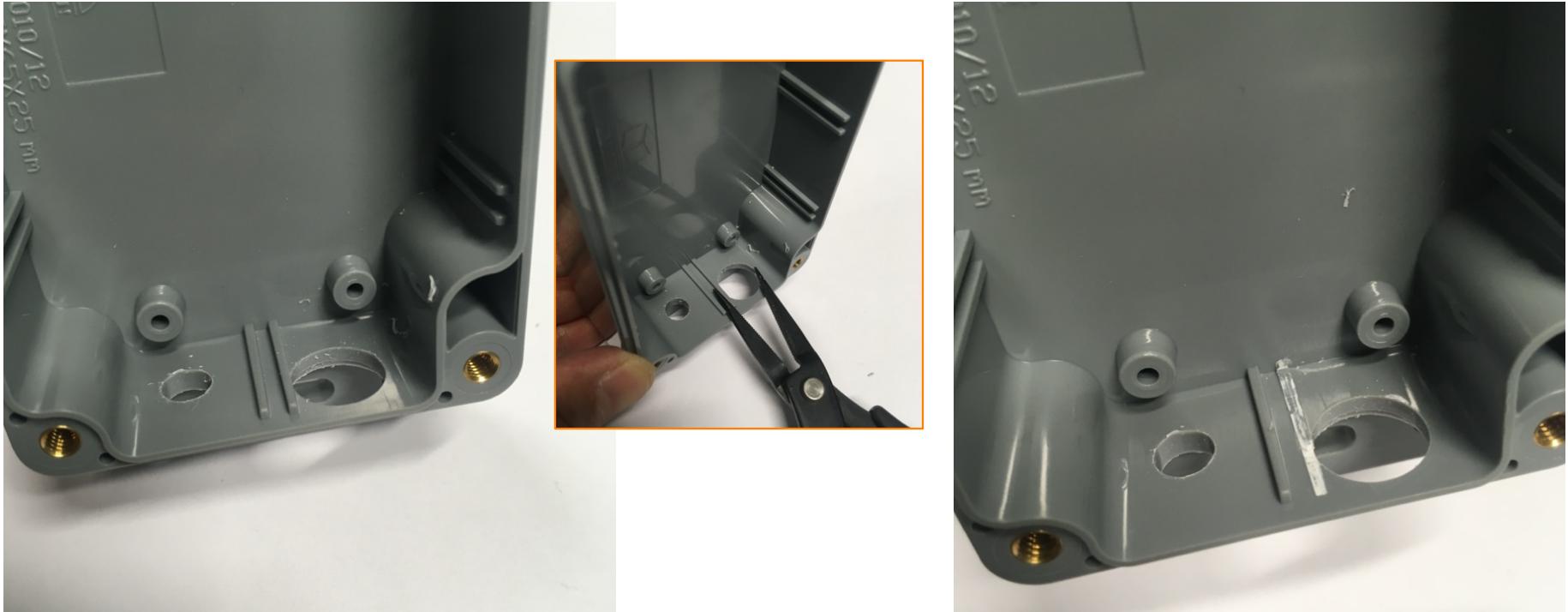
41mm for the right edge:

- measure from the flat side as the corner is round

9mm from the outside bottom

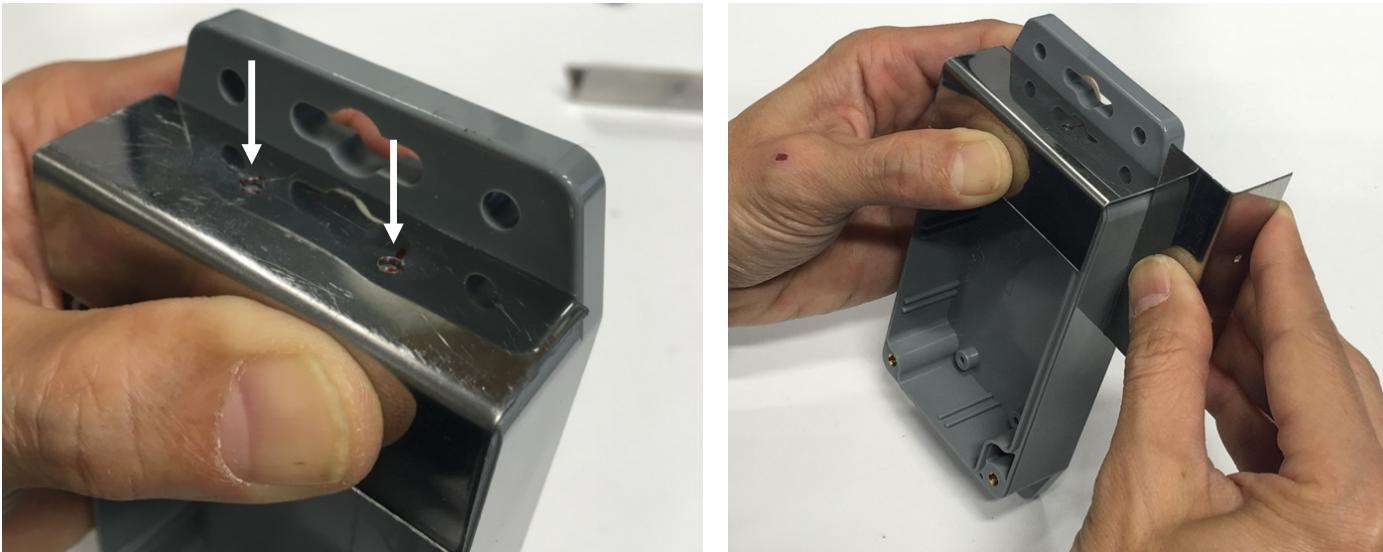
use a 13mm drill bit for metal

# Remove unwanted plastic part



again, we need to remove the plastic reinforcement part for this particular enclosure  
use a flat cutter for instance to remove and smooth the inside part  
(a small plier can be used first to remove most of the part)

# Going for larger production



If you need to prepare many enclosures, it may be faster to first make an assembly jig (here a piece of metal) for marking the holes

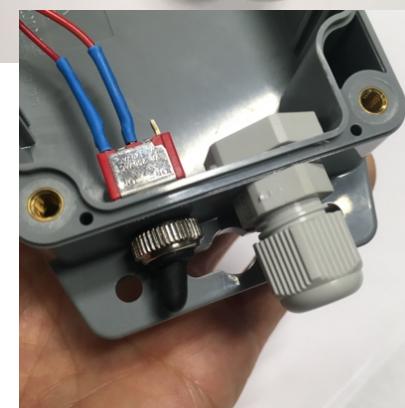
But be sure to precisely align the edge of the jig with the edge of the enclosure

# Placing switch and cable gland

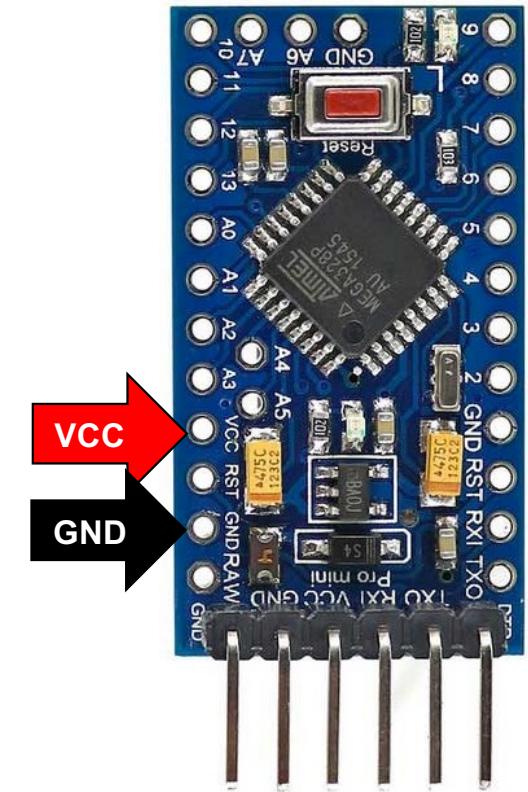
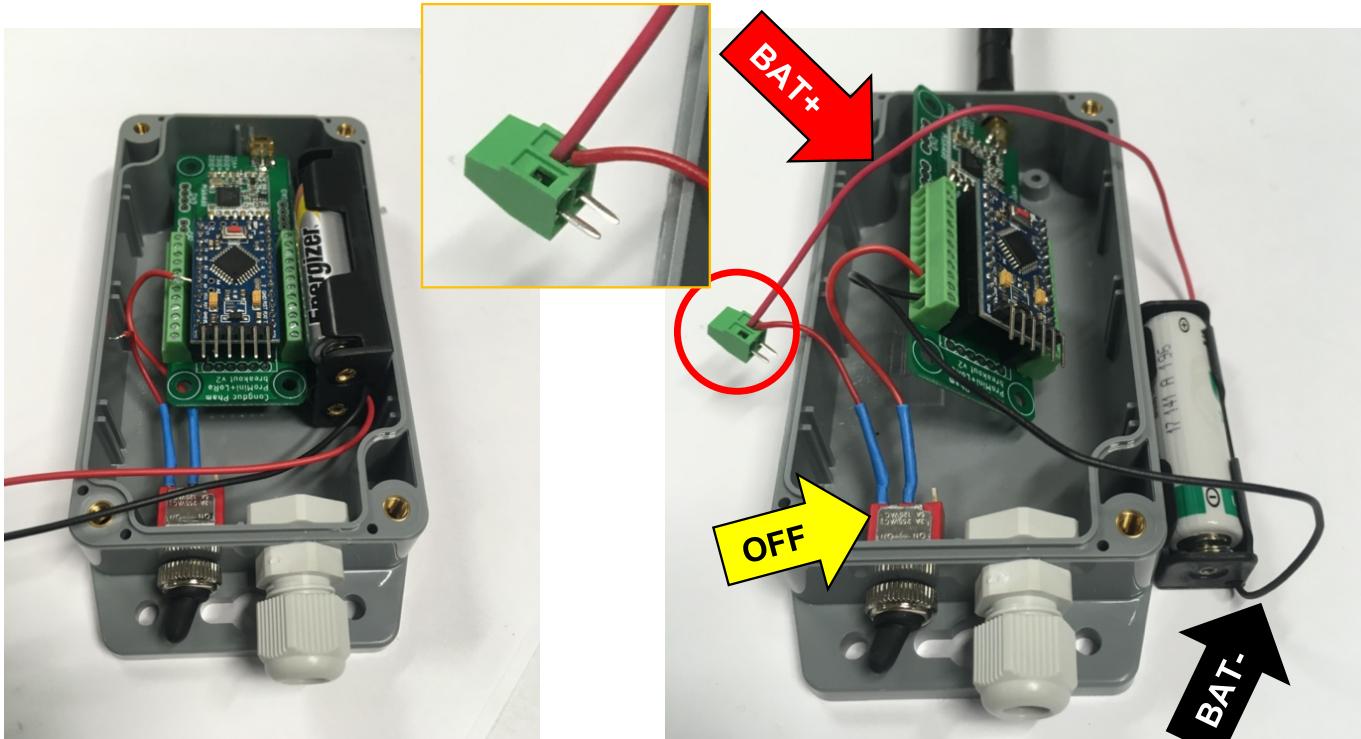


test that everything is OK

the switch has a water-proof cap

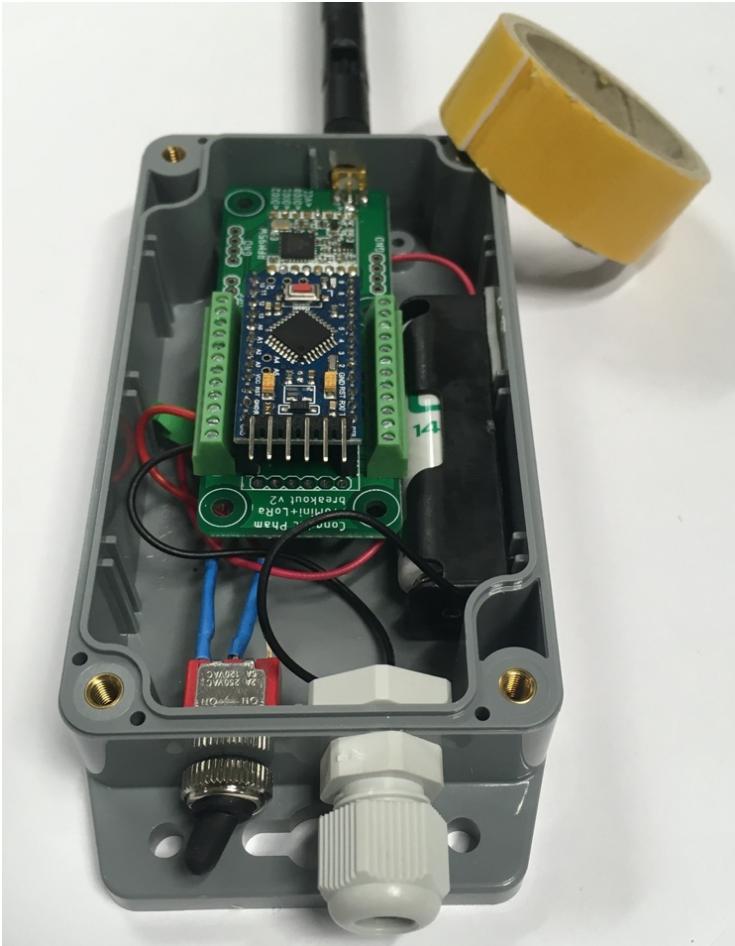


# Connecting switch, battery & board



Connect together switch "off" pin (left) with BAT +  
 Connect directly switch "on" pin (middle) to board's VCC  
 Connect directly BAT – to board's GND  
 Toggling the switch to the right will then power the board

# Putting it altogether

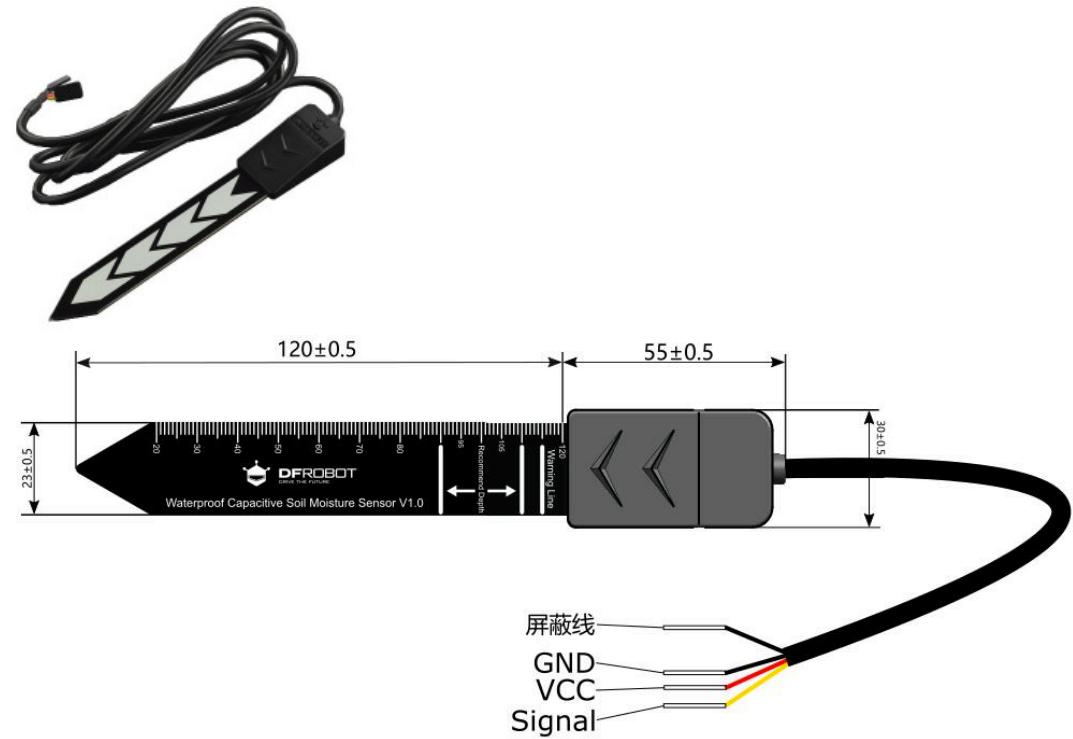


Here we use a 1-AA battery pack  
Fix the battery pack with double-side tape  
e.g. those used to fix mirrors on wall  
Picture shows a 3.6V Lithium battery

This dedicated video will show  
how to build the outdoor soil device  
Video n°2: <https://youtu.be/zcazzDbXvHk>



# Last step: wire the SEN0308



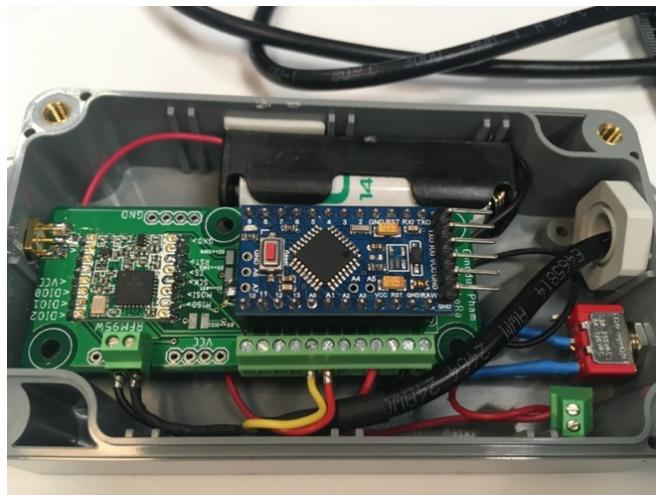
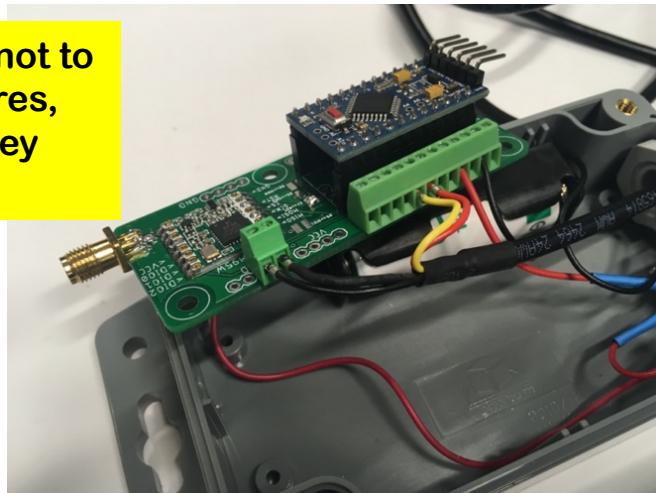
Insert sensor's wire through cable gland

Connect SEN0308's wires to board:

- VCC to board's A1
- GND to board's GND (there are 2 GND wires)
- Signal to board's A0

# The complete soil sensor device

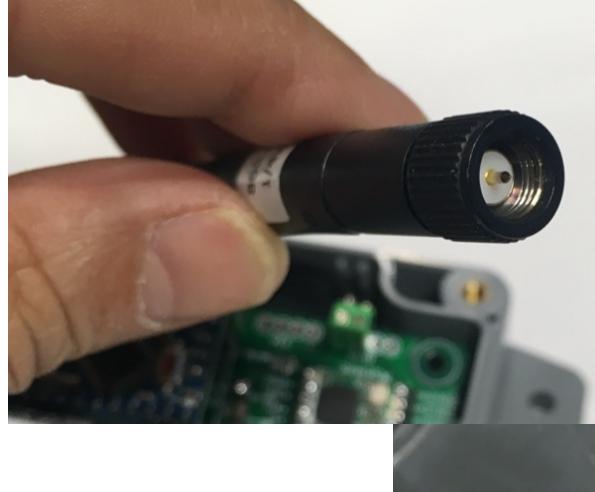
Tighten (but not too much!) all wires, check that they are not loose



Do not forget to tighten the cable gland as well



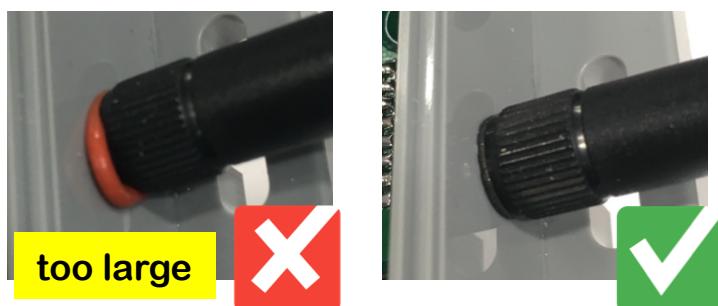
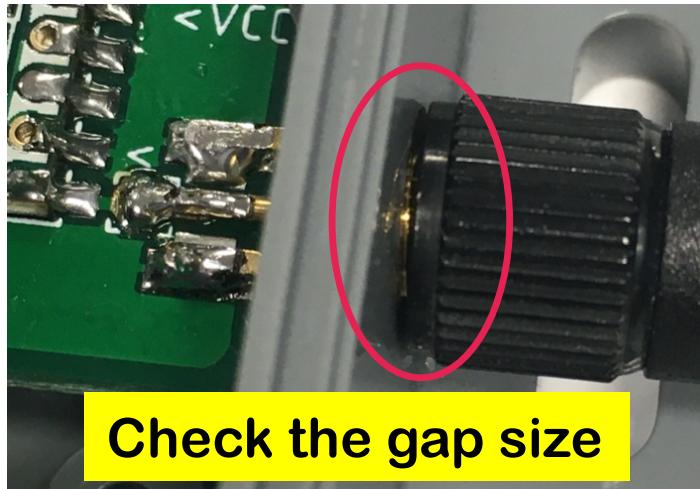
# Connecting antenna



Be sure to connect the matching antenna  
Here, SMA female with SMA male antenna  
Need to screw the antenna in all the way

The antenna junction is **critical because this**  
**is where rain water can come in**

# Waterproofing the antenna junction



Even when the antenna is screwed in all the way, there might still be a gap

Even with no apparent gap, it is necessary to waterproof the junction

Take an o-ring for that purpose, but do not take it too thick or too large!

Too thick: the antenna will not be screwed in all the way!

There are o-ring for SMA connector but the gap size is an indication

# Do not have or can not use o-ring?

Maybe the gap is too big? Use silicon joint sealant



Put small amount of silicon around the antenna junction (use a flat screw driver or other flat tool)



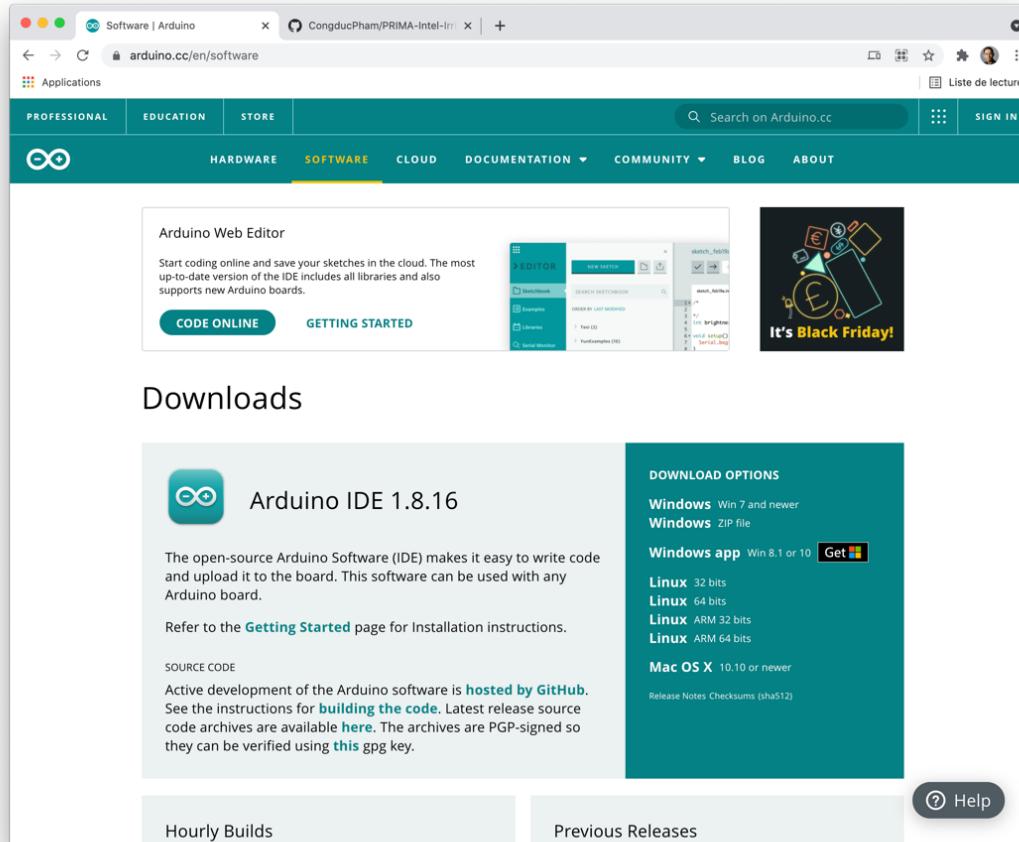
Use a wet toothpick to finish and clean the silicon all around the antenna junction



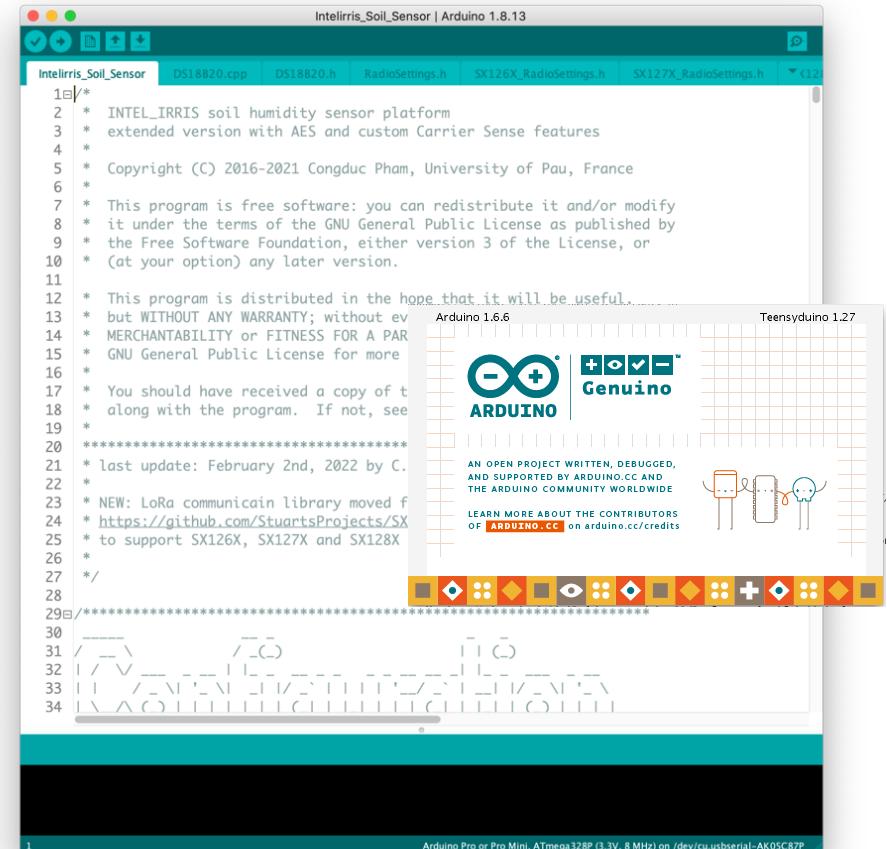
check especially  
the back side

**Even with o-ring, it is good to add silicon!**

# Getting the software: Arduino IDE



The screenshot shows the Arduino website's software download page. It features a header with tabs for PROFESSIONAL, EDUCATION, STORE, HARDWARE, SOFTWARE, CLOUD, DOCUMENTATION, COMMUNITY, BLOG, and ABOUT. Below the header, there's a section for the Arduino Web Editor and a "Downloads" section. The "Downloads" section highlights the Arduino IDE 1.8.16 version, showing download links for Windows, Windows app, Linux, Linux ARM, Mac OS X, and Source Code. There are also links for Hourly Builds and Previous Releases.



The screenshot shows the Arduino IDE interface with the sketch "Intelirris\_Soil\_Sensor" open. The code editor displays the following header comments:

```

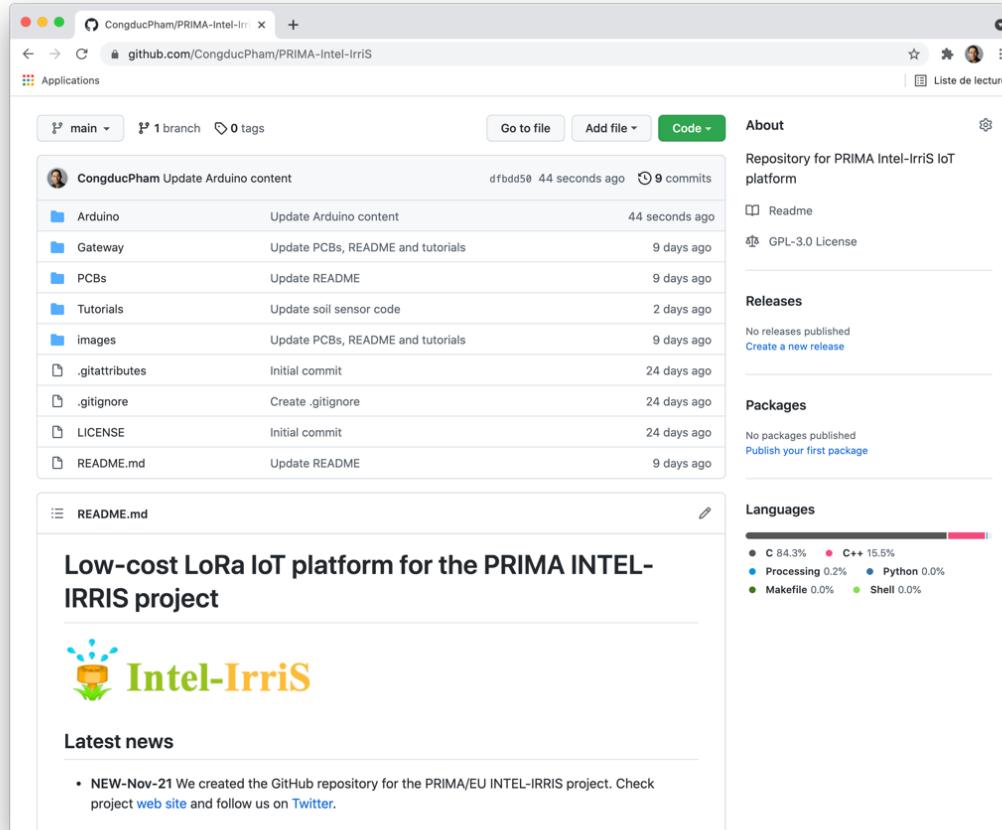
1 //*
2 * INTEL_IRRIS soil humidity sensor platform
3 * extended version with AES and custom Carrier Sense features
4 *
5 * Copyright (C) 2016-2021 Congduc Pham, University of Pau, France
6 *
7 * This program is free software: you can redistribute it and/or modify
8 * it under the terms of the GNU General Public License as published by
9 * the Free Software Foundation, either version 3 of the License, or
10 * (at your option) any later version.
11 *
12 * This program is distributed in the hope that it will be useful,
13 * but WITHOUT ANY WARRANTY; without even
14 * MERCHANTABILITY or FITNESS FOR A PARTICULAR
15 * GNU General Public License for more
16 *
17 * You should have received a copy of the
18 * along with the program. If not, see
19 *
20 ****
21 * last update: February 2nd, 2022 by C.
22 *
23 * NEW: LoRa communication library moved to
24 * https://github.com/StuartsProjects/SX
25 * to support SX126X, SX127X and SX128X
26 *
27 */
28 ****
29 */
30 */
31 */
32 */
33 */
34 */

```

The interface includes toolbars for Arduino 1.6.6 and Teensyduino 1.2.7, and a status bar at the bottom indicating "Arduino Pro or Pro Mini, ATmega328P (3.3V, 8 MHz) on /dev/cu.usbserial-AK05C87P".

Install latest version of Arduino IDE from  
<https://www.arduino.cc/en/software>

# Getting the software: Intel-IrriS code

The screenshot shows the GitHub repository page for CongducPham/PRIMA-Intel-IrriS. The repository has 1 branch and 0 tags. The main commit is by CongducPham, updating Arduino content. The repository is described as a "Repository for PRIMA Intel-IrriS IoT platform". It includes a Readme, a GPL-3.0 License, and no releases or packages published. The Languages section shows C at 84.3%, C++ at 15.5%, Processing at 0.2%, Python at 0.0%, Makefile at 0.0%, and Shell at 0.0%. The README.md file describes the Low-cost LoRa IoT platform for the PRIMA INTEL-IRRIS project.

On your computer, create a sketch folder

Then download the whole repository as ZIP file

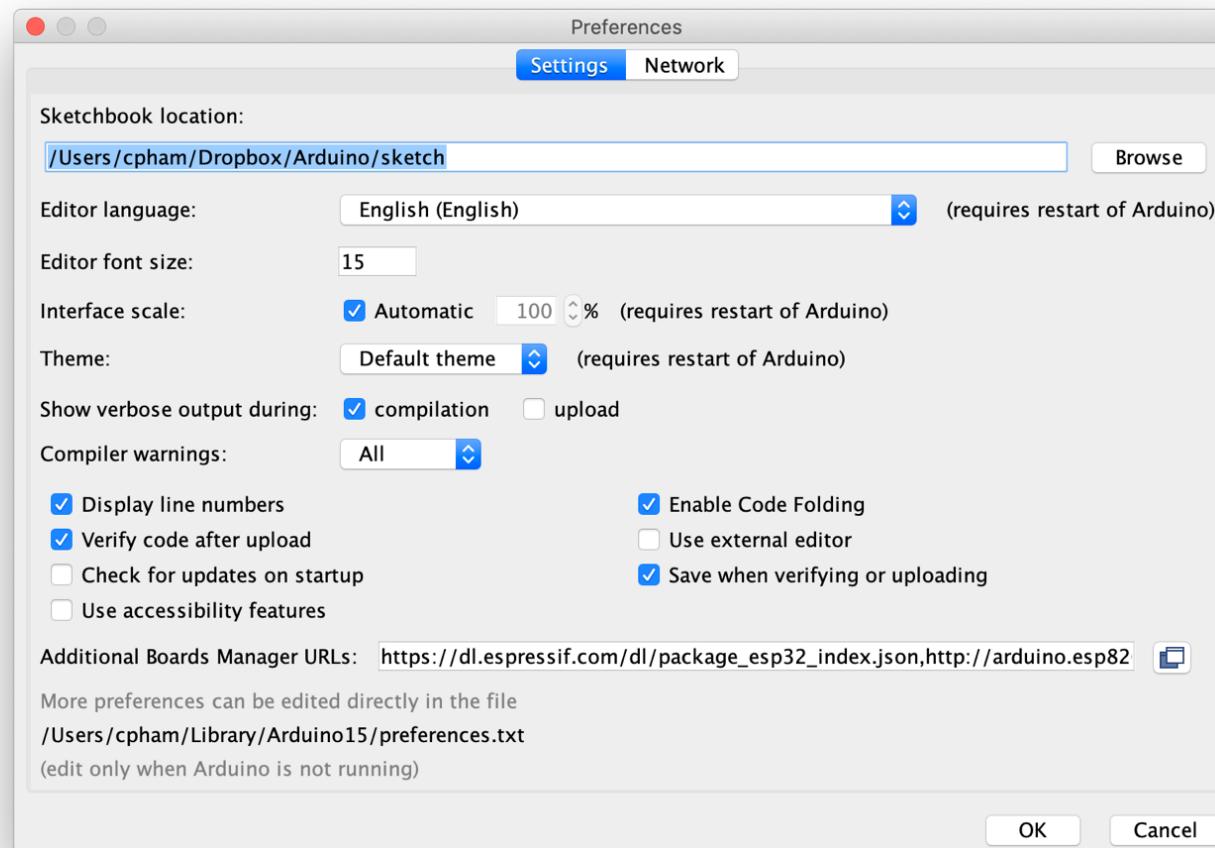
Unzip the file and copy the content of Arduino folder into your sketch folder

The entire Intel-IrriS GitHub repository is hosted here  
<https://github.com/CongducPham/PRIMA-Intel-IrriS>



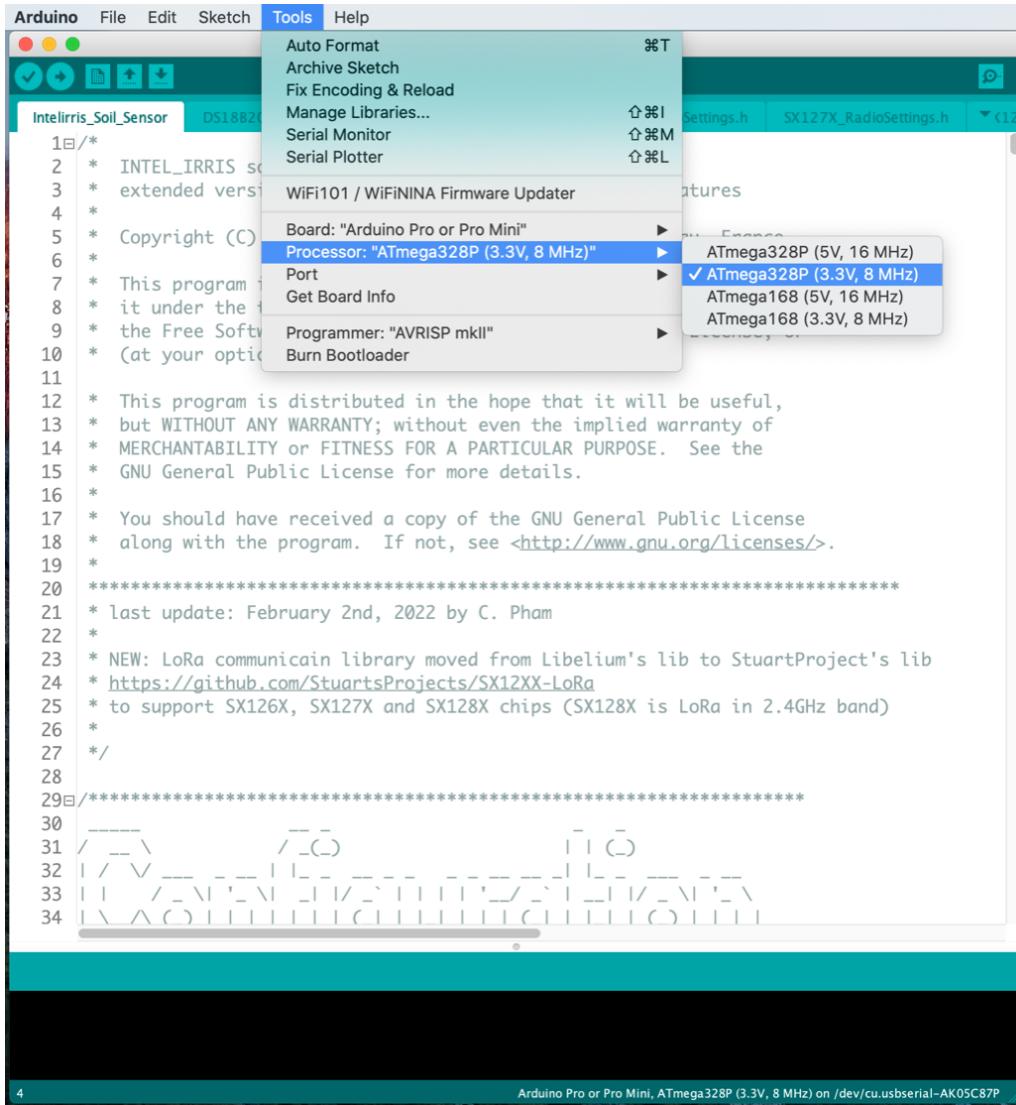
**GitHub**

# Setting your Arduino IDE



Run Arduino IDE, open Preferences  
 Indicate your sketch folder in Sketchbook location

# Compiling the soil sensor code



The screenshot shows the Arduino IDE interface. The sketch titled "Intelirris\_Soil\_Sensor" is open. The "Tools" menu is pulled down, showing various options like Auto Format, Archive Sketch, and Board selection. Under Board selection, it lists "Board: Arduino Pro or Pro Mini" and "Processor: ATmega328P (3.3V, 8 MHz)". A dropdown menu shows four options: ATmega328P (5V, 16 MHz), ATmega328P (3.3V, 8 MHz) (which is checked), ATmega168 (5V, 16 MHz), and ATmega168 (3.3V, 8 MHz). The main code area displays the source code for the soil sensor sketch.

```

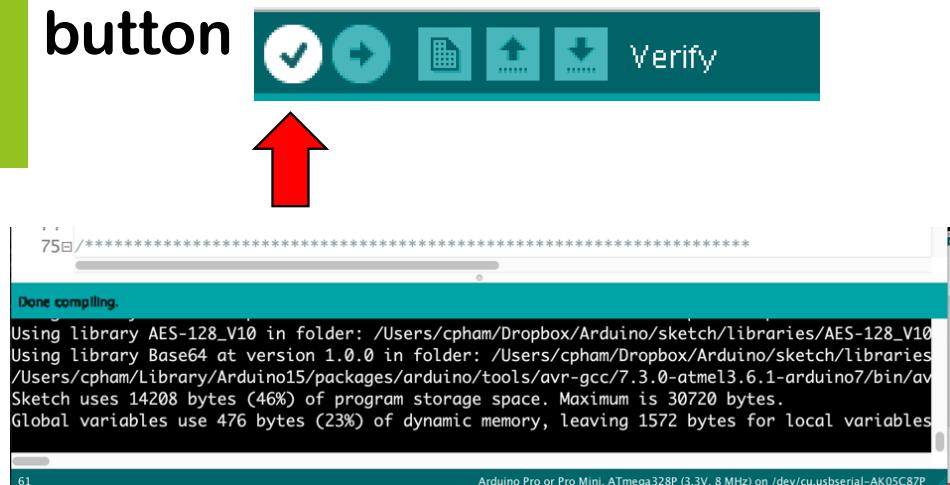
1 ///*
2 *  INTEL_IRRIS
3 *  extended version
4 *
5 *  Copyright (C)
6 *  This program is
7 *  it under the
8 *  the Free Software
9 *  (at your option)
10 *
11 *  This program is distributed in the hope that it will be useful,
12 *  but WITHOUT ANY WARRANTY; without even the implied warranty of
13 *  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
14 *  GNU General Public License for more details.
15 *
16 *  You should have received a copy of the GNU General Public License
17 *  along with the program. If not, see <http://www.gnu.org/licenses/>.
18 *
19 ****
20 * last update: February 2nd, 2022 by C. Pham
21 *
22 * NEW: LoRa communicain library moved from Libelium's lib to StuartProject's lib
23 * https://github.com/StuartsProjects/SX12XX-LoRa
24 * to support SX126X, SX127X and SX128X chips (SX128X is LoRa in 2.4GHz band)
25 *
26 */
27 */
28 */
29 */
30 */
31 */
32 */
33 */
34 */

```

**Open Intelirris\_Soil\_Sensor sketch – no change required**

**Select the ProMini board,  
3.3V and 8MHz version**

**Then click on the "verify"  
button**

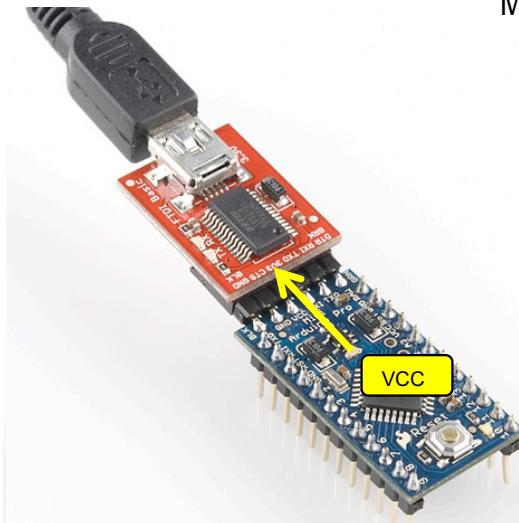




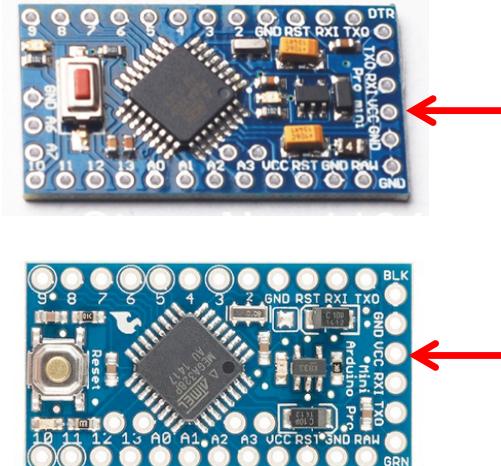
# Never transmit without antenna

- NEVER, NEVER transmit without an antenna
- Doing so can damage the radio module
- If your board is already connected to the radio module and you need to flash the board, connect the antenna
- If you need to update the existing code and your device already run a code that transmit data, connect the antenna
- It is safer when programming the device to remove the Arduino board from the female header and program it disconnected from the radio module
- If you deploy a device, make sure that the antenna is correctly connected before powering on the device and realizing any transmission test

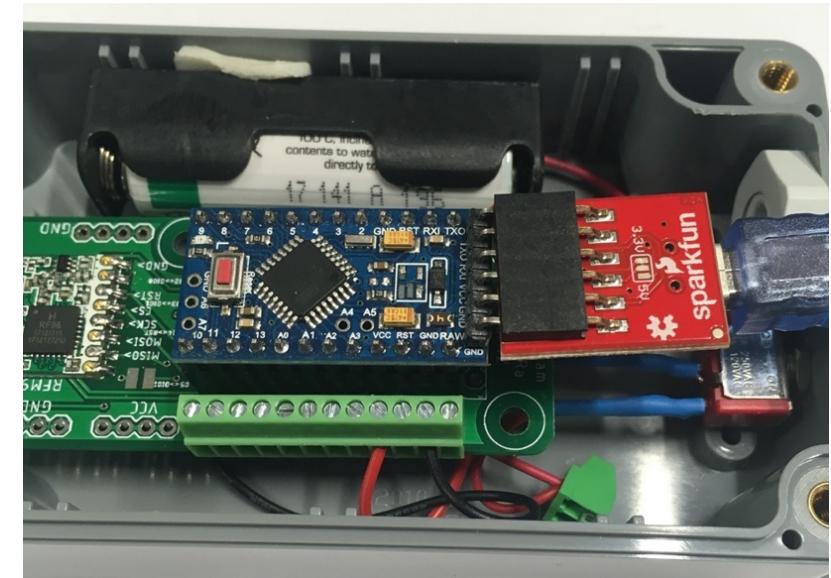
# Connecting with an FTDI cable



Most Chinese clone version, check the VCC pin



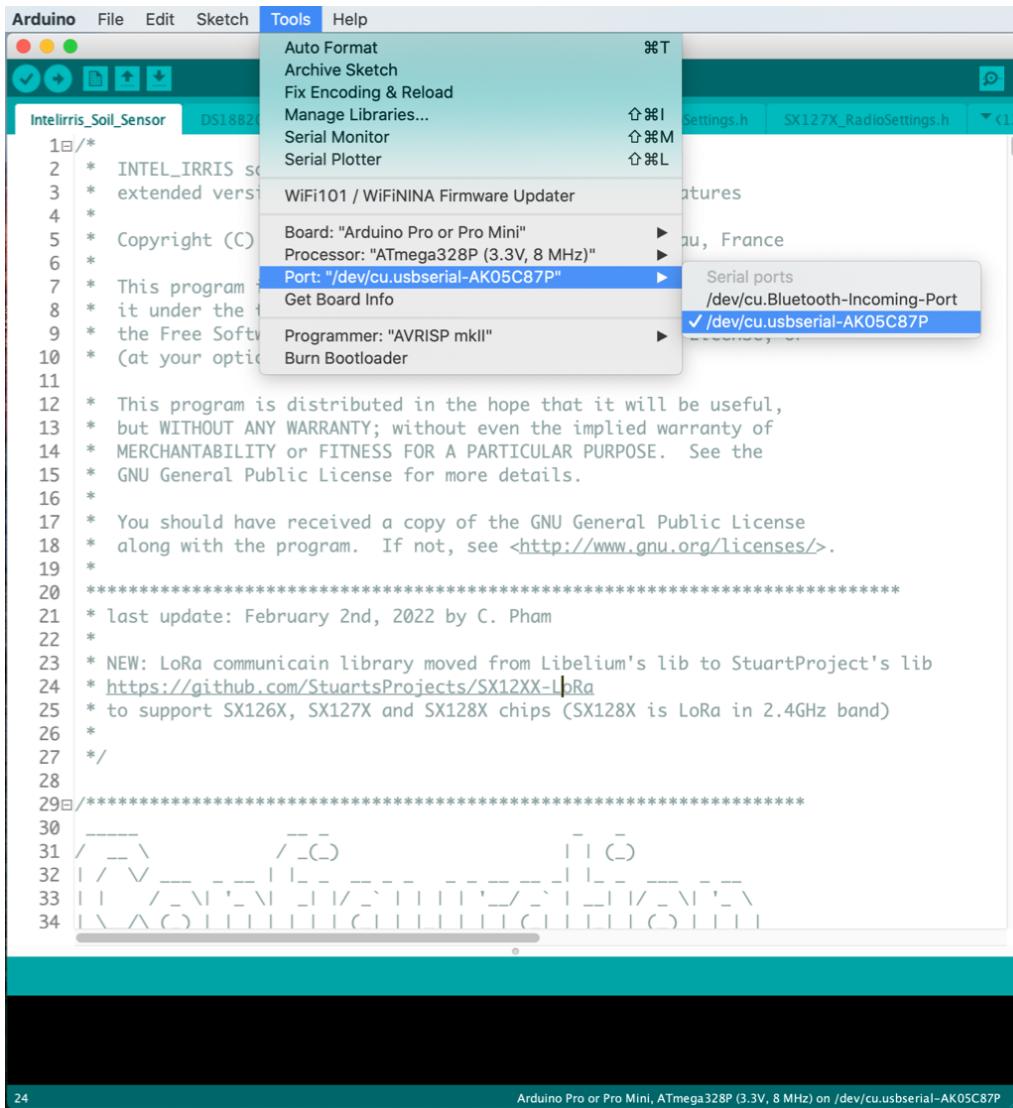
Original Sparkfun  
version



For the ProMini, you need to have an FTDI breakout cable working at 3.3v

Check the VCC pin position and make it to correspond to the VCC pin of the FTDI breakout.

# Select serial port for uploading



After connecting the cable to your computer/laptop USB port, try to find the serial port

If you don't find it, you may need to install specific drivers

<https://learn.sparkfun.com/tutorials/how-to-install-ch340-drivers/all>

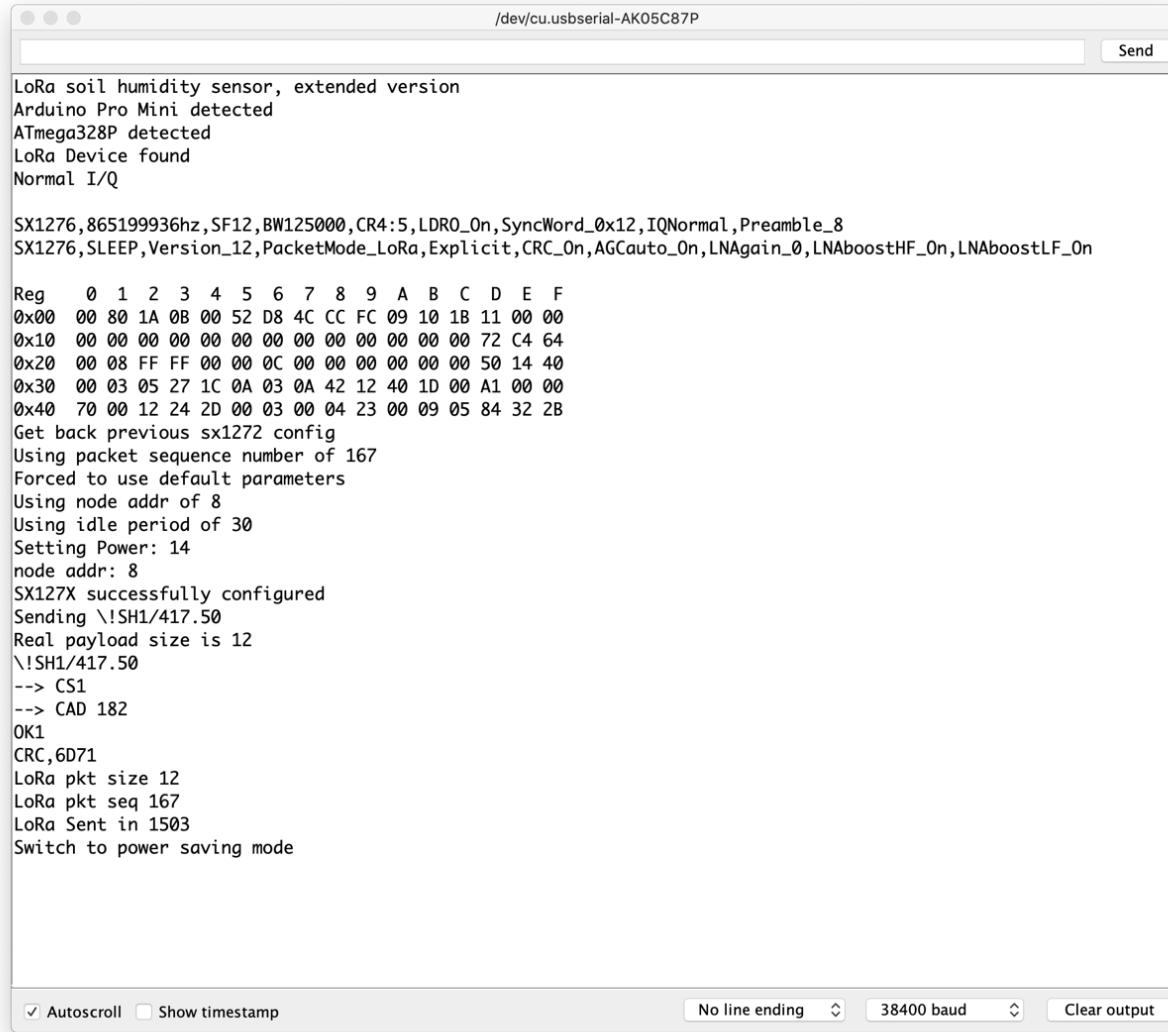
**Click on the "upload" button**



And wait until upload is completed

```
27 */  
28  
29  
30  
  
Done uploading.  
Using library LowPower at version 1.0 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/  
Using library OneWire at version 2.3.2 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/  
Using library Dallas-Temperature at version 3.7.7 in folder: /Users/cpham/Dropbox/Arduino/sketch/  
Using library AES-128_V10 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/AES-128_V10  
Using library Base64 at version 1.0.0 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/  
/Users/cpham/Library/Arduino15/packages/arduino/tools/avr-gcc/7.3.0-atmel3.6.1-arduino7/bin/avr  
Sketch uses 14208 bytes (46%) of program storage space. Maximum is 30720 bytes.  
Global variables use 476 bytes (23%) of dynamic memory, leaving 1572 bytes for local variables
```

# Checking that device is operational



The screenshot shows a terminal window titled '/dev/cu.usbserial-AK05C87P' displaying the output of a LoRa soil humidity sensor. The output includes:

- Device detection: LoRa soil humidity sensor, extended version; Arduino Pro Mini detected; ATmega328P detected; LoRa Device found; Normal I/Q.
- Radio parameters: SX1276, 865199936hz, SF12, BW125000, CR4:5, LDR0\_On, SyncWord\_0x12, IQNormal, Preamble\_8.
- Configuration: SX1276, SLEEP, Version\_12, PacketMode\_LoRa, Explicit, CRC\_On, AGCAuto\_On, LNAGain\_0, LNABoostHF\_On, LNABoostLF\_On.
- Registers (Reg 0-15): Hex values for each register.
- Configuration steps: Get back previous sx1272 config, Using packet sequence number of 167, Forced to use default parameters, Using node addr of 8, Using idle period of 30, Setting Power: 14, node addr: 8.
- Transmission: SX127X successfully configured, Sending \!SH1/417.50, Real payload size is 12, \!SH1/417.50.
- Status: --> CS1, --> CAD 182, OK1, CRC, 6D71.
- Power saving: LoRa pkt size 12, LoRa pkt seq 167, LoRa Sent in 1503, Switch to power saving mode.

At the bottom of the window are checkboxes for 'Autoscroll' and 'Show timestamp', and dropdown menus for 'No line ending' and '38400 baud' along with a 'Clear output' button.

**Open serial monitor**

**Set baud rate to 38400**

**See output from board**

**Check that  
transmission is OK**

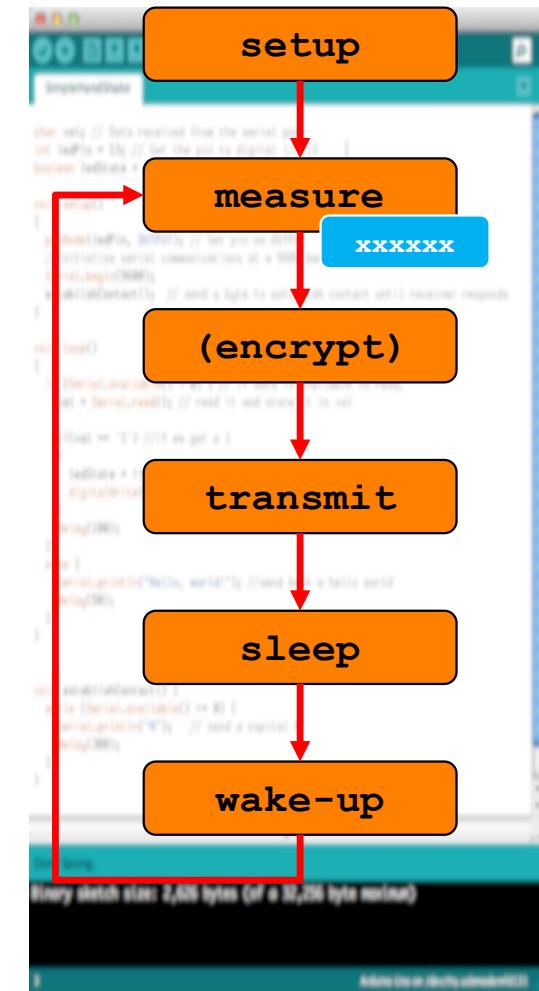
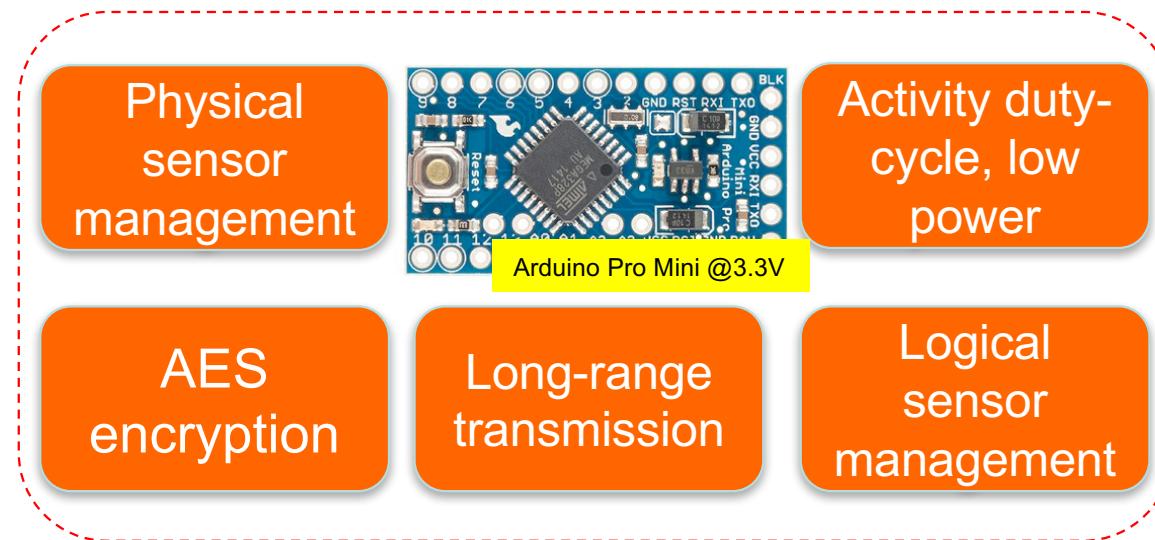
# Generic & cyclic behavior

2 sensors can be attached

SEN0308: soil humidity  
 A0 (signal), A1 (pwr)



DS18B20: soil temperature  
 6 (signal), 7 (pwr)



# Transmission to non-LoRaWAN gateway



Default parameters

raw LoRa  
SF12BW125  
865.2 MHz  
node id is 8  
1 msg/60mins  
1 sensor  
ex: \! SH1/750.50



This dedicated video will show all these steps, from wiring the SEN0308 to testing transmission to the gateway  
Video n°3: <https://youtu.be/zcazzDbXvHk>



# Transmission to WaziGate



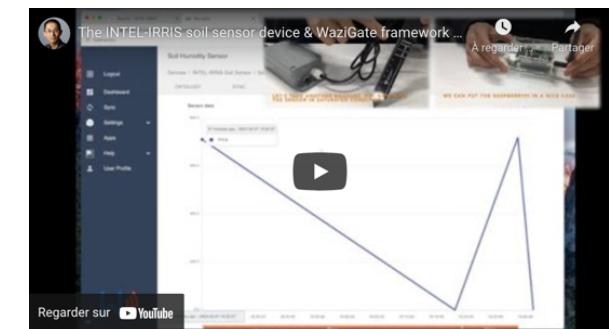
Parameters for  
INTEL-IRRIS WaziGate



SF12BW125  
868.1MHz | 433.175MHz  
Node id is 26011DAA  
1 msg/60mins  
1 sensor  
XLPP data

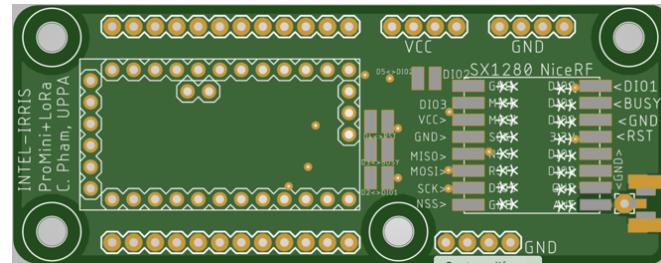


This dedicated video will show all these steps, from connecting the SEN0308 to testing transmission to the WaziGate  
Video n°4: <https://youtu.be/j-1Nk0tv0xM>



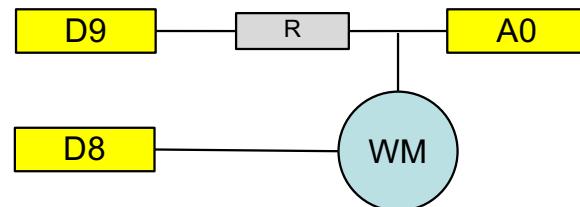
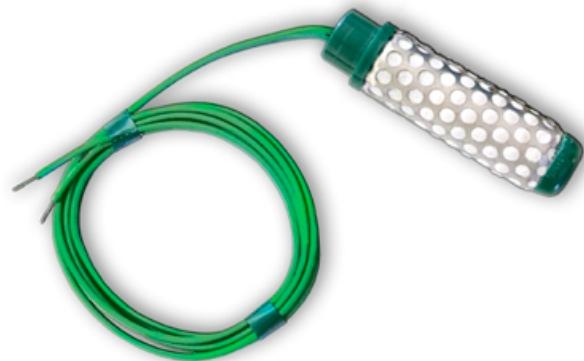
# Advanced options

- The Intelirris\_Soil\_Sensor sketch can
  - Drive SX127X, SX126X and SX128X LoRa chips
  - SX128X requires a dedicated PCB for NiceRF SX1280



- Send AES 128-bit encrypted packet
- Send uplink LoRaWAN packet (encrypted)
- Receive downlink LoRaWAN packet
- Can read additionally from a DS18B20 soil temperature sensor
- Can read from a Watermark tensiometer sensor
- Support WaziSense and WaziDev boards

# With a Watermark sensor



with a Watermark sensor, the "pseudo-AC Short Pulse" method will be used – see <https://www.irrometer.com/200ss.html>  
 D9 and D8 will be used to alternating power the sensor  
 A0 will be used to read signal from sensor

# 2 versions of the soil device





Starter-kit

Autonomous

Intelligent Irrigation

Plug-&-Sense

In-the-box