

# 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 (using the latest IRD PCB v4.1)



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

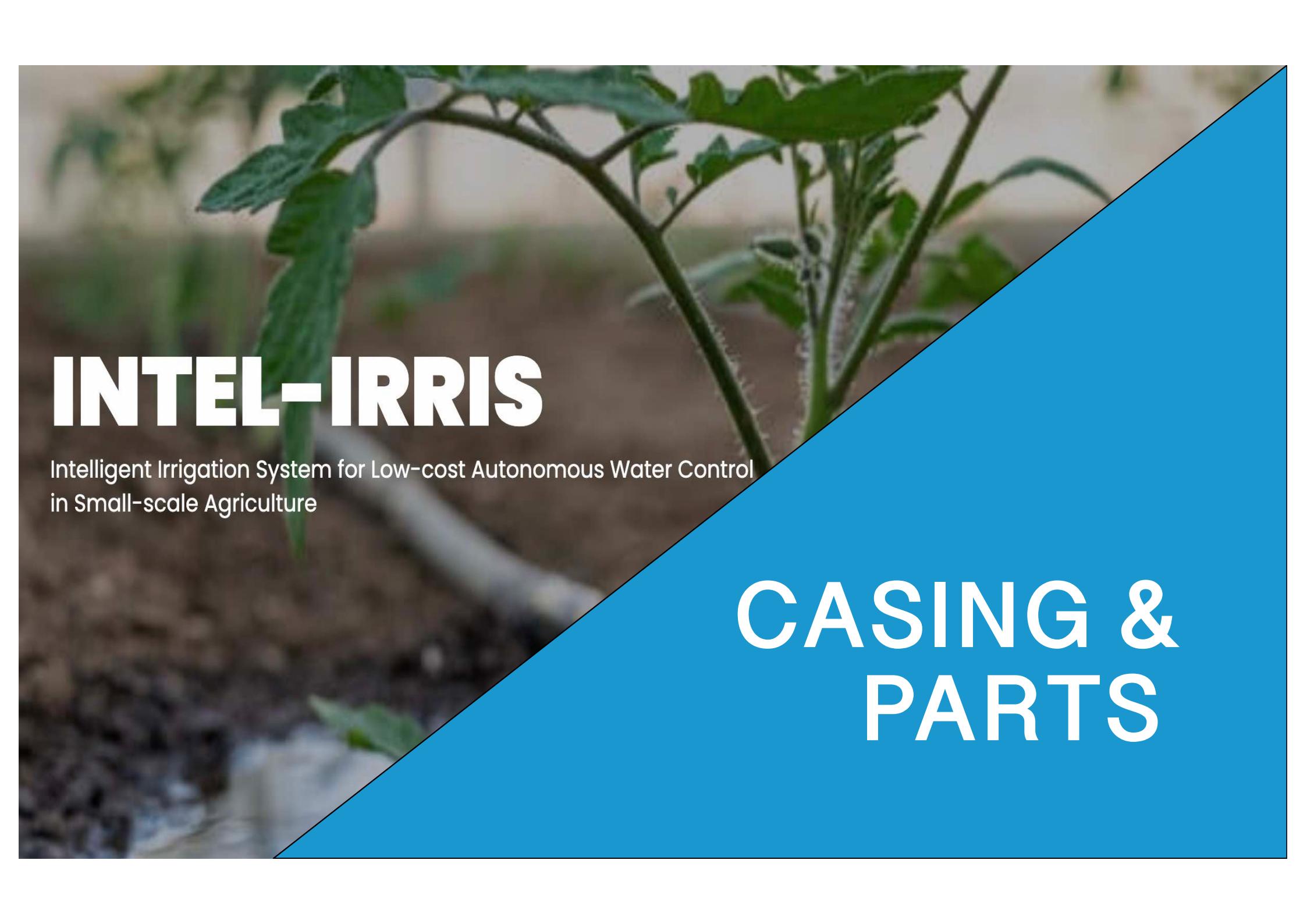
- Build on low-cost embedded & open IoT gateway expertise
- Implement the “Intelligent Irrigation in-the-box” with "plug-&-sense" approach
- Model complex water-soil-plant interaction
- Embed Decision Support System (DSS) and disruptive Artificial Intelligence (AI)
- Integration of various knowledge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



# INTEL-IRRIS's starter-kit

○ From idea to reality!



A close-up photograph of a young green plant with several leaves. In the background, a white plastic pipe lies on the brown soil, with a thin black tube connected to it, likely part of an irrigation system.

# INTEL-IRRIS

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

# CASING & PARTS

# List of components for IRD PCB



**For AA  
alkaline  
batteries**



**For SOLAR  
with NiMh**



**For SOLAR**

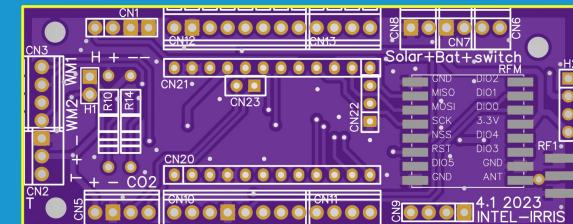


SMA  
male



# INTEL-IRRIS

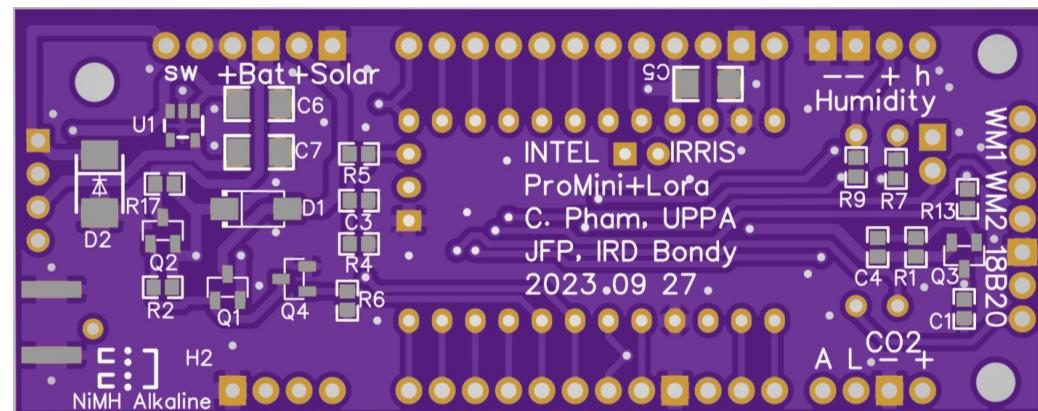
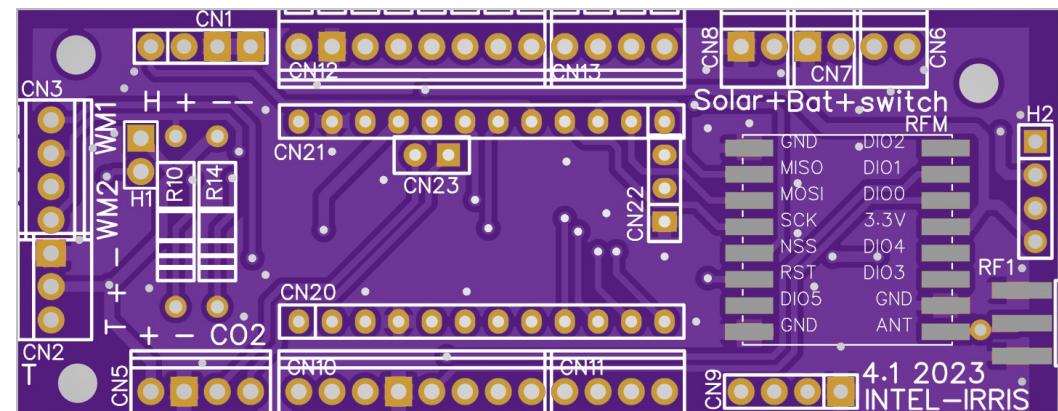
# Intelligent Irrigation System for Low-cost Autonomous Water Control in Small-scale Agriculture

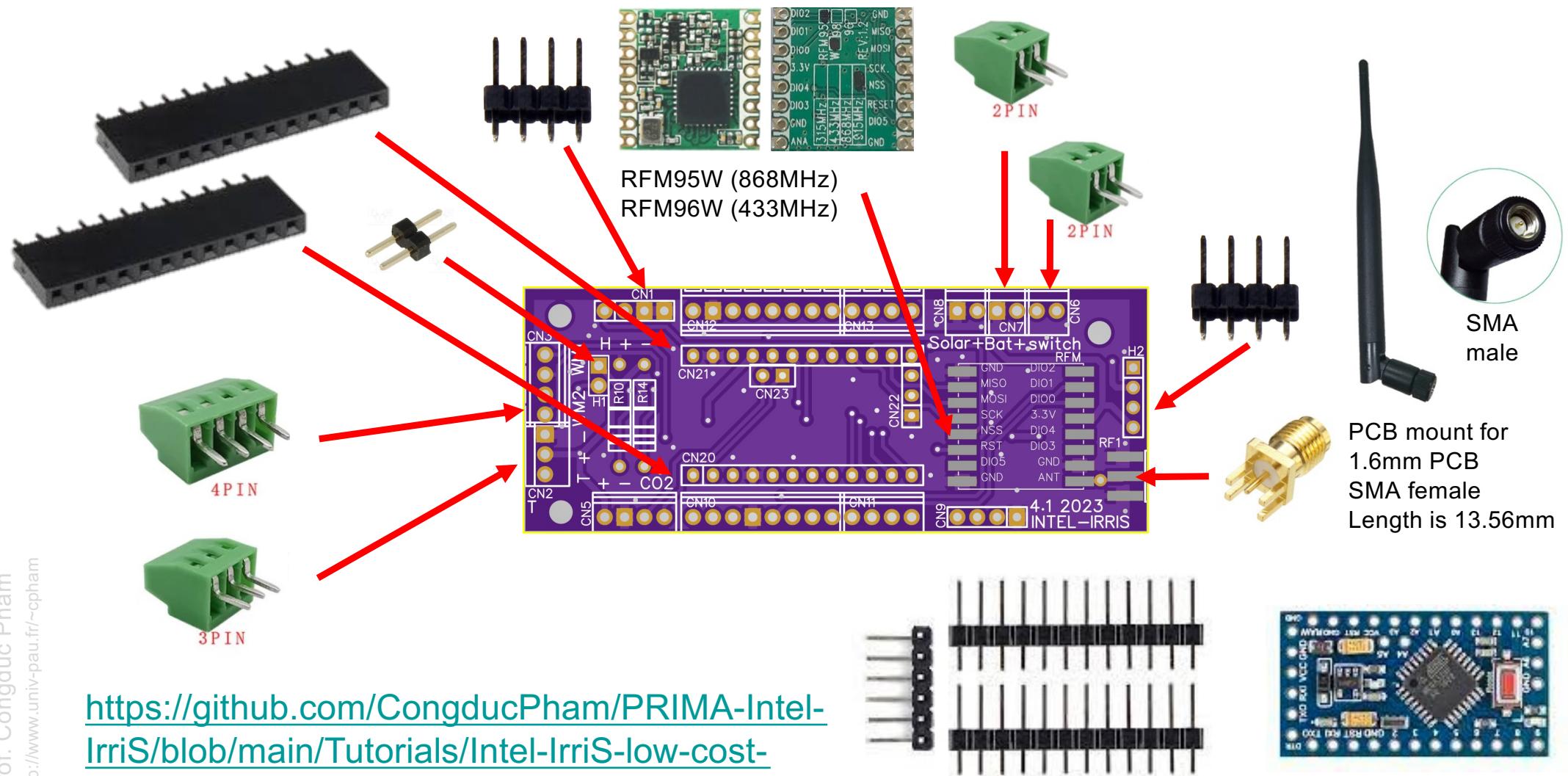


# LATEST IRD PCB V4.1

# The IRD PCB v4.1 – raw version

- You can order the raw version of the IRD PCB, which means only the PCB, without any electronic components soldered by the manufacturer
- It is the so-called DIY approach
- The raw version is not intended to have solar charging capabilities





<https://github.com/CongducPham/PRIMA-Intel-IrriS/blob/main/Tutorials/Intel-IrriS-low-cost-sensor-hardware-parts.pdf>

Arduino ProMini 3.3V 8MHz

# Tutorial video

- This dedicated video will show how to solder all the components  
Video n°1: <https://youtu.be/3jdQ0Uo0phQ>
- The "old" version of the INTEL-IRRIS PCB is shown on the video but you can easily adapt for the new IRD PCB v4.1





# Good soldering

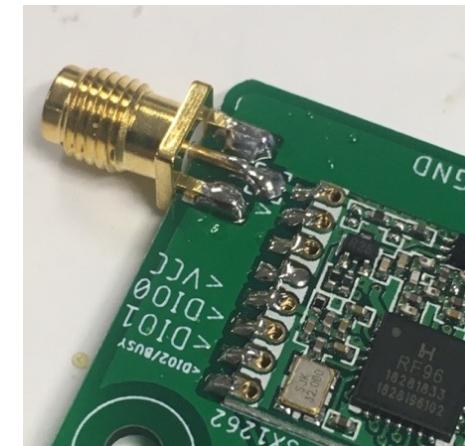
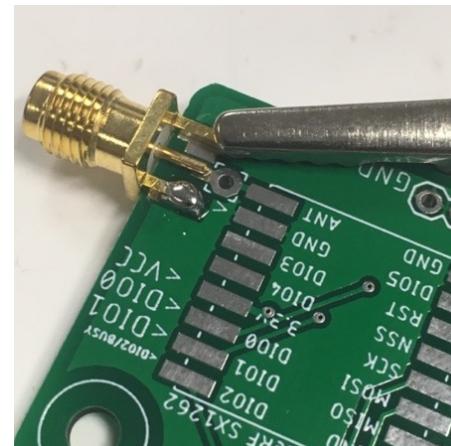
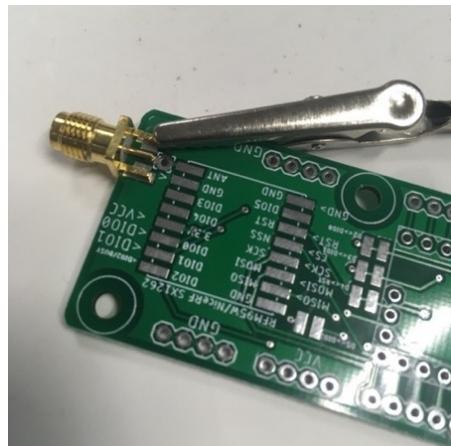
- It is important to have good soldering to avoid any issue when deploying the devices
- Check this great soldering tutorial from Collin's lab
  - <https://youtu.be/RB4P1HTmwLc>



- 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

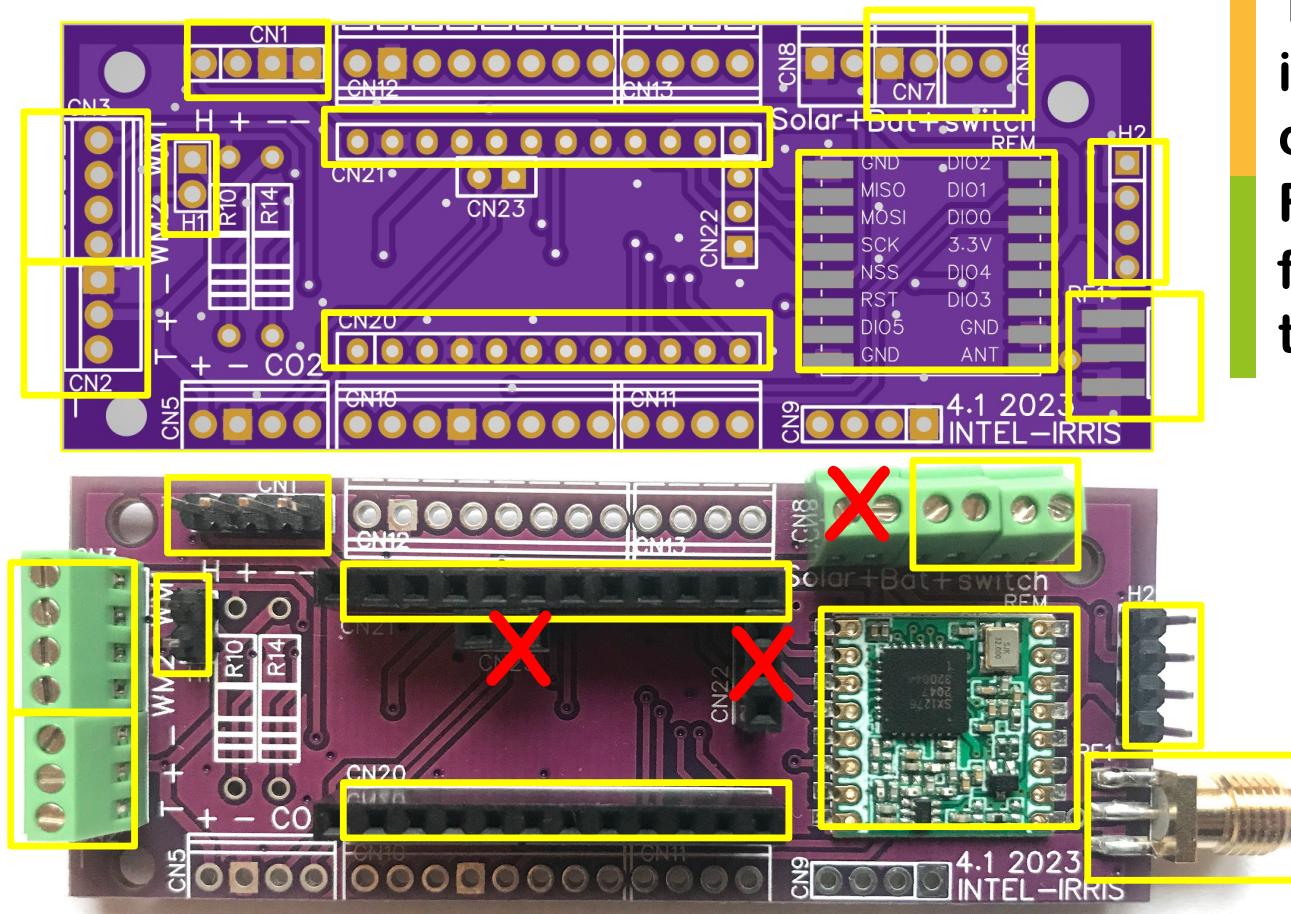
# Updates on Video n°1

- Clip the SMA connector to the PCB to correctly align the connector before soldering one of the GND pad, then remove clip and finish soldering
- **Or**, gently press SMA connector pads so that insertion on PCB is harder and SMA connector will not move during soldering
- Make sure that the GND pads and the central pin are not touching each other, in doubt, do a continuity check



# Wiring with IRD PCB v4.1 (raw version)

- First, solder the various components



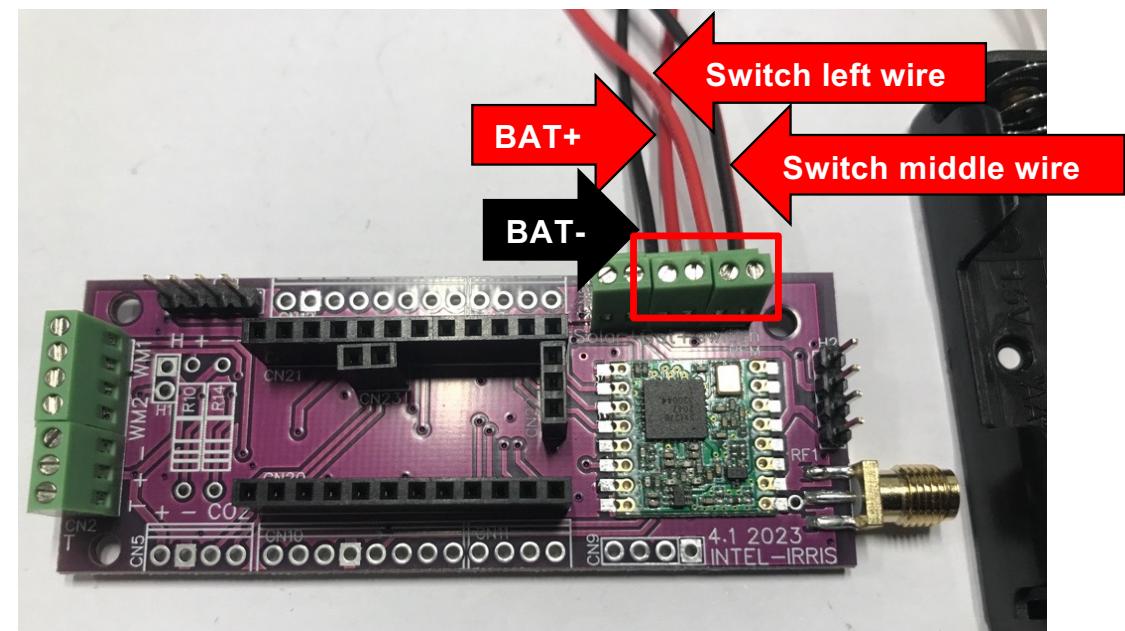
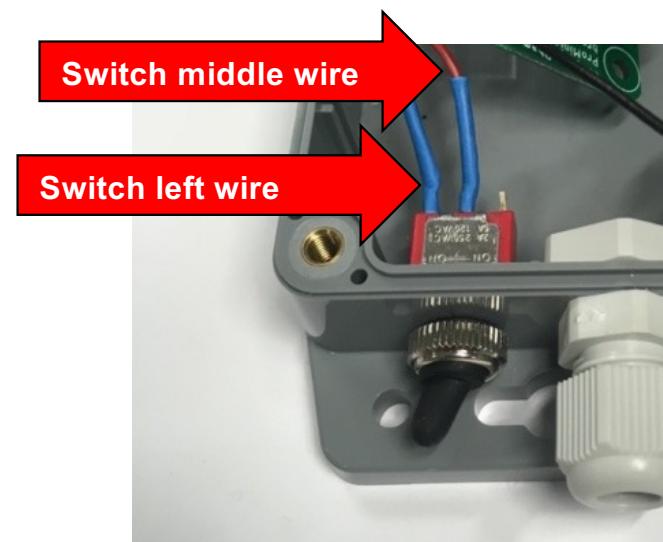
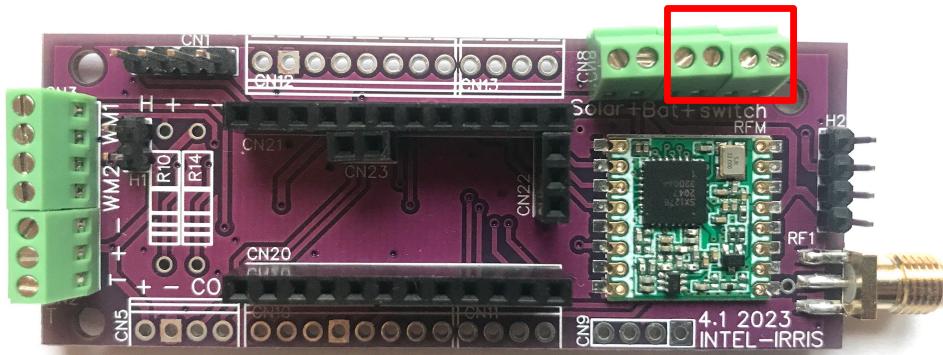
The raw version is not intended to have solar charging capabilities. For solar charging, the full version (see later) is the dedicated PCB

Not needed for the raw PCB which is not designed to have the solar circuit manually soldered

# Wiring with IRD PCB v4.1 (raw version)

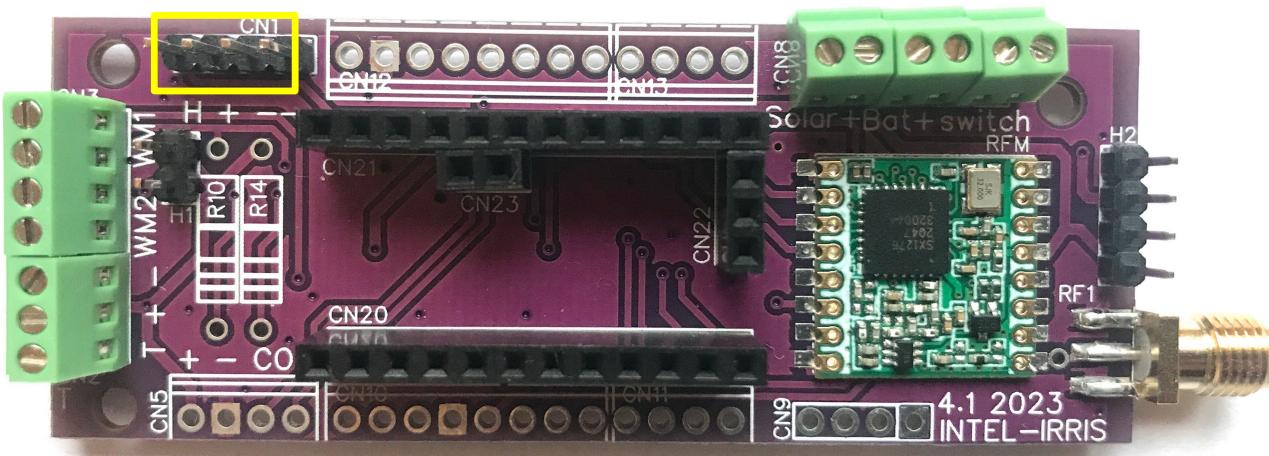
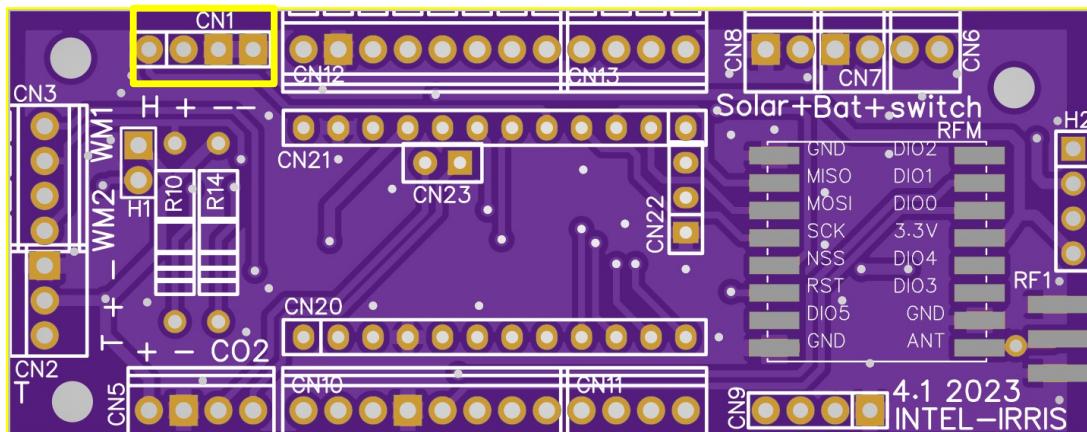


- Power wires (from AA alkaline battery)



# Wiring with IRD PCB v4.1 (raw version)

- SEN0308 capacitive

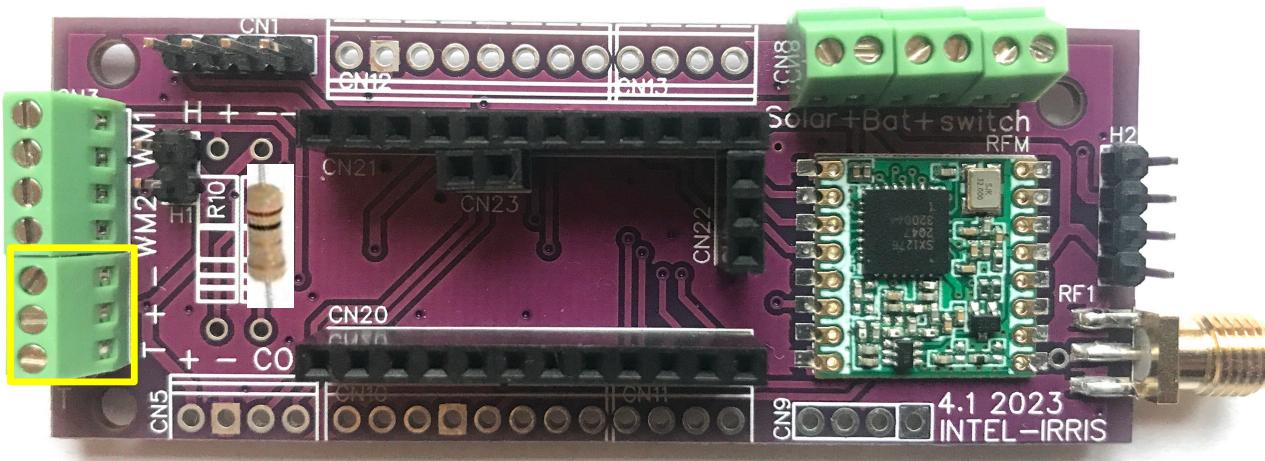
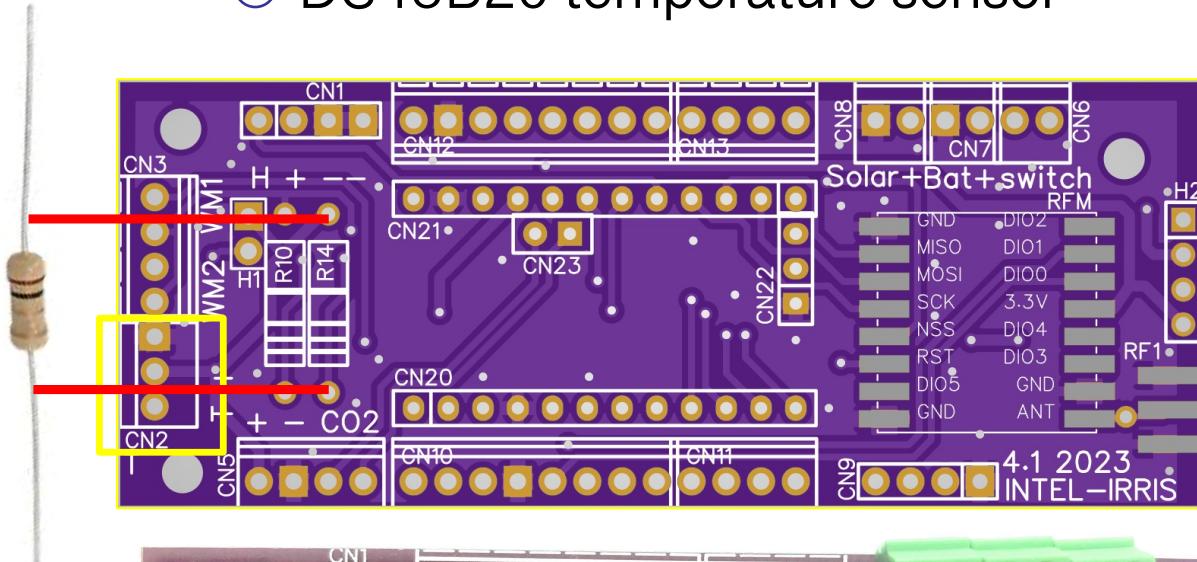


Just connect the sensor in the dedicated header  
 -- are the 2 black wires  
 + is red and H is yellow



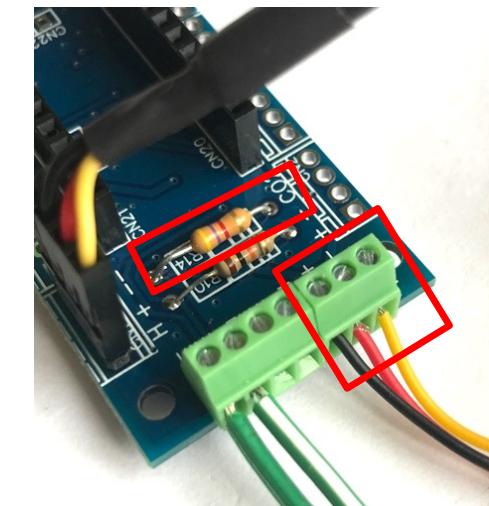
# Wiring with IRD PCB v4.1 (raw version)

- DS18B20 temperature sensor



Solder a 4.7kOhms resistor  
then wire in the dedicated  
terminal block

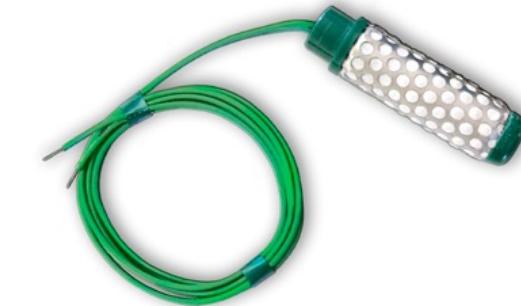
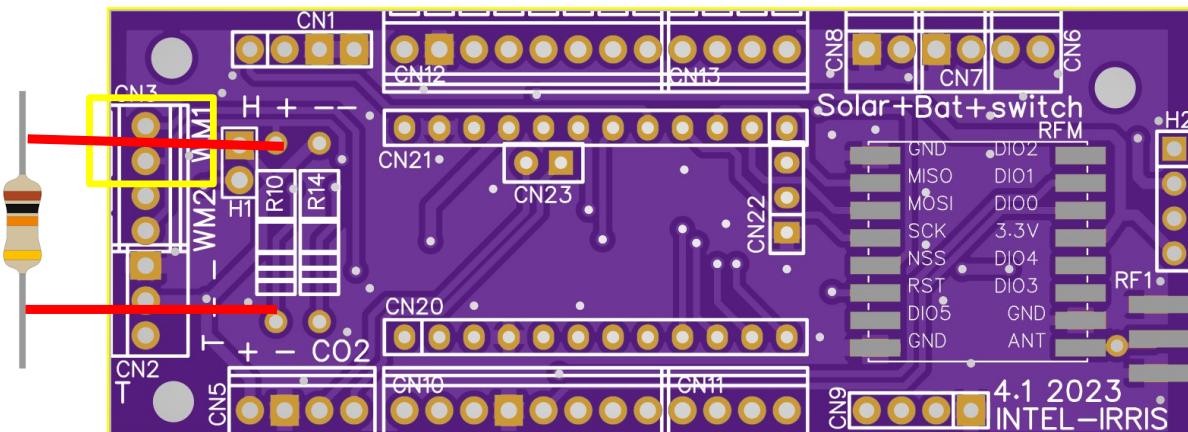
T+ - : Yellow, Red, Black wires



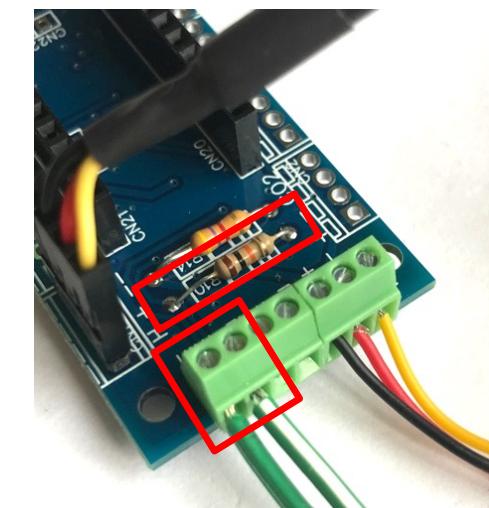
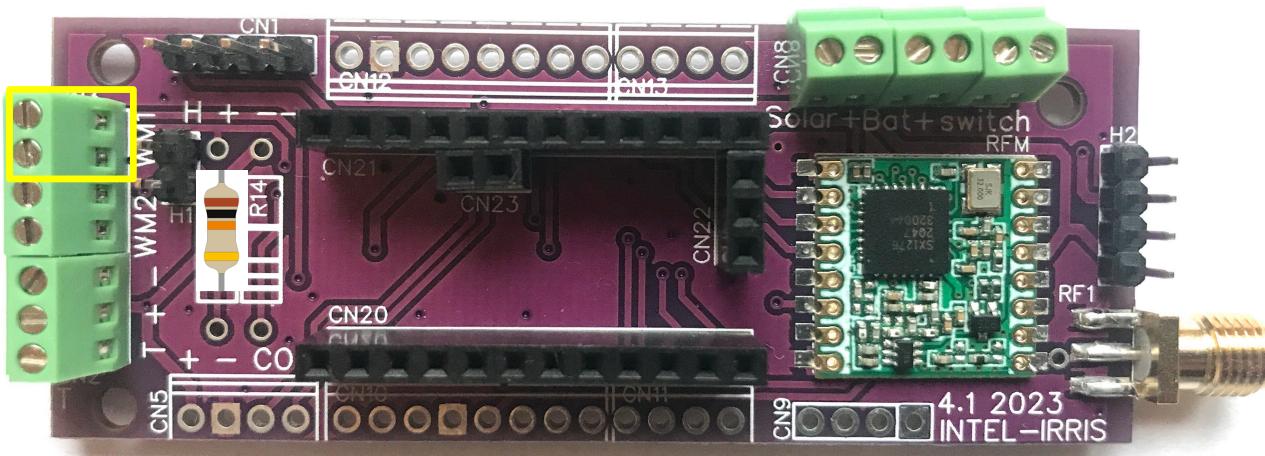
# Wiring with IRD PCB v4.1 (raw version)



## ➊ First Watermark



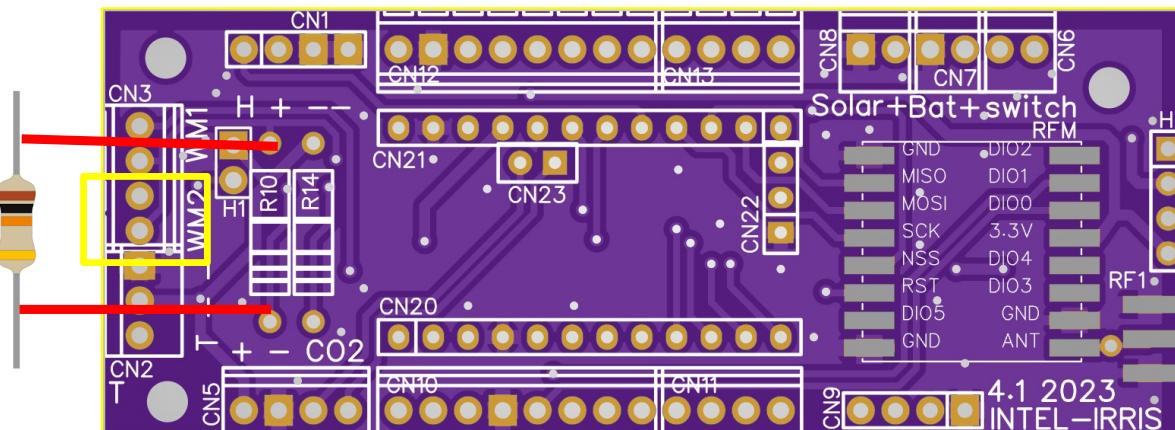
Solder a 10kOhms resistor  
then wire in the dedicated  
WM1 terminal block



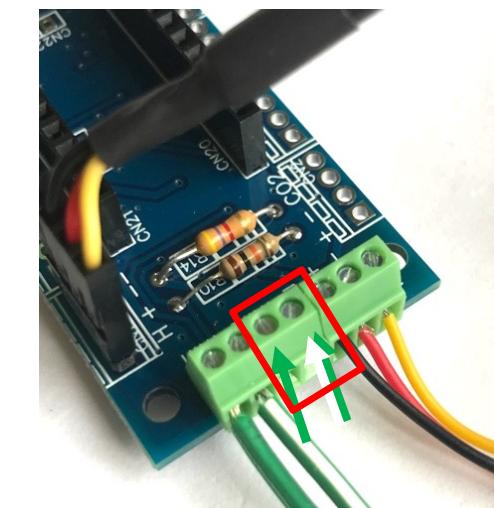
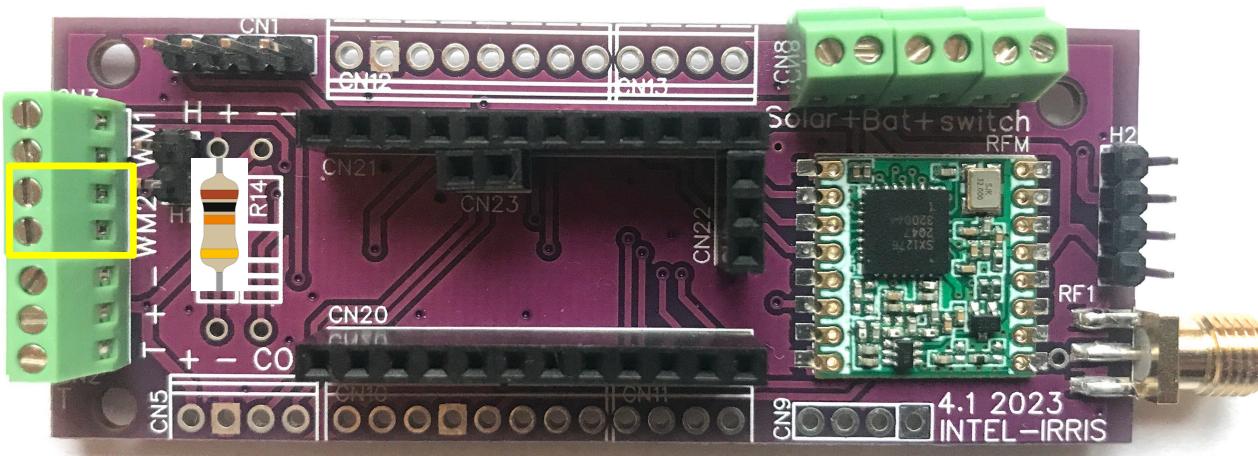
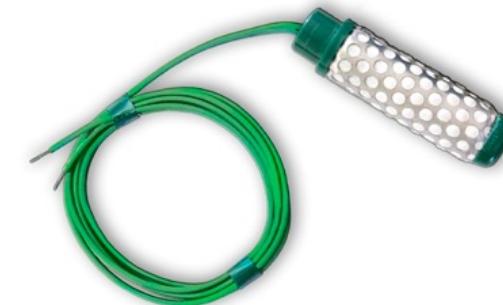
# Wiring with IRD PCB v4.1 (raw version)



- Second Watermark

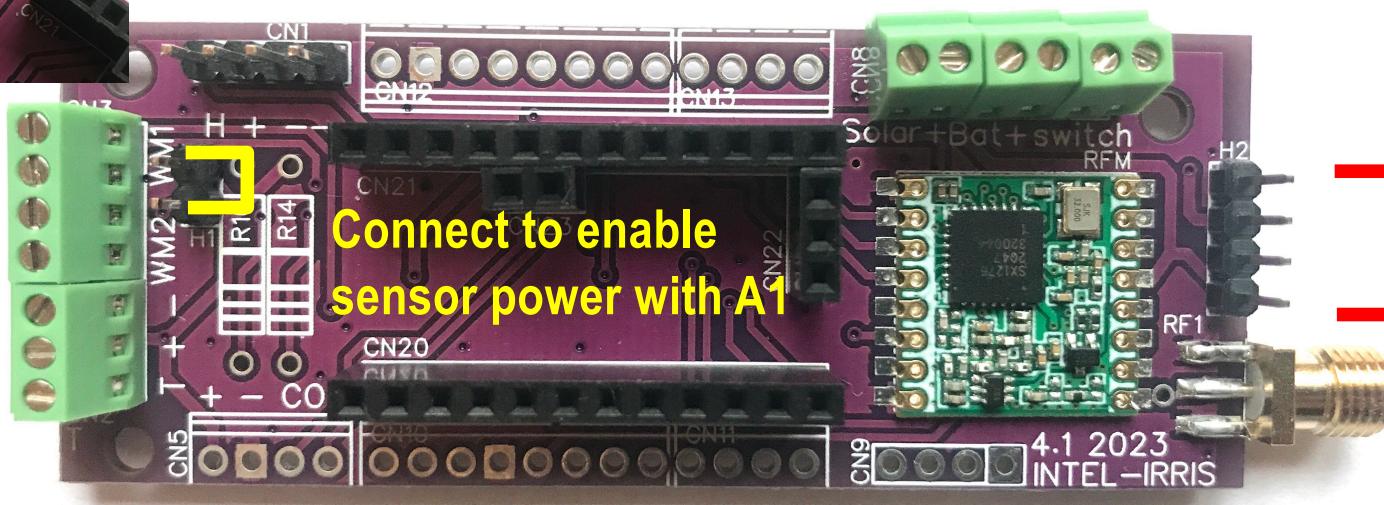
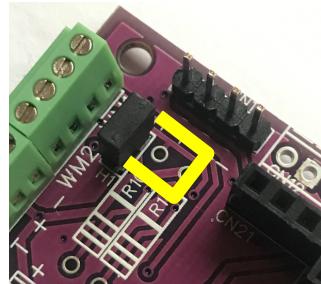


No additional resistor needed  
just wire in the dedicated  
WM2 terminal block



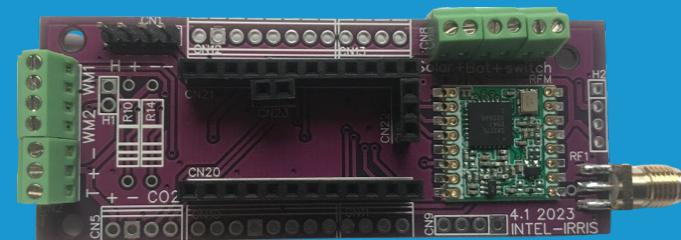
# Wiring with IRD PCB v4.1 (raw version)

- The raw version is not intended to have the solar circuit soldered manually on the back side
  - 2 alkaline AA batteries will be used similar to the simple PCB v2
  - It is necessary to short the 2 external pins of H2 connector and the 2 pins of H1 connector



# INTEL-IRRIS

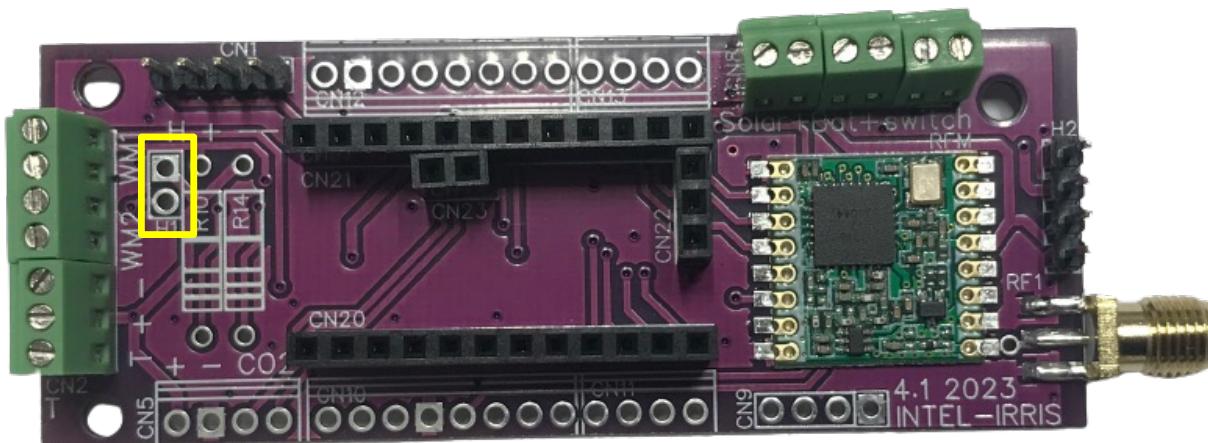
Intelligent Irrigation System for Low-cost Autonomous Water Control  
in Small-scale Agriculture



## LATEST IRD PCBA V4.1

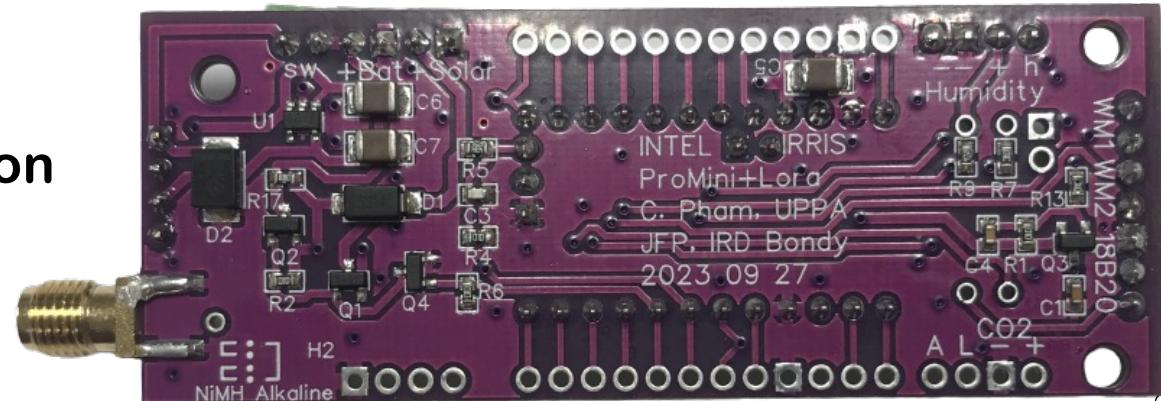
# Wiring with IRD PCB v4.1 (PCBA)

- The PCBA is already fully assembled, including the resistors for the temperature and watermark sensors (on the back side)



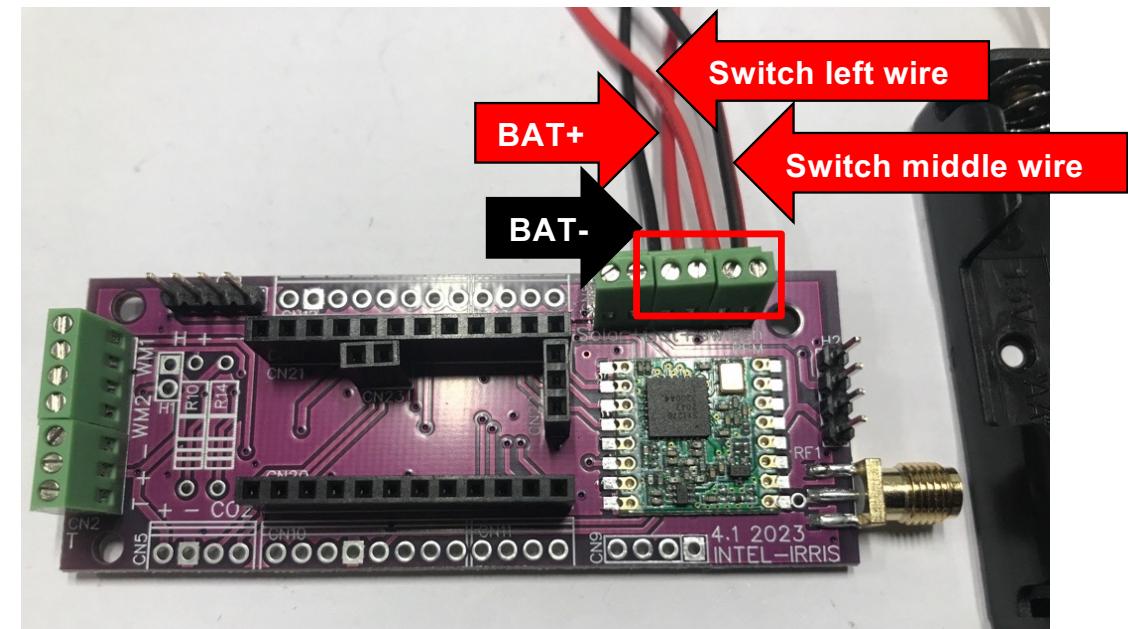
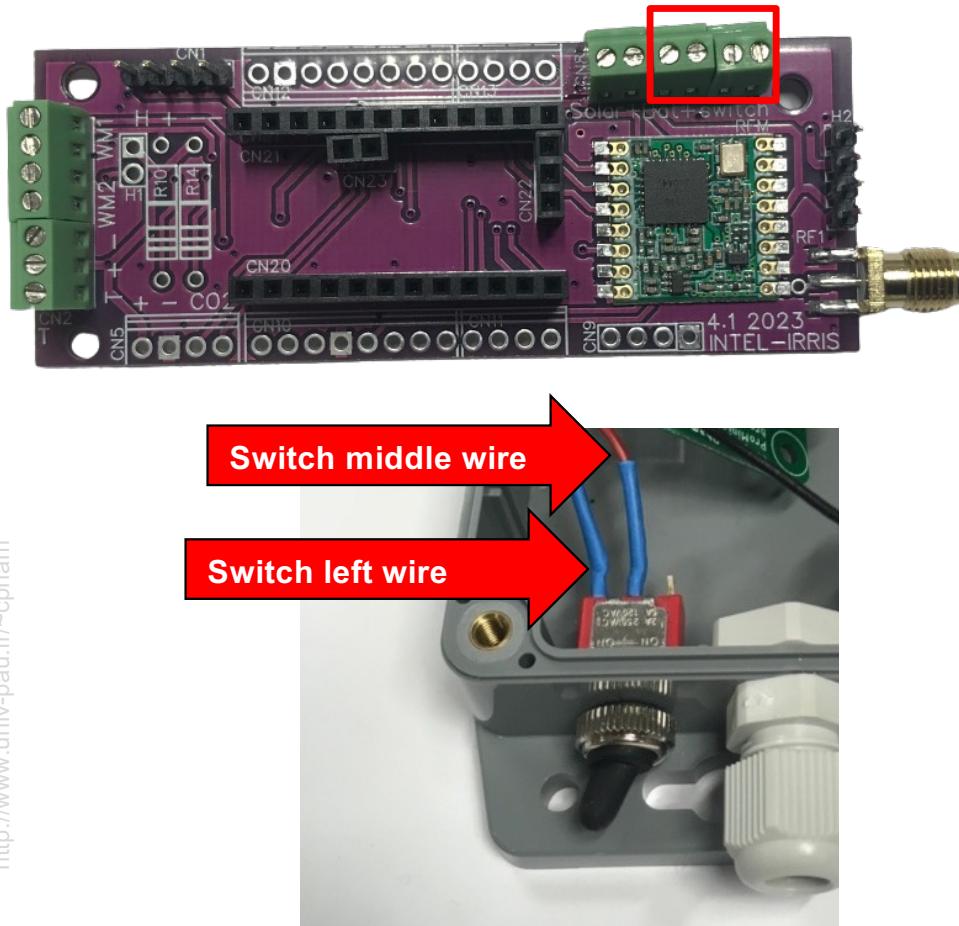
The full version  
does not need to  
have header for H1

# The solar circuit is on the back side



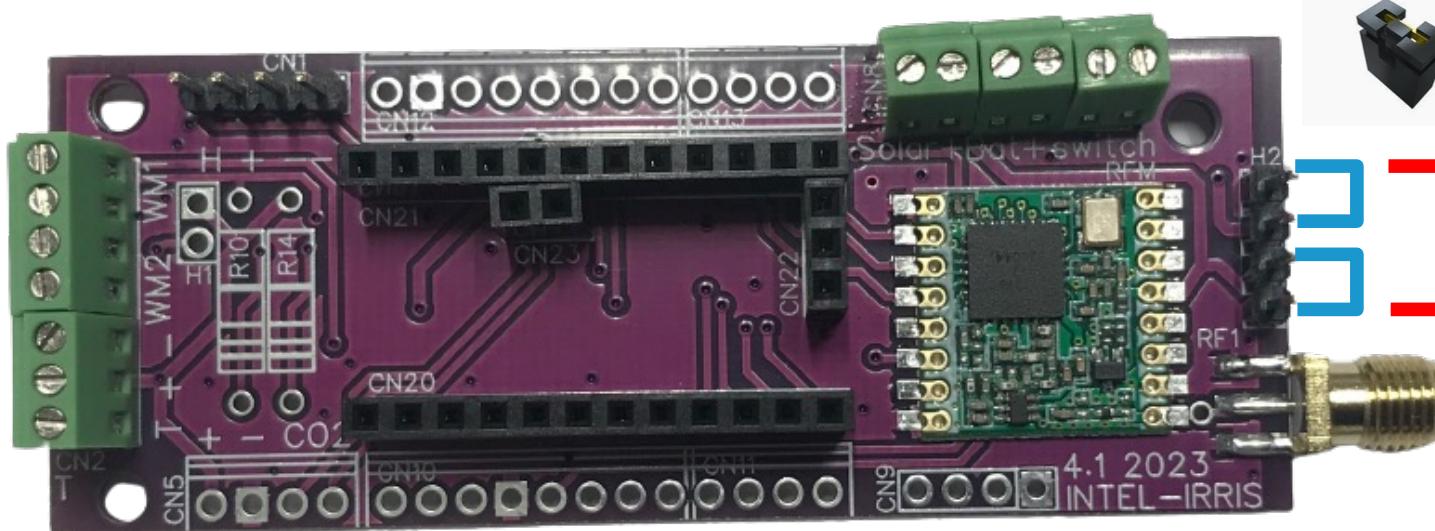
# Wiring with IRD PCB v4.1 (PCBA)

- Power wires (from NiMh batteries or AA alkaline batteries)

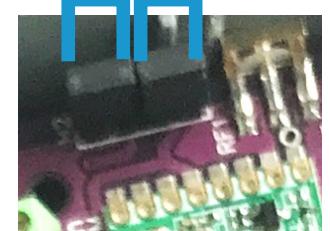


# Wiring with IRD PCB v4.1 (PCBA)

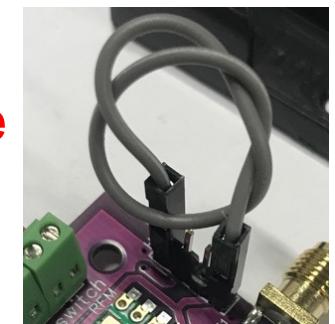
- The full version (PCBA) comes with the solar charging circuit
- **BUT** you can still power it with 2 alkaline AA batteries without using a solar panel to avoid external fragile parts
- You can select between **solar charging with NiMh batteries** or **no solar charging with 2 alkaline AA batteries**



Connect  
for SOLAR  
with NiMh



Connect  
for alkaline  
batteries

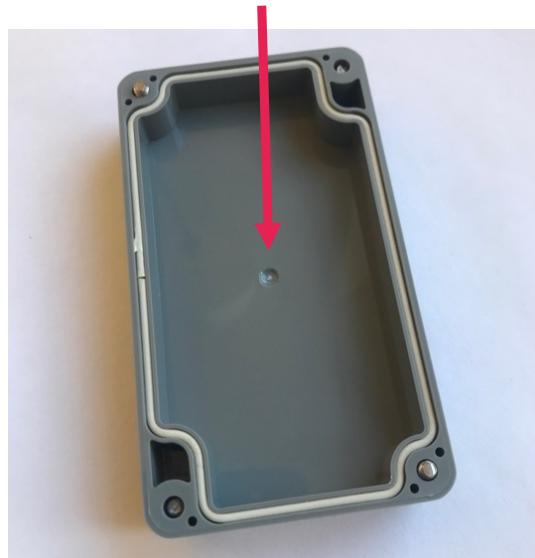


# PCBA: Installing solar panel (1)

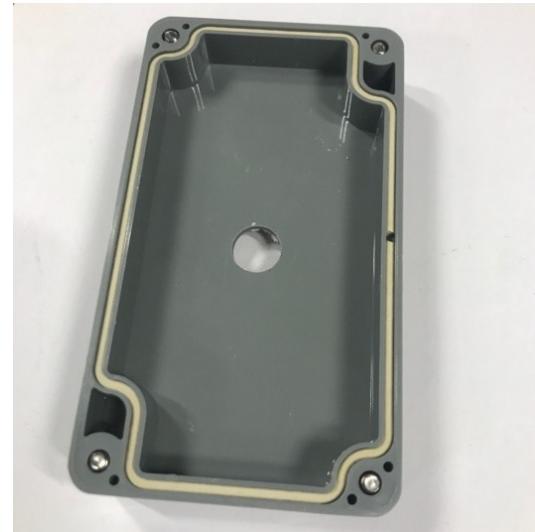
## ○ Mini solar panel

6 V 0.6 W 100 mA 60x90mm

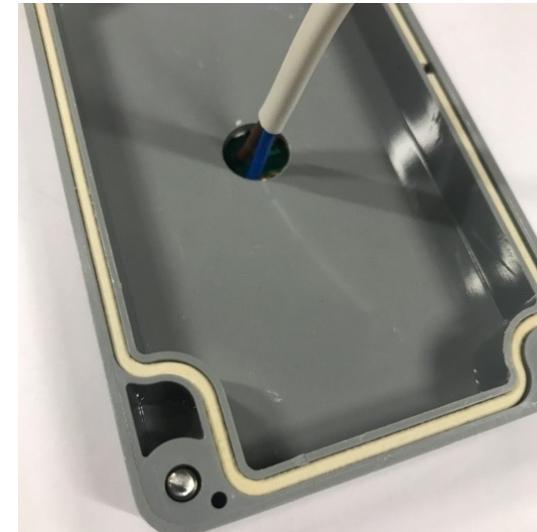
The case cover has  
a mark indicating the  
center



Drill a hole with a  
10mm to 12mm  
drill bit for metal

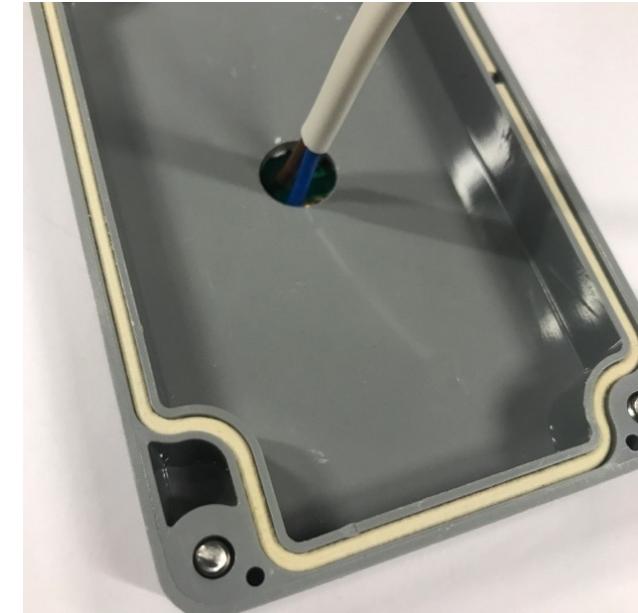
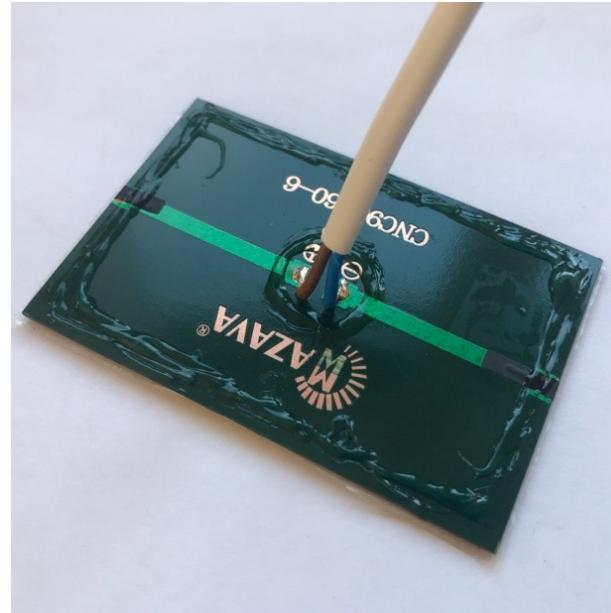
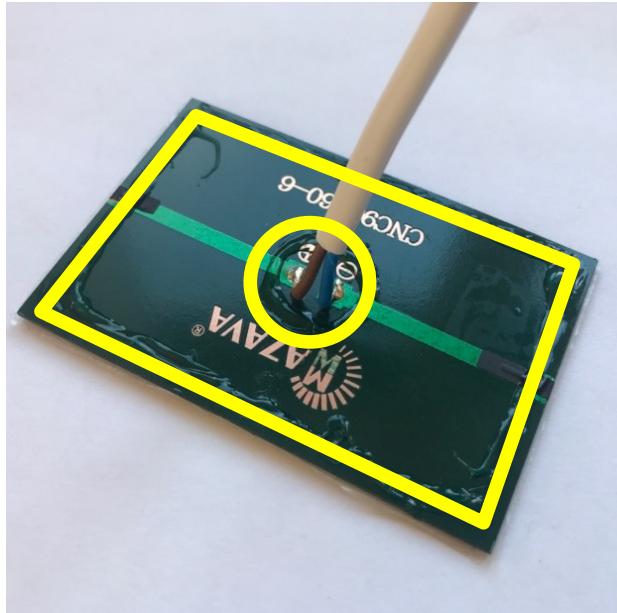


Solder 2 wires on + and -



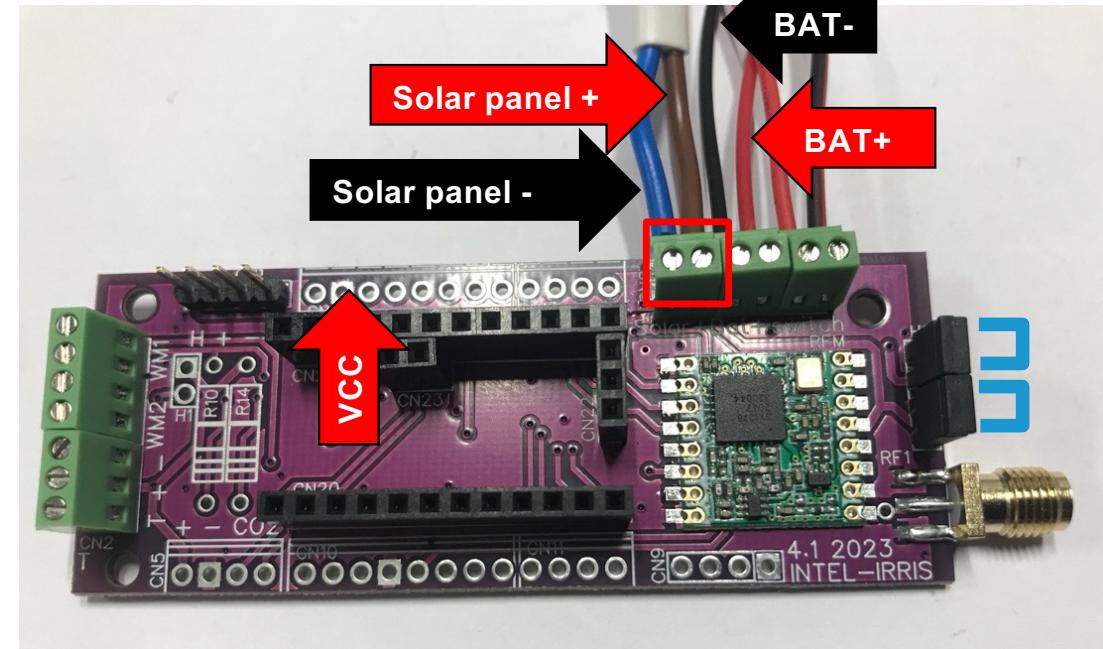
# PCBA: Installing solar panel (2)

- Use silicon or glue to assemble the solar panel on the case cover
- Make sure there is no discontinuity of glue, also add glue around the centered hole surrounding the wires



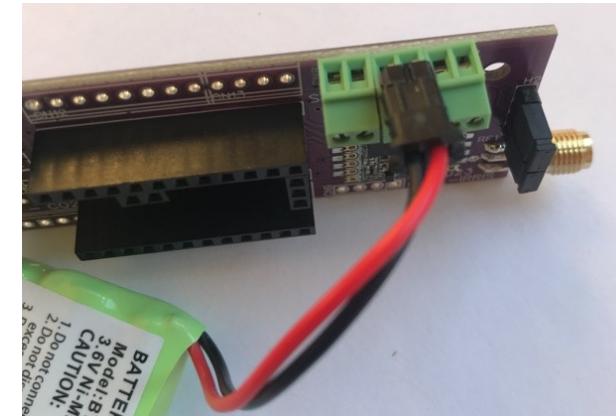
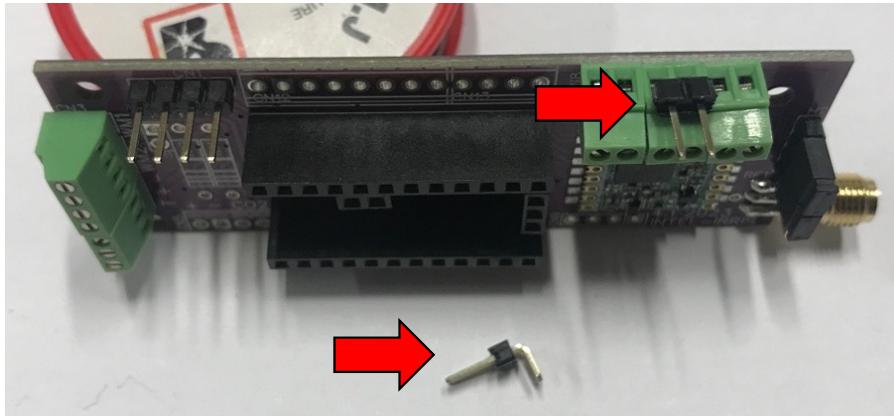
# Wiring with IRD PCB v4.1 (PCBA)

## ➊ Solar panel wires



# Wiring with IRD PCB v4.1 (PCBA)

- Connect NiMh batterie packs with a 2-pin female connector
- You can screw in a 2-pin male 90° header

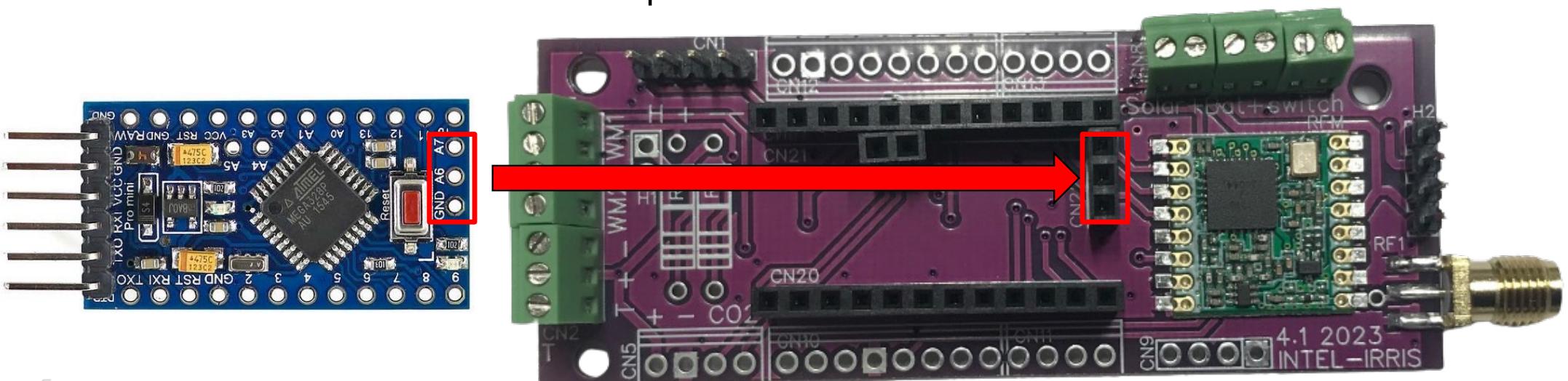


You can buy NiMh  
battery packs with  
a 2 pin connector  
that can be used  
for fast connection

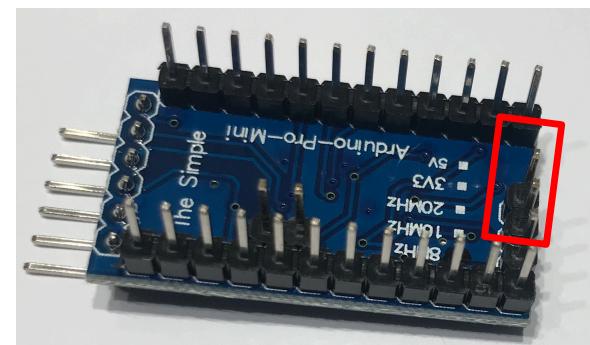


# IMPORTANT: Arduino ProMini

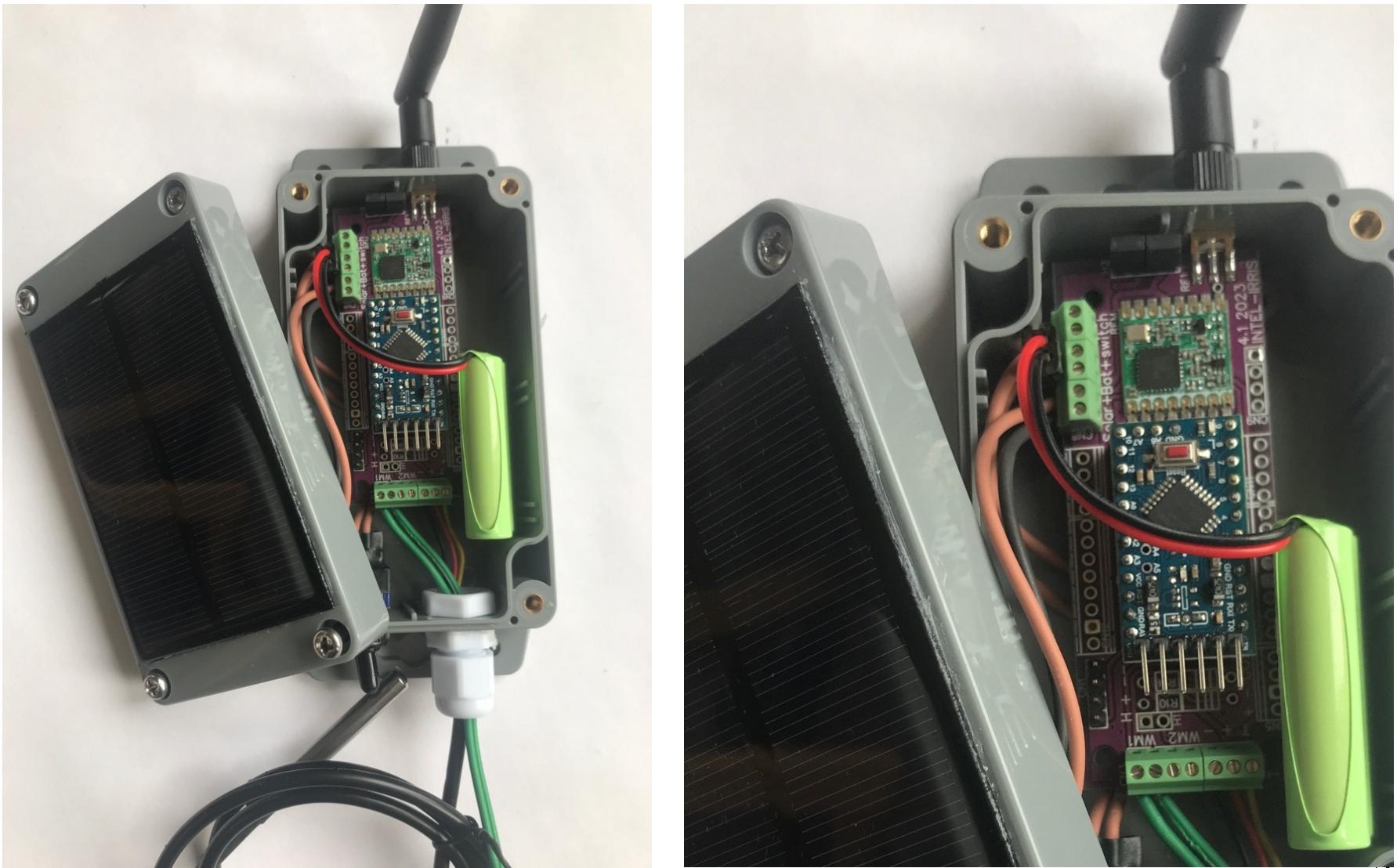
- For solar charging, using the IRD PCBA v4.1, the Arduino ProMini **MUST** have pin A7 connected!



- So, it is **MANDATORY** to solder a 3-pin header for GND, A6 & A7 on the Arduino ProMini board

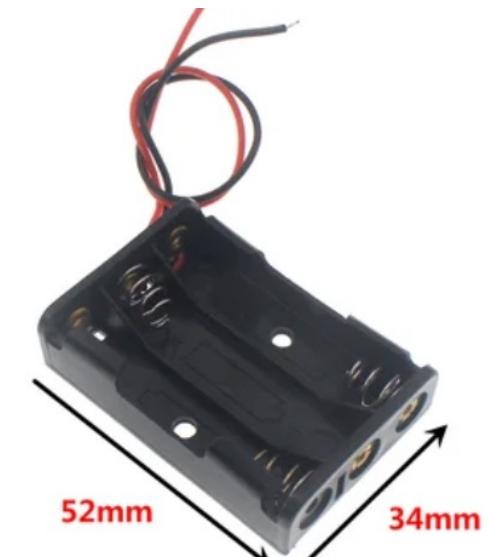
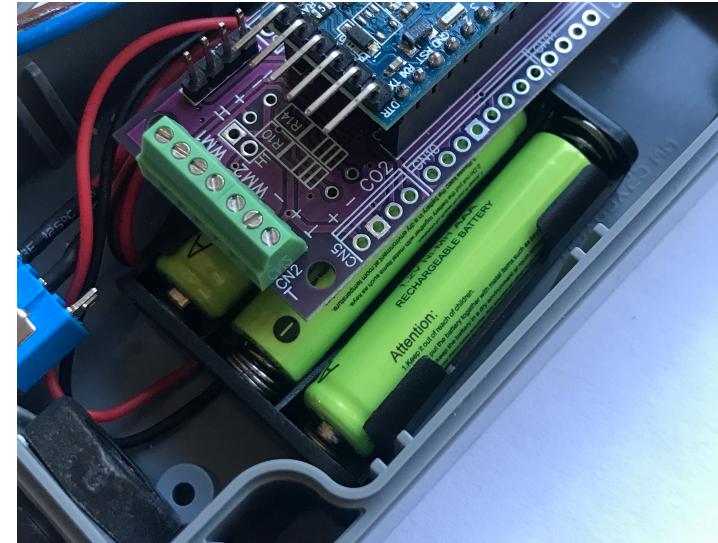
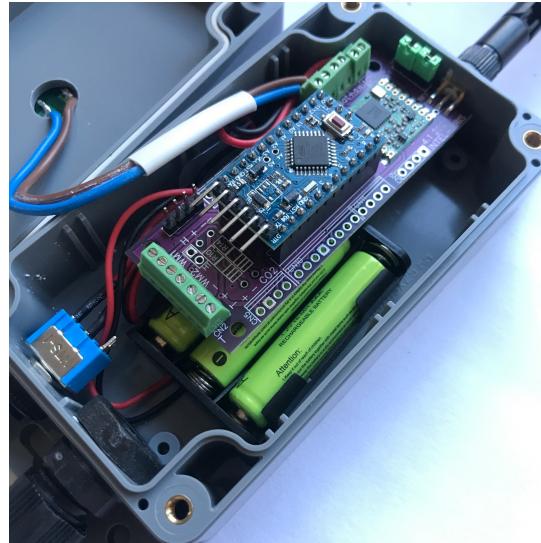


# Soil device with PCBA v4.1, final result



# Using 3xAAA batteries

- You can alternatively use 3xAAA NiMh rechargeable batteries with a battery holder
- If this is the case, you need to adjust the antenna hole on the external casing described in slide 36
- This would allow you to slip the 3xAAA battery holder under the PCBA board



A close-up photograph of a young green plant with several leaves. A thin, clear tube, likely made of PVC or Teflon, is attached to the stem near the base of a leaf, presumably for irrigation. The background is blurred, showing more plants and possibly a garden setting.

# INTEL-IRRIS

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

# INTEGRATION & CASING

# Get an enclosure for outdoor usage



Here, it is an IP65 box which dimension is 115 x 65 x 40mm

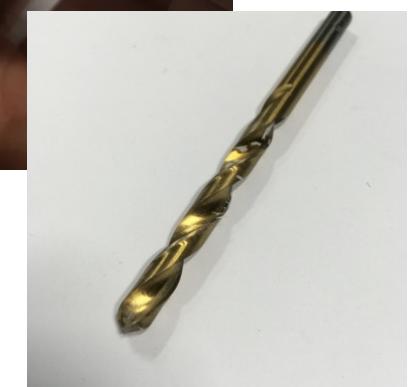
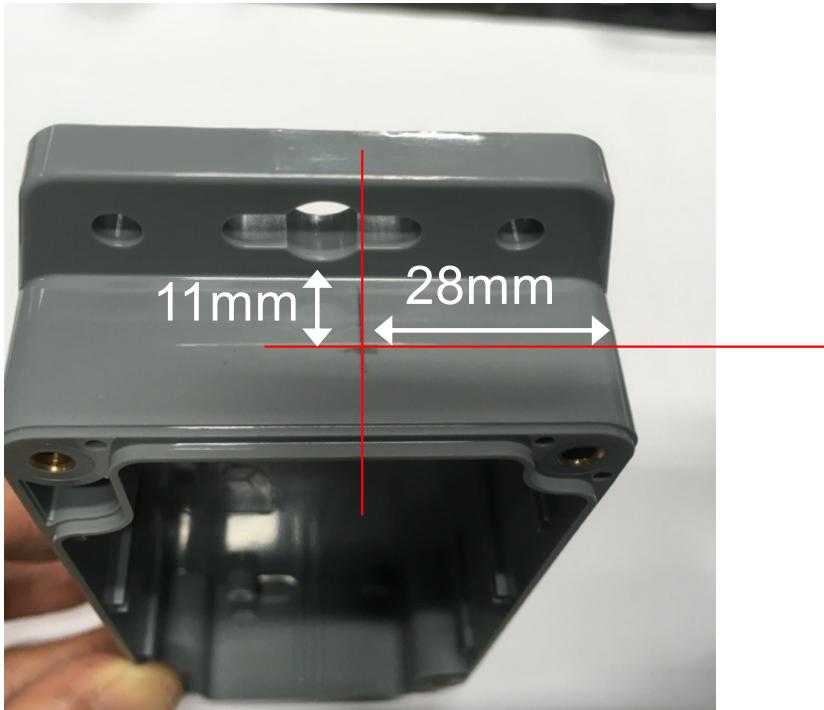
<https://www.gotronic.fr/art-boitier-abs-etanche-g304m-17977.htm>

# 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

11mm from the outside bottom

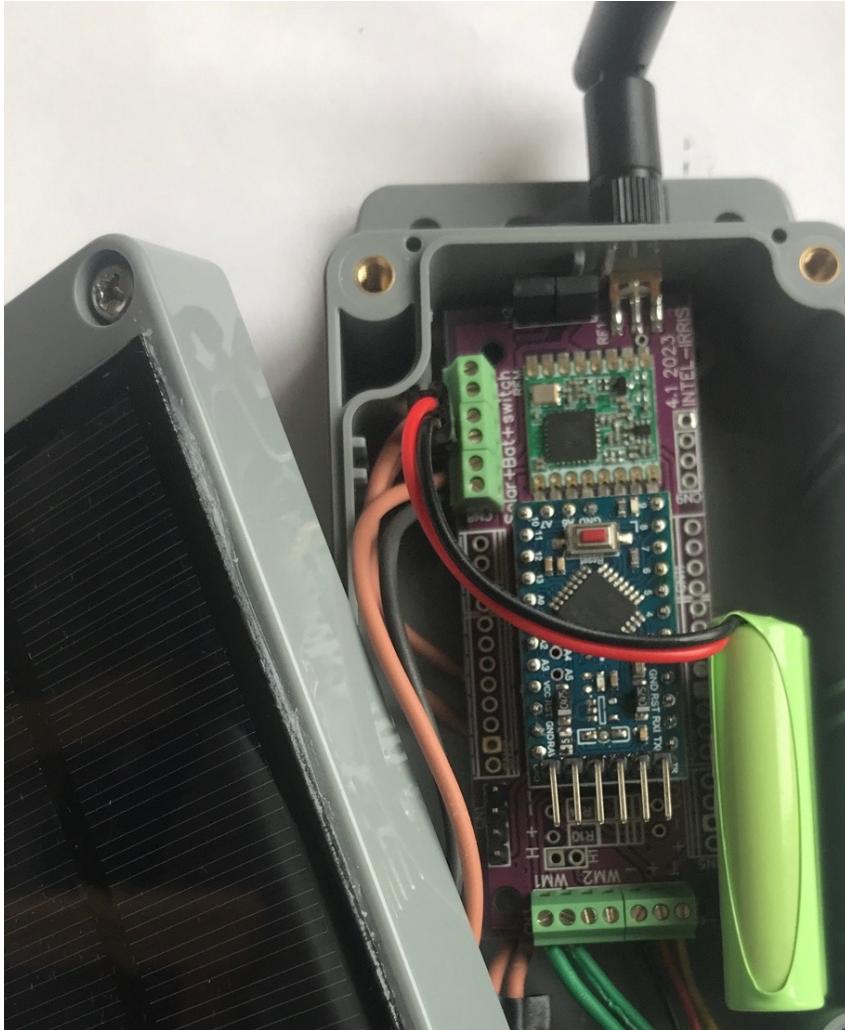
**use a 7mm drill bit for metal, not for wood nor concrete!**

# Remove unwanted plastic part



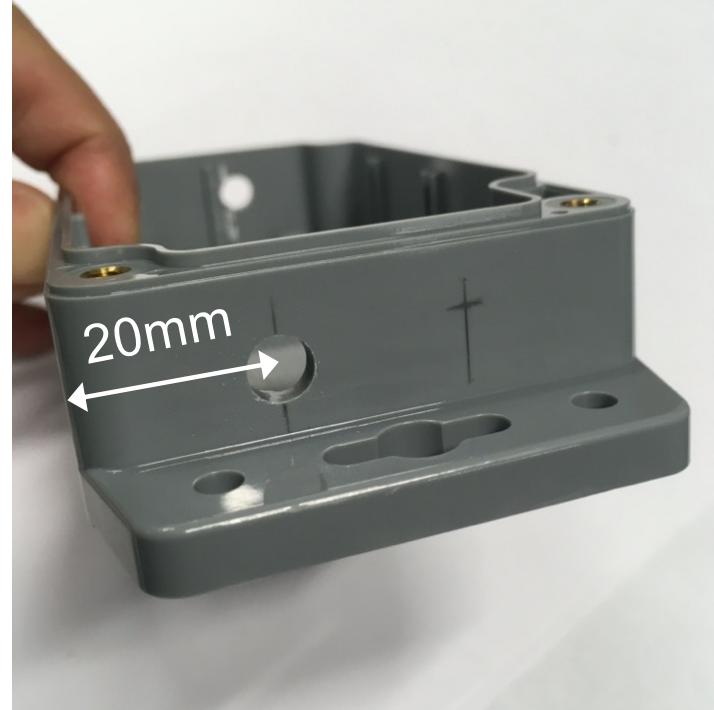
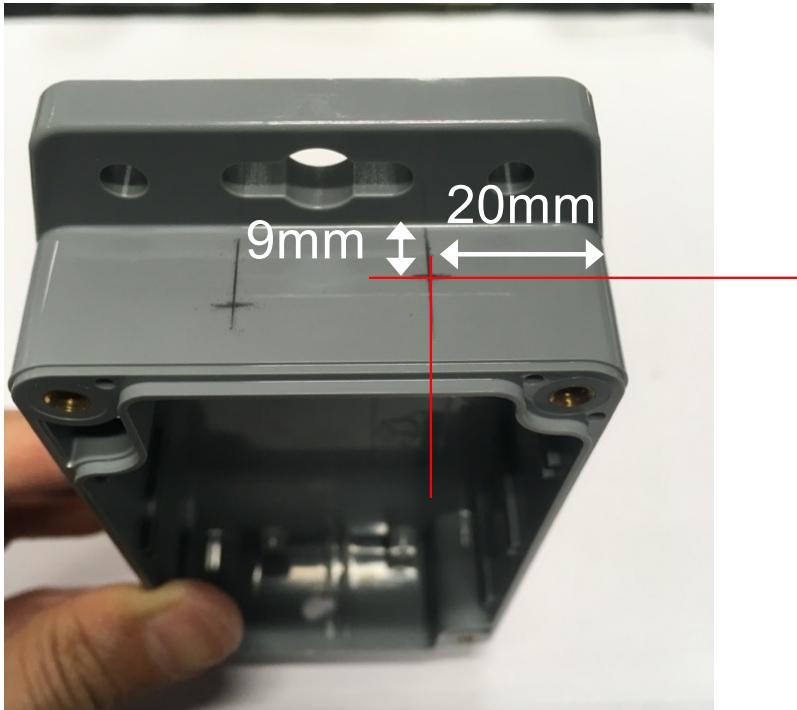
we need to remove the plastic reinforcement part for this particular enclosure  
use a flat cutter for instance to remove and smooth the inside part  
(a small plier can be used first to remove most of the part)

# Placing the PCB board



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

# 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



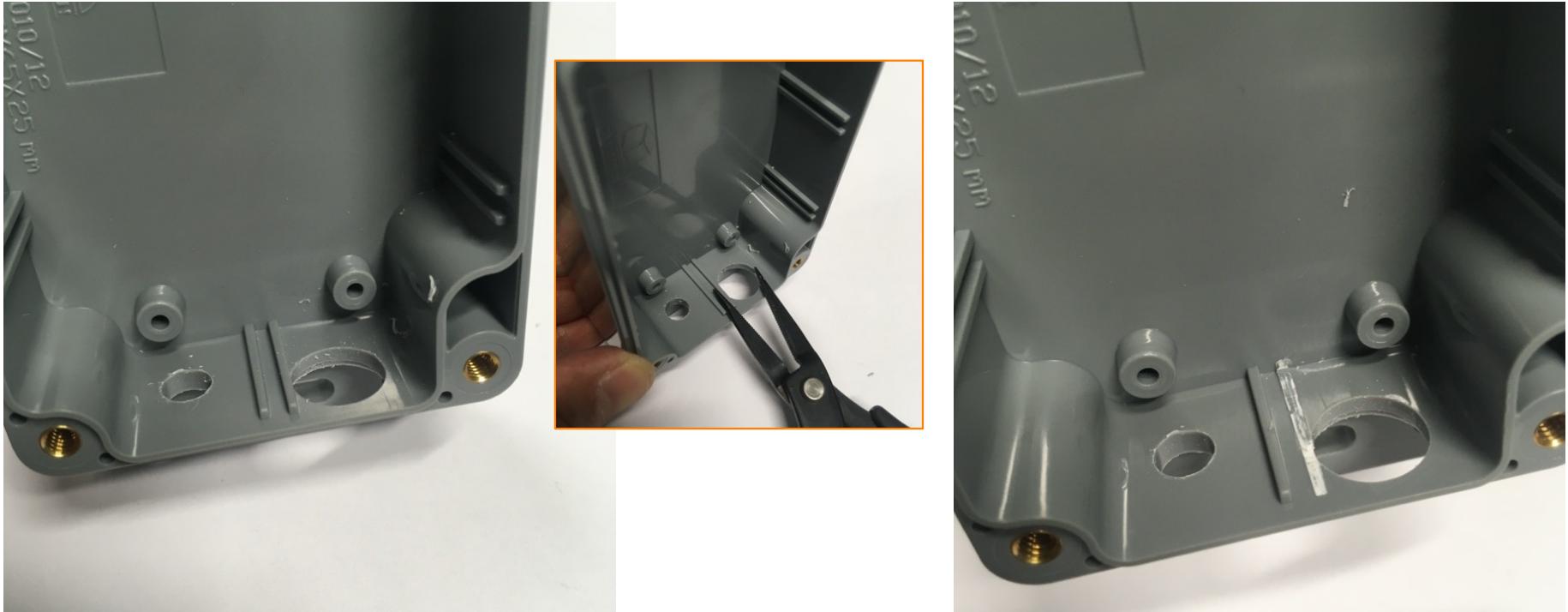
41mm for the right edge:

- measure from the flat side as the corner is round

9mm from the outside bottom

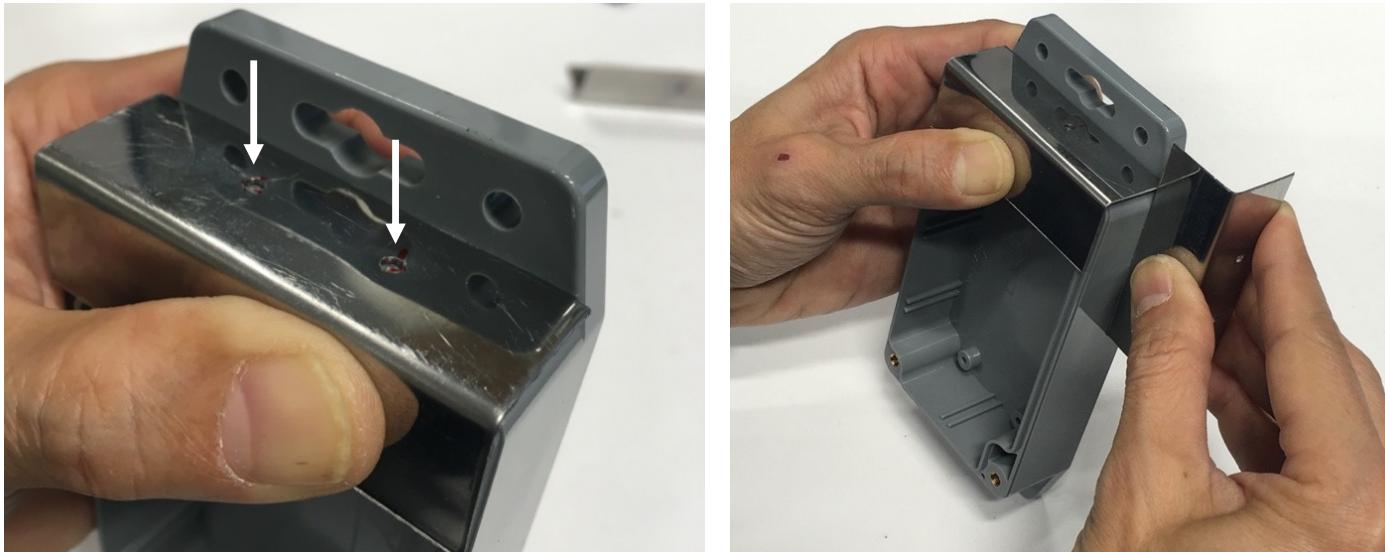
use a 13mm drill bit for metal, but it is recommended to use a step drill bit to first get a 12mm hole before using the 13mm drill bit

# 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)

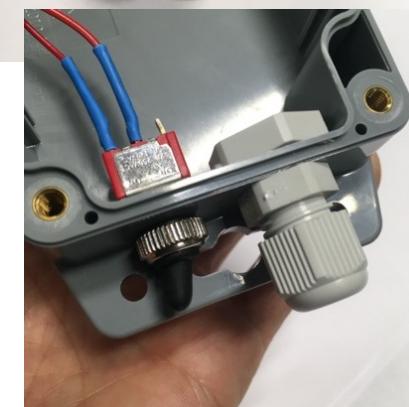
# Going for larger production



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

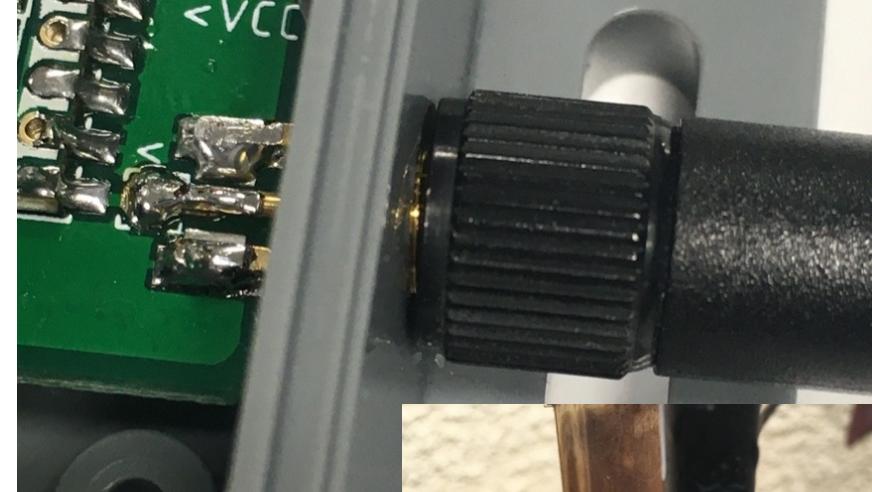
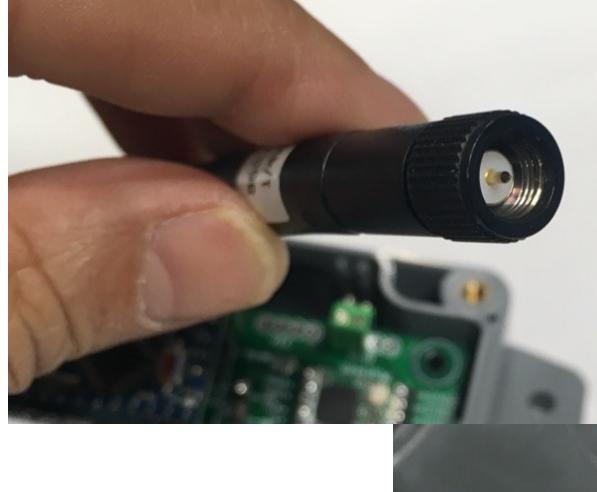
# Placing switch and cable gland



test that everything is OK

the switch has a water-proof rubber cap which should be carefully tighten

# Connecting antenna

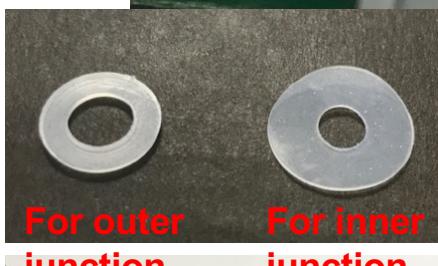


Be sure to connect the matching antenna  
Here, SMA female with SMA male antenna  
Need to screw the antenna in all the way

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

# Waterproofing the antenna junction

Check the gap size



See example in  
the list of part



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

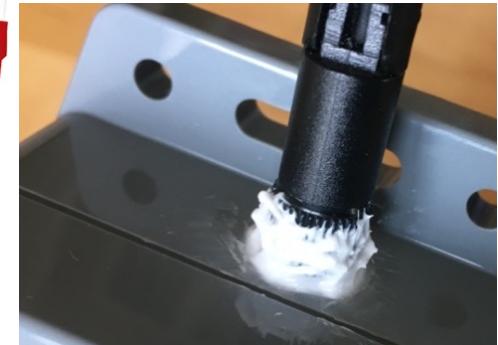
Even with no apparent gap, it is  
necessary to waterproof the  
junction

Take flat silicon seals 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!

# Do not have or can not use flat seal?

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

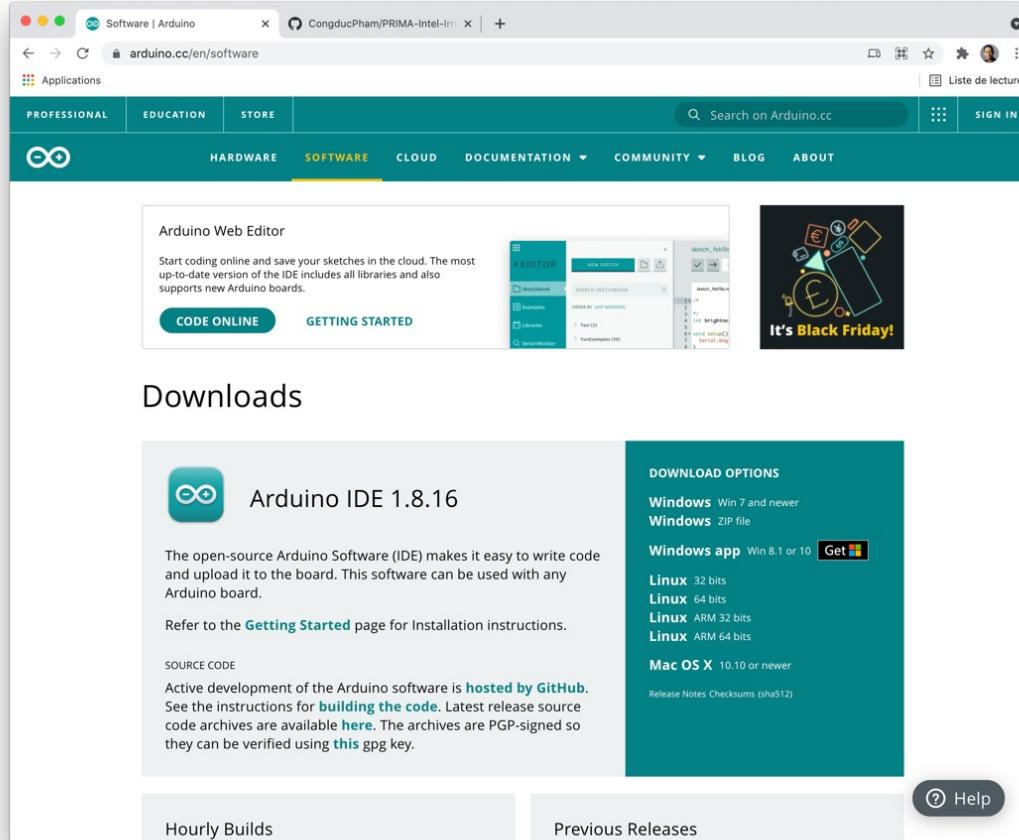
A close-up photograph of a young green plant with several leaves. In the background, there are more plants and some irrigation pipes or hoses. The overall color palette is earthy tones like browns, greens, and tans.

# INTEL-IRRIS

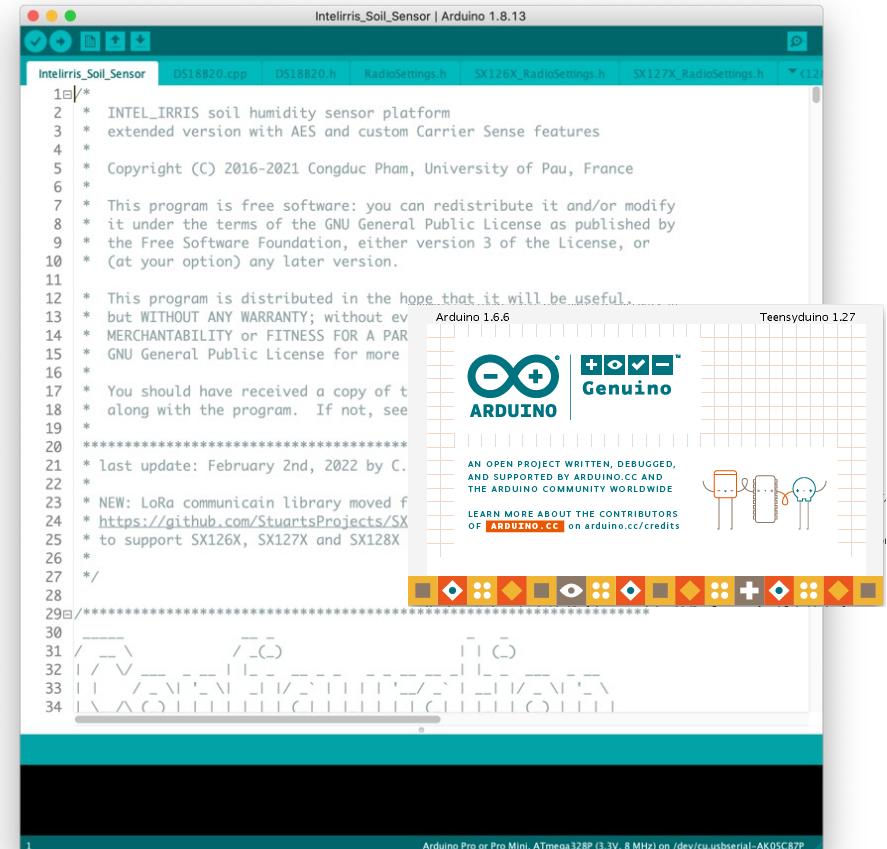
Intelligent Irrigation System for Low-cost Autonomous Water Control  
in Small-scale Agriculture

# PROGRAM THE ARDUINO

# Getting the software: Arduino IDE



The screenshot shows the Arduino Software (IDE) download page. It features the Arduino Web Editor interface at the top, followed by a "Downloads" section. In the "Downloads" section, there is a large button for "Arduino IDE 1.8.16". Below this button, there is a brief description of what the IDE is used for and a link to the "Getting Started" page. Further down, there is a "SOURCE CODE" section with instructions on how to verify the code using a GPG key. At the bottom of the page, there are links for "Hourly Builds" and "Previous Releases".



The screenshot shows the Arduino IDE 1.8.16 interface with an open sketch titled "Inteliris\_Soil\_Sensor". The code editor displays the following sketch:

```

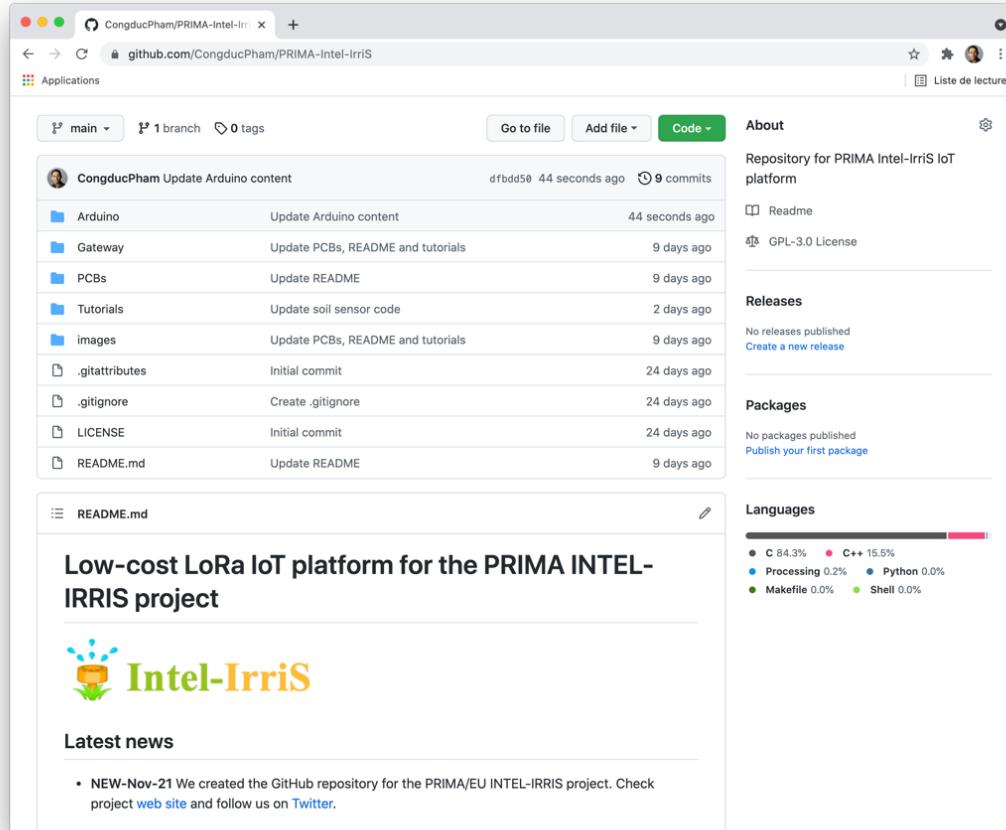
1 /*
2 * INTEL_IRRIS soil humidity sensor platform
3 * extended version with AES and custom Carrier Sense features
4 *
5 * Copyright (C) 2016-2021 Congduc Pham, University of Pau, France
6 *
7 * This program is free software: you can redistribute it and/or modify
8 * it under the terms of the GNU General 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 "PROFESSIONAL", "EDUCATION", "STORE", "HARDWARE", "SOFTWARE", "CLOUD", "DOCUMENTATION", "COMMUNITY", "BLOG", and "ABOUT". The right side of the interface shows the Arduino logo and other branding.

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

# Getting the INTEL-IRRIS code



The screenshot shows the GitHub repository page for [CongducPham/PRIMA-Intel-Irris](https://github.com/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 "Low-cost LoRa IoT platform for the PRIMA INTEL-IRRIS project". It includes sections for About, Releases, Packages, and Languages. The Languages section shows C (84.3%), C++ (15.5%), Processing (0.2%), Python (0.0%), Makefile (0.0%), and Shell (0.0%). A sidebar on the left shows latest news, including a note about the creation of the GitHub repository.

On your computer, create a sketch folder

Then download the whole repository as ZIP file

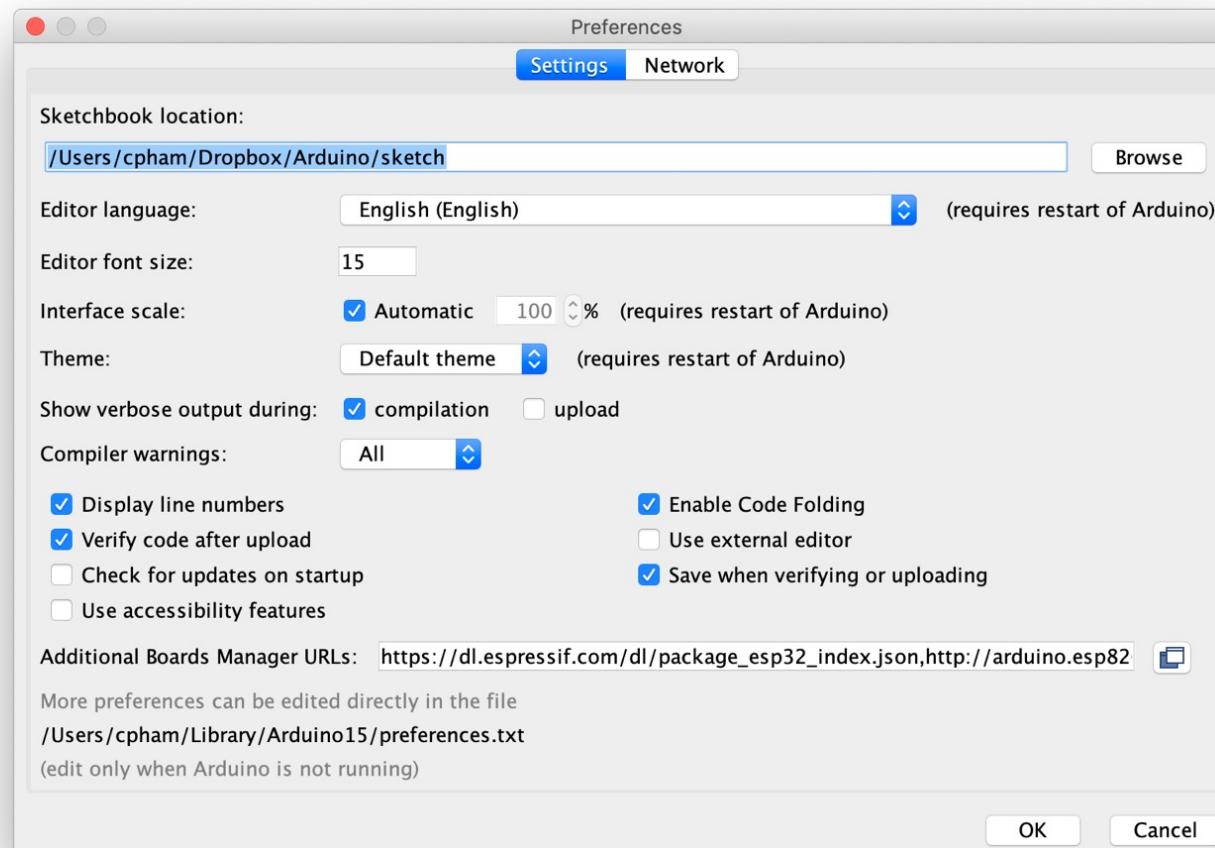
Unzip the file and copy the content of Arduino folder into your sketch folder

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

# Config for IRD PCB v4.1

- Uncomment `IRD_PCB` in `BoardSettings.h`

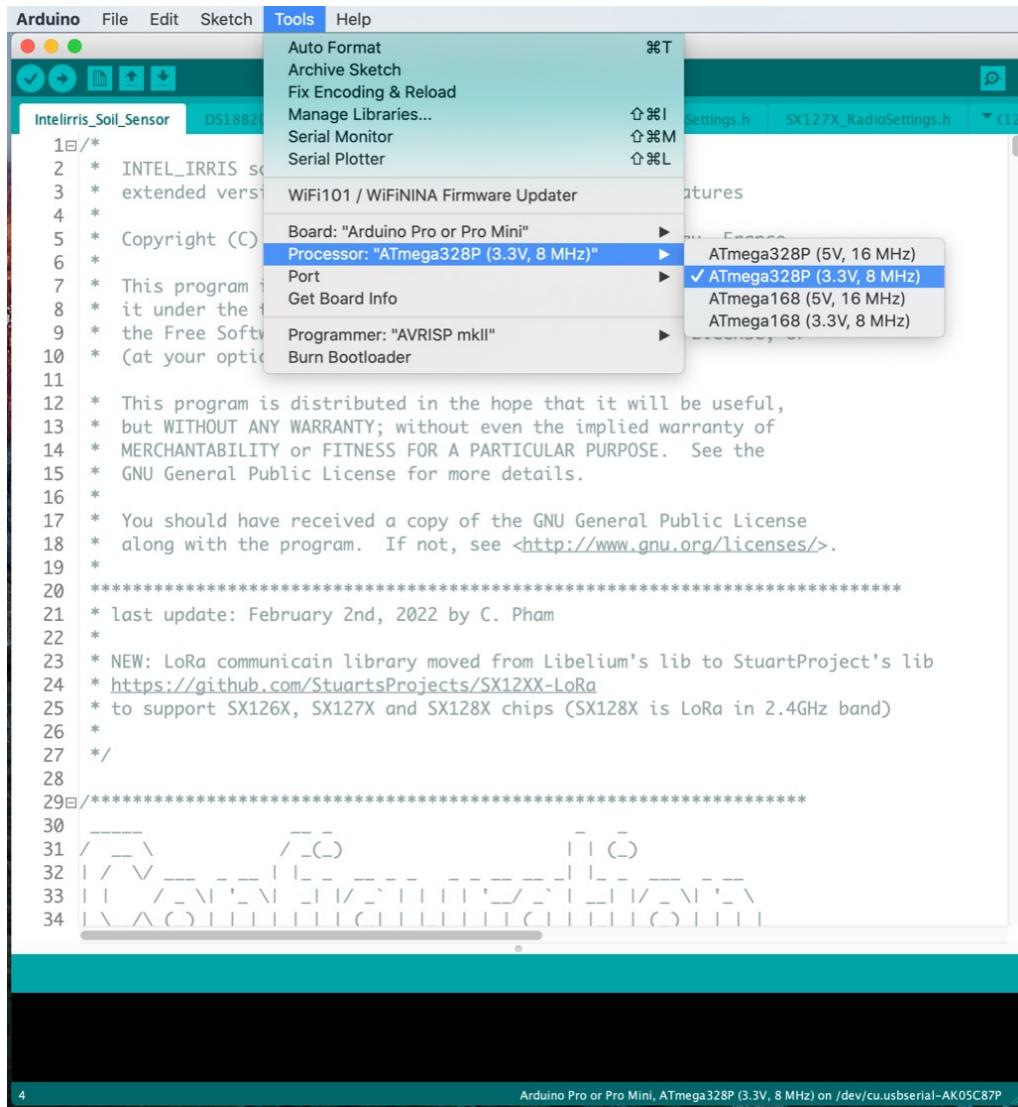
```
//uncomment for WAZISENSE v2 board
//#define WAZISENSE

//uncomment for IRD PCB board
#define IRD_PCB
```

- ONLY for solar version, uncomment `SOLAR_BAT` in `BoardSettings.h`

```
///////////////////////////////
// uncomment only if the IRD PCB is running on solar panel
// MUST be commented if running on alkaline battery
// code for SOLAR_BAT has been written by Jean-François Printanier from IRD
#define SOLAR_BAT
// do not change if you are not knowing what you are doing
#define NIMH
```

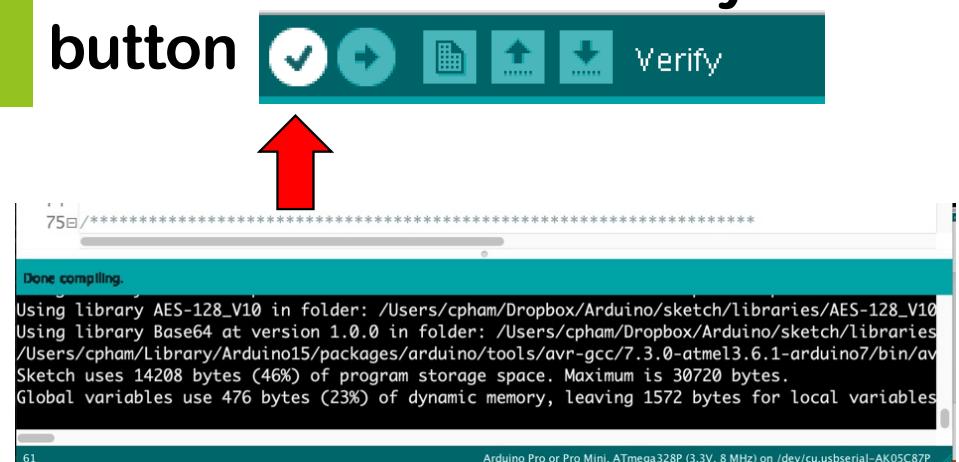
# Compiling the soil sensor code



**Open Intelirris\_Soil\_Sensor sketch – no change required for default capacitive sensor**

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

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



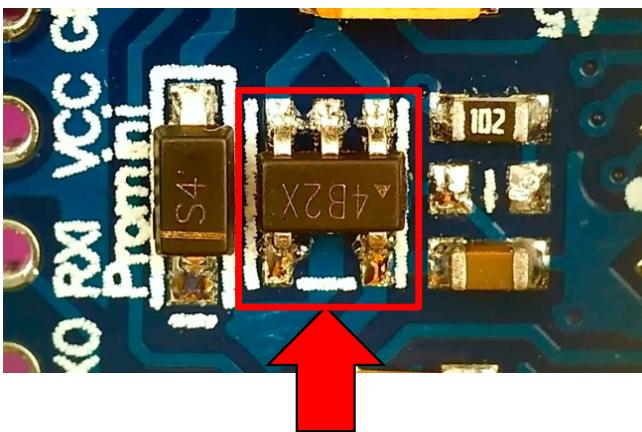


# Never transmit without antenna

- 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, then 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

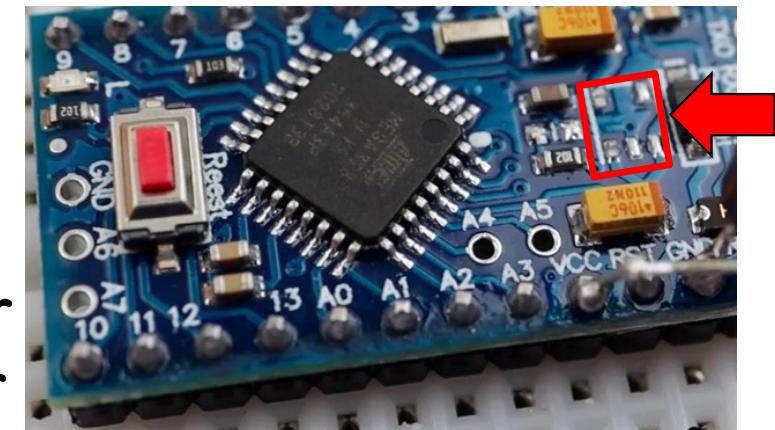


# Reduce power consumption



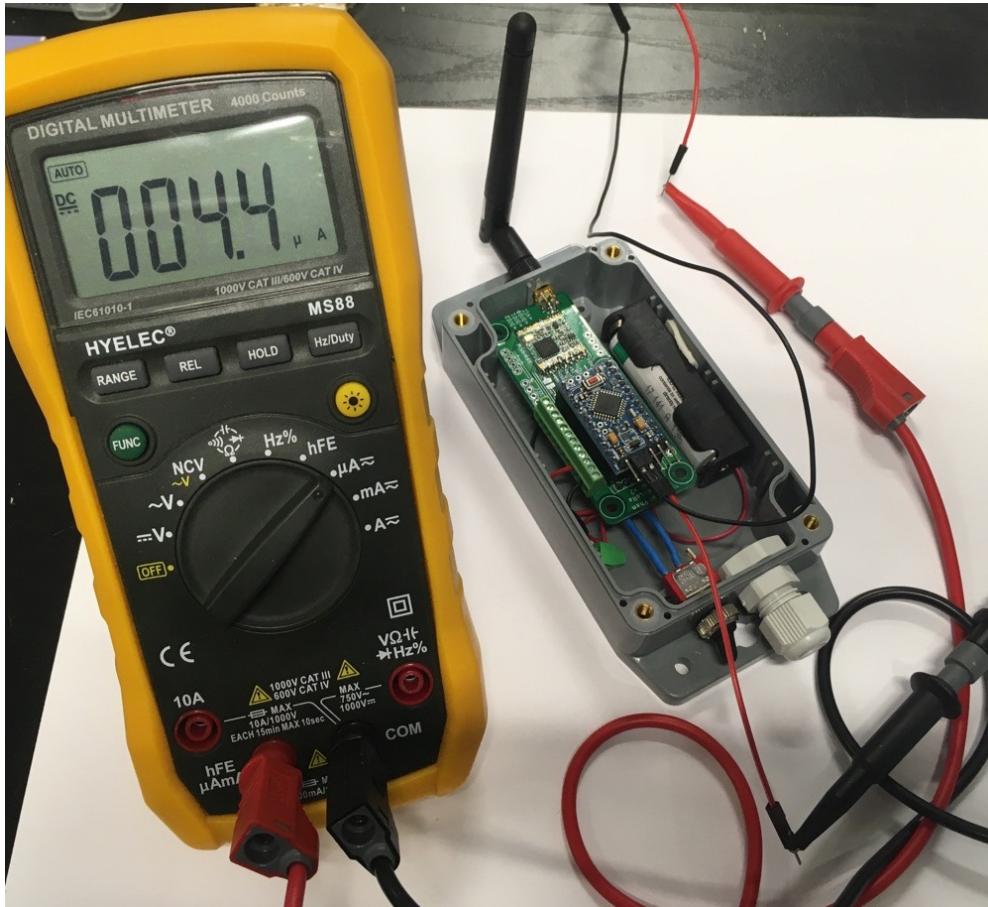
**Do not forget** to remove the power LED by just clipping it off with some wire cutters

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



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

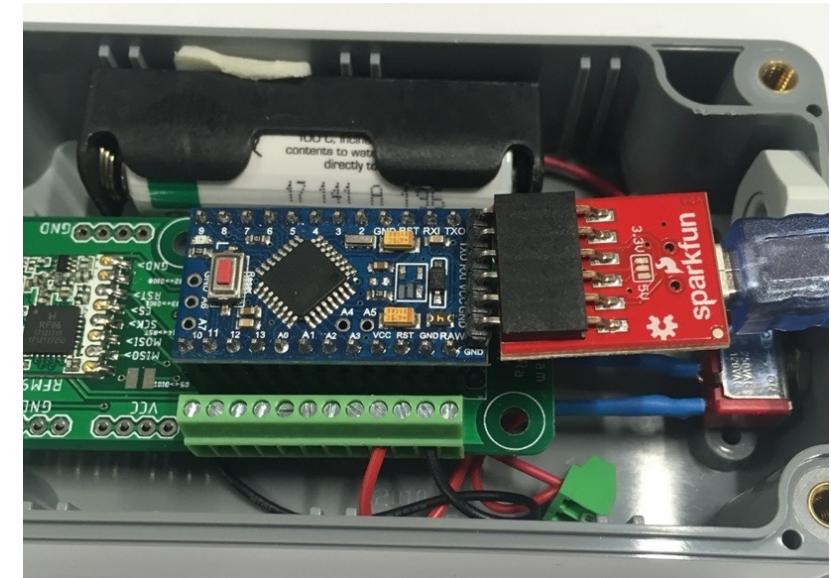
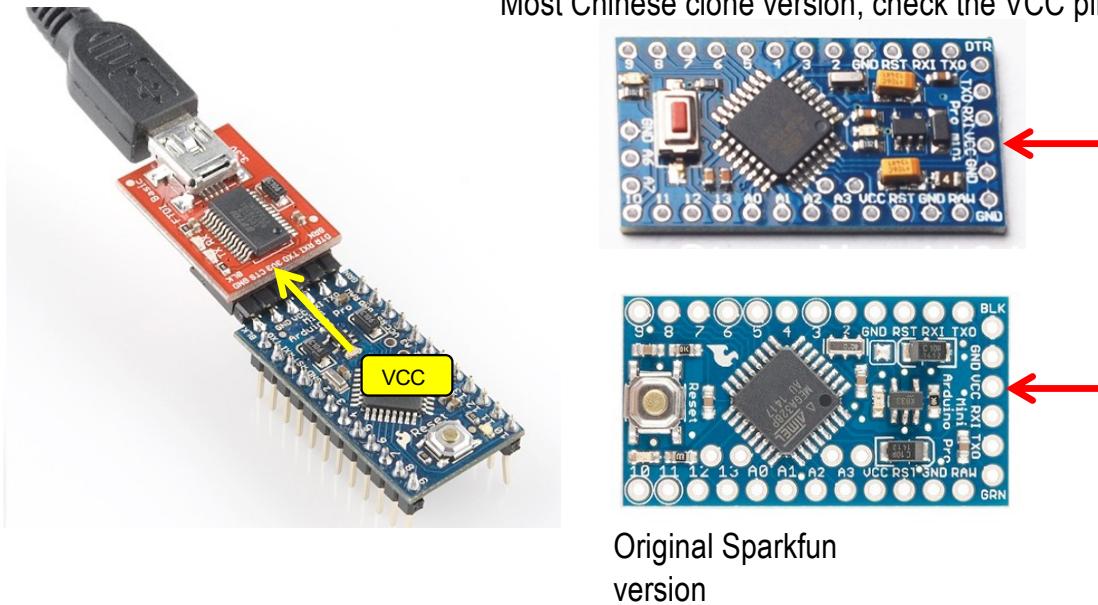
# Power consumption in deep sleep



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 2 AA batteries

# Connecting with an FTDI cable



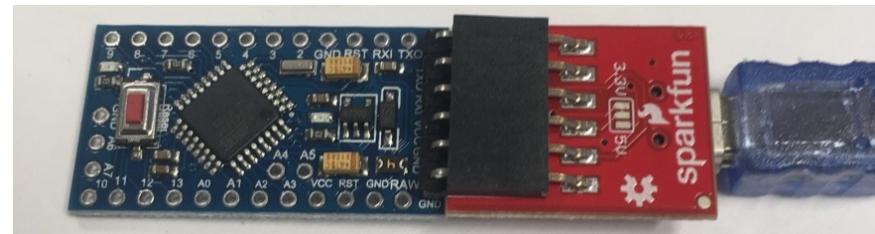
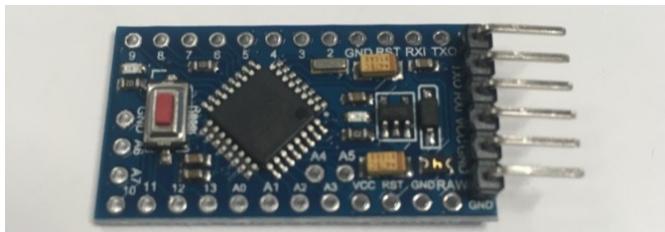
For the ProMini, you need to have an FTDI breakout cable working at 3.3v

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



# 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.

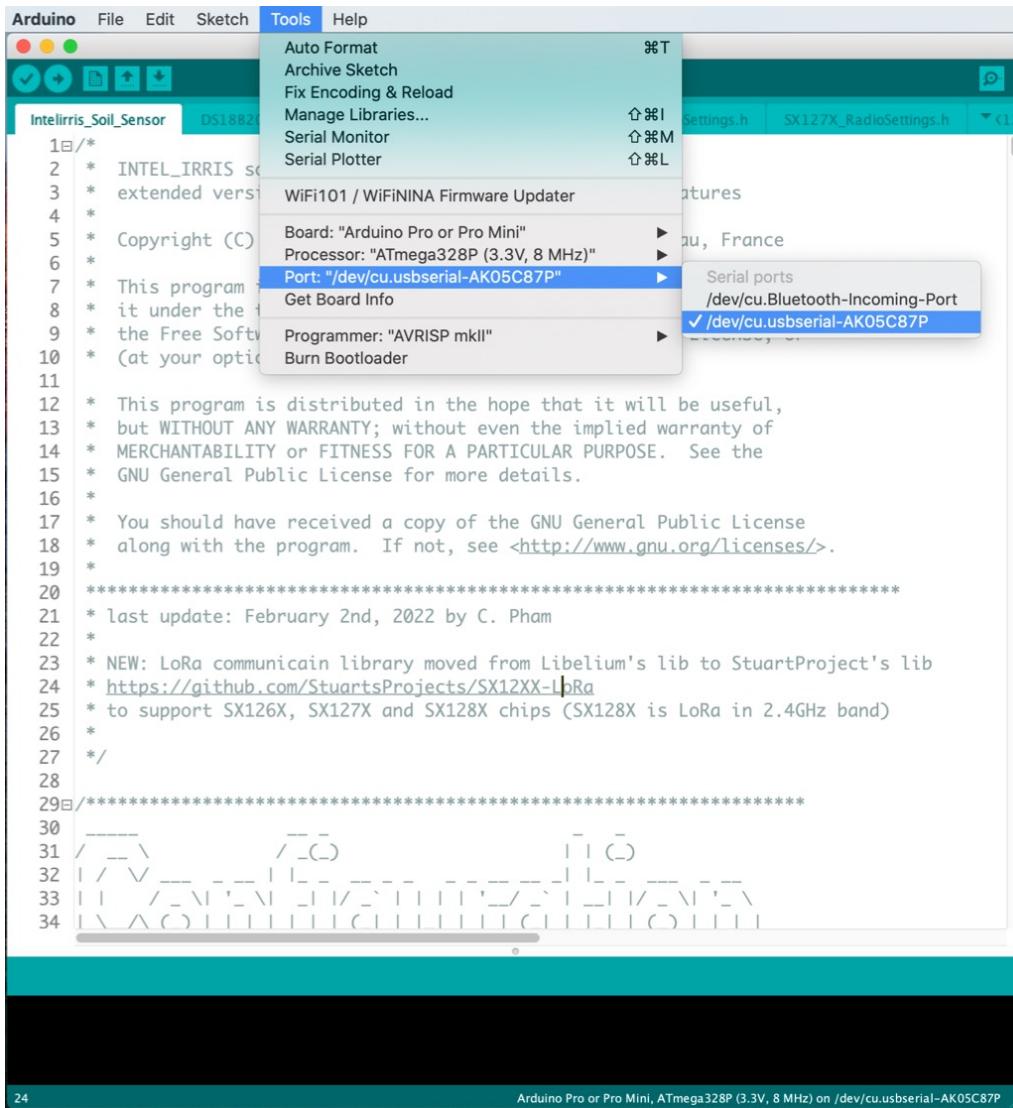
Using library LowPower at version 1.0 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/
Using library OneWire at version 2.3.2 in folder: /Users/cpham/Dropbox/Arduino/sketch/libraries/
Using library Dallas-Temperature at version 3.7.7 in folder: /Users/cpham/Dropbox/Arduino/sketch/
Using library 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

# 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

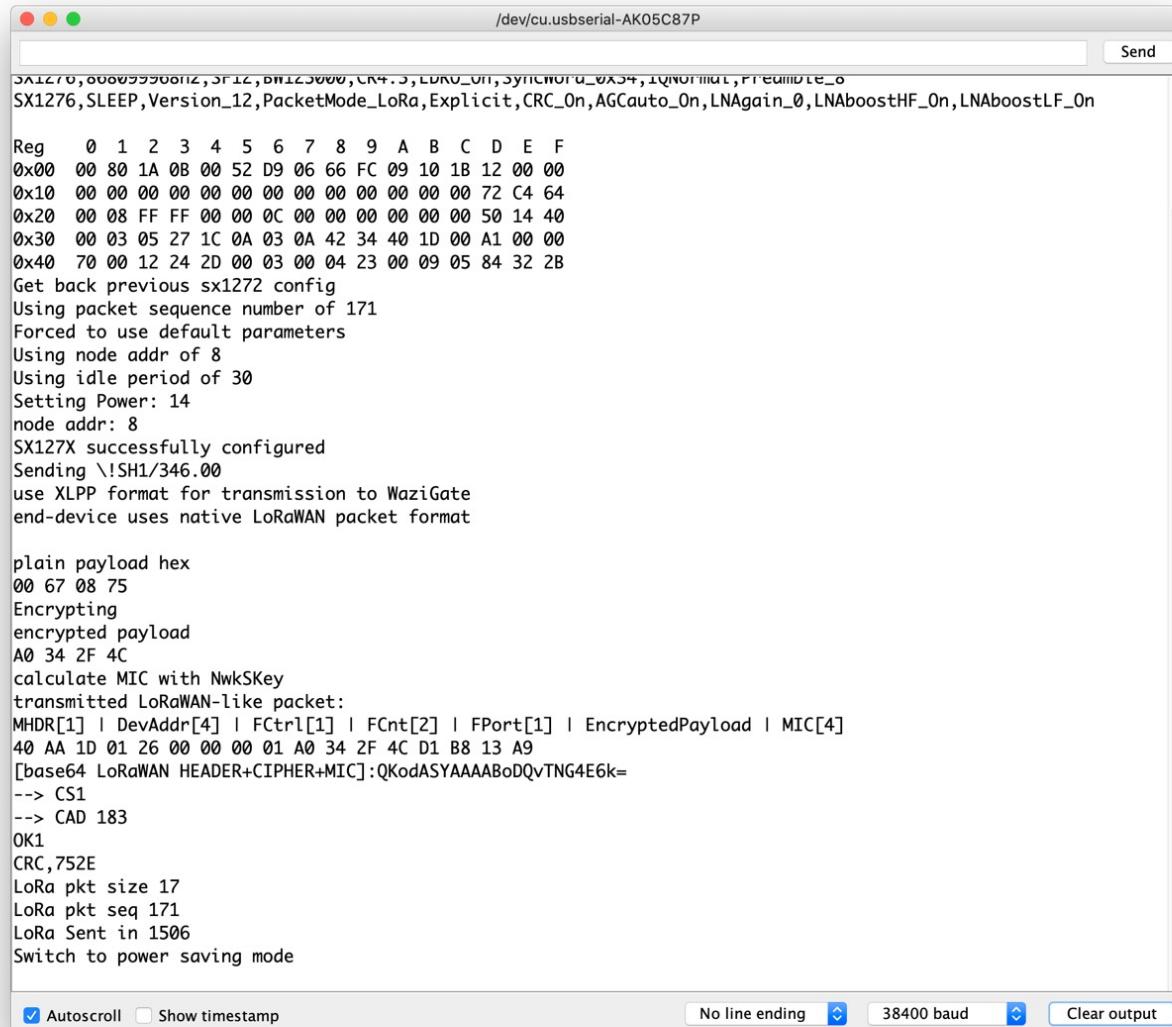
<https://learn.sparkfun.com/tutorials/how-to-install-ch340-drivers/all>

**Click on the "upload" button**



And wait until upload is completed

# Checking that device is operational



```

/dev/cu.usbserial-AK05C87P
Send

SX1276,0000000000000000,3F12,B0123000,CR4.5,LDRU_0H,SYNCHRO_0X34,TQNormal,Preamble_0
SX1276,SLEEP,Version_12,PacketMode_LoRa,Explicit,CRC_On,AGCAuto_On,LNAgain_0,LNAboostHF_On,LNAboostLF_On

Reg 0 1 2 3 4 5 6 7 8 9 A B C D E F
0x00 00 80 1A 0B 00 52 D9 06 66 FC 09 10 1B 12 00 00
0x10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 72 C4 64
0x20 00 08 FF FF 00 00 0C 00 00 00 00 00 00 00 50 14 40
0x30 00 03 05 27 1C 0A 03 0A 42 34 40 1D 00 A1 00 00
0x40 70 00 12 24 2D 00 03 00 04 23 00 09 05 84 32 2B
Get back previous sx1272 config
Using packet sequence number of 171
Forced to use default parameters
Using node addr of 8
Using idle period of 30
Setting Power: 14
node addr: 8
SX127X successfully configured
Sending \!SH1/346.00
use XLPP format for transmission to WaziGate
end-device uses native LoRaWAN packet format

plain payload hex
00 67 08 75
Encrypting
encrypted payload
A0 34 2F 4C
calculate MIC with NwkSKey
transmitted LoRaWAN-like packet:
MHDR[1] | DevAddr[4] | FCtrl[1] | FCnt[2] | FPort[1] | EncryptedPayload | MIC[4]
40 AA 1D 01 26 00 00 00 01 A0 34 2F 4C D1 B8 13 A9
[base64 LoRaWAN HEADER+CIPHER+MIC]:QKodASYAAABoDQvTNG4E6k=
--> CS1
--> CAD 183
OK1
CRC,752E
LoRa pkt size 17
LoRa pkt seq 171
LoRa Sent in 1506
Switch to power saving mode

 Autoscroll  Show timestamp
  No line ending  38400 baud  Clear output

```

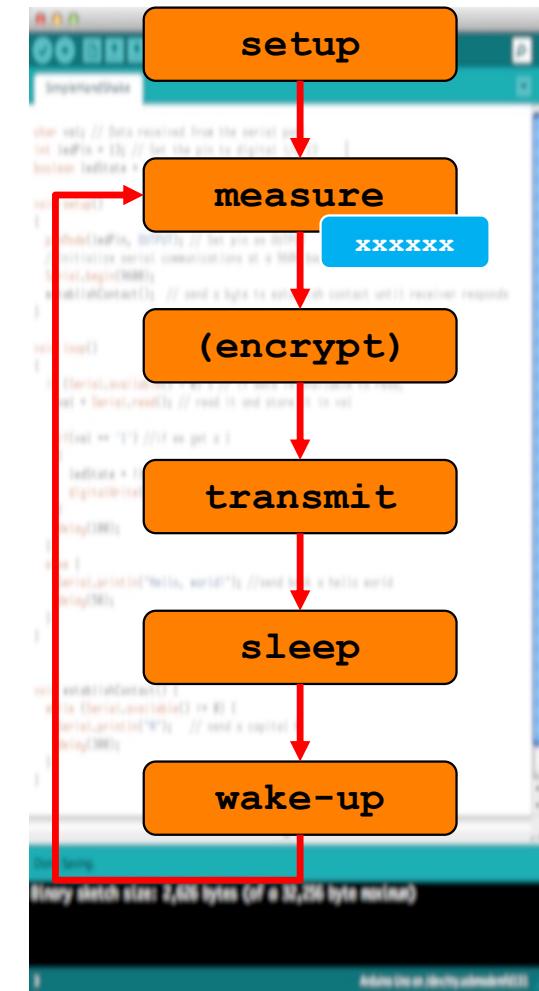
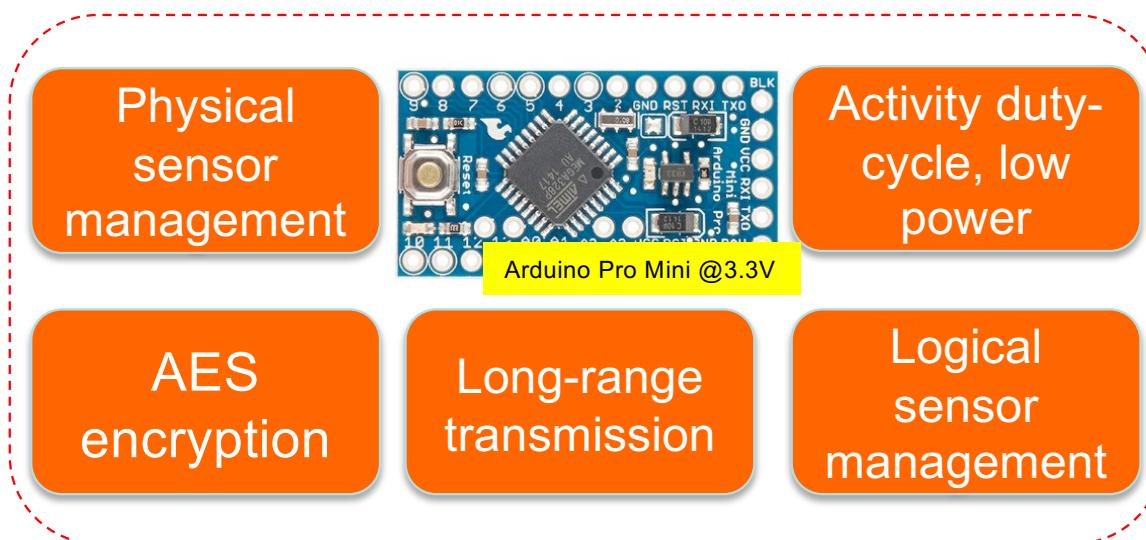
**Open serial monitor**

**Set baud rate to 38400**

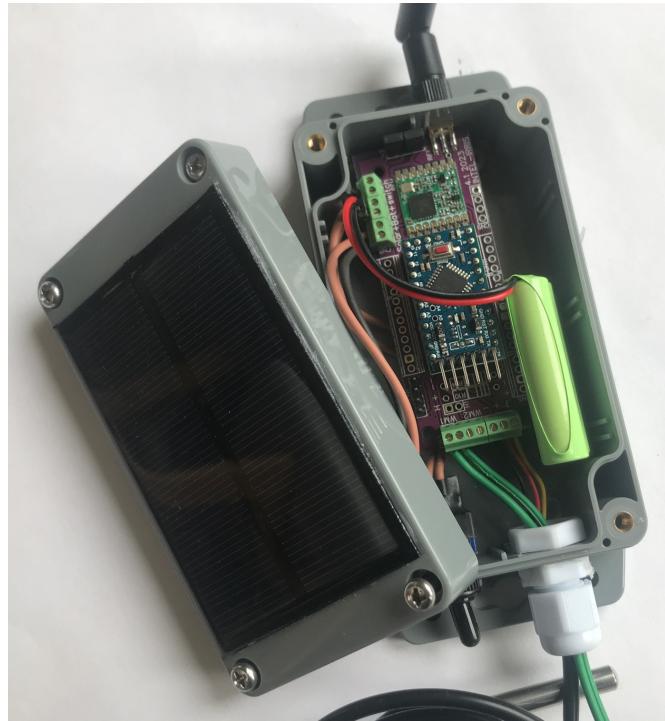
**See output from board**

**Check that  
transmission is OK**

SEN0308  
capacitive soil humidity  
A0 (signal), A1 (power)



# Transmission to WaziGate



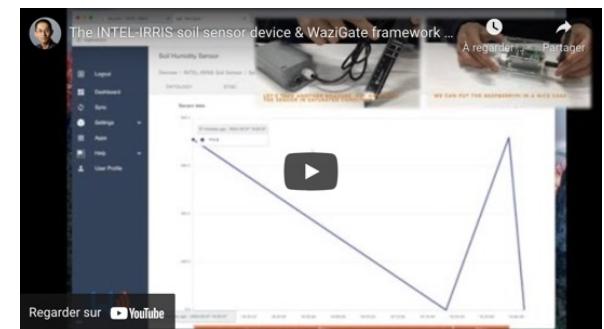
Parameters for  
INTEL-IRRIS WaziGate



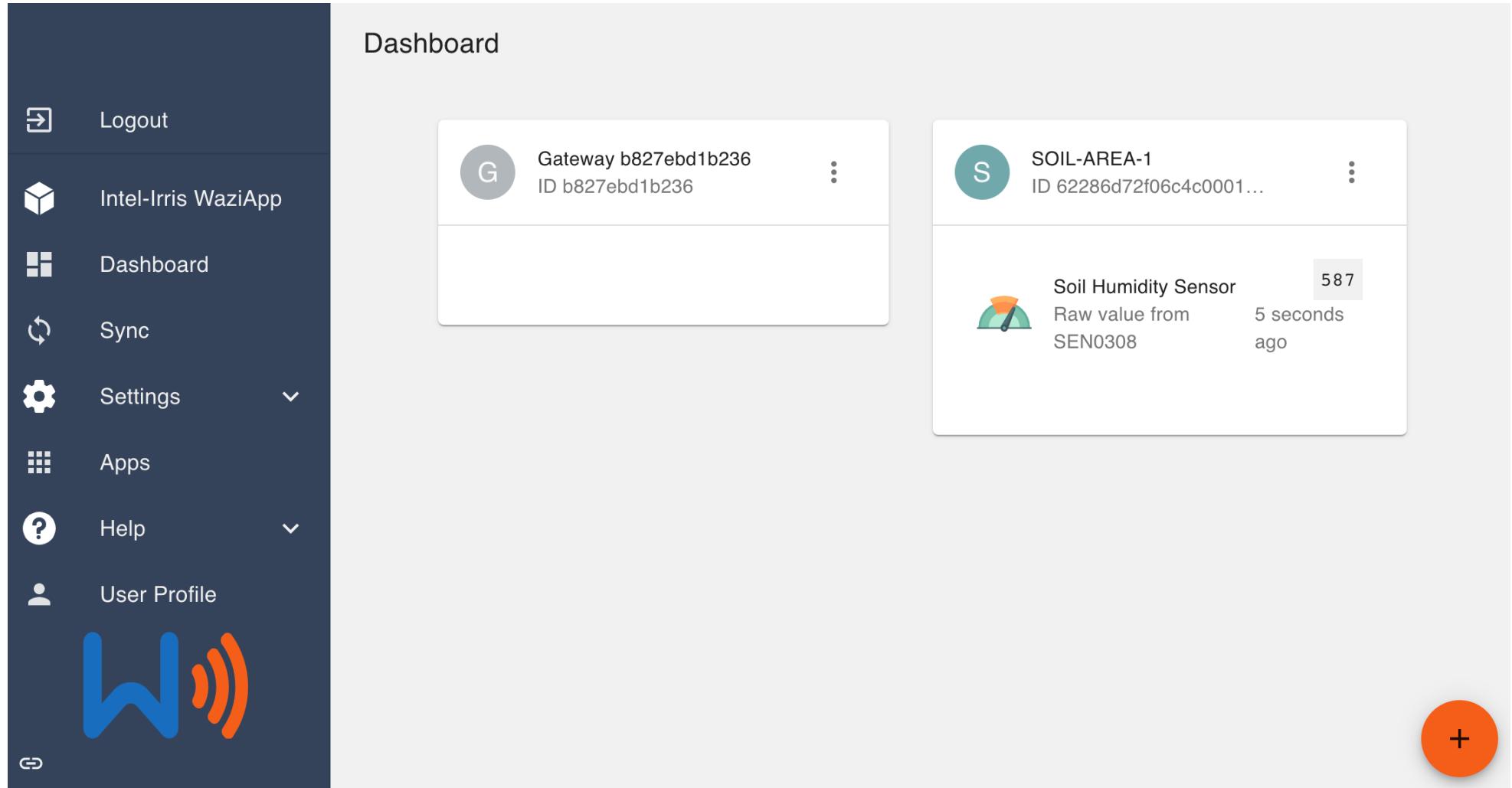
SF12BW125  
868.1MHz | 433.175MHz  
Node id is 26011DAA  
1 msg/60mins  
1 sensor  
XLPP data



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>



# See it on the dashboard



The screenshot shows the Intel-Irris WaziApp dashboard. On the left is a dark sidebar with white icons and text:

- Logout
- Intel-Irris WaziApp
- Dashboard
- Sync
- Settings
- Apps
- Help
- User Profile

At the bottom of the sidebar is a blue icon with a blue 'W' and orange signal waves.

The main area is titled "Dashboard". It displays two cards:

- Gateway b827ebd1b236**  
ID b827ebd1b236
- SOIL-AREA-1**  
ID 62286d72f06c4c0001...

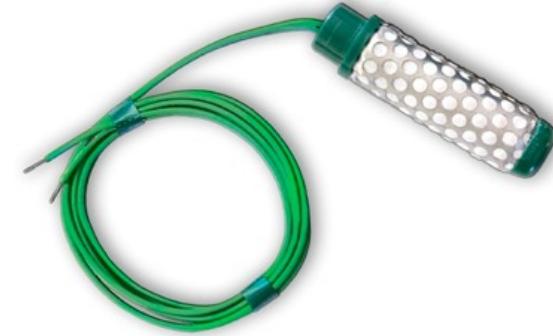
Below these cards is a detailed view of the Soil Humidity Sensor card:

- Soil Humidity Sensor**
- Raw value from SEN0308
- 587 (value)
- 5 seconds ago

A red circular button with a white plus sign is located in the bottom right corner of the dashboard area.

# Software configuration for WM

- Default resistor value is 10kOhms
- Change in `watermark.h` file  
the `WM_RESISTOR` value to match  
the one you are using if needed
- The fully assembled version has a 10kOhms resistor

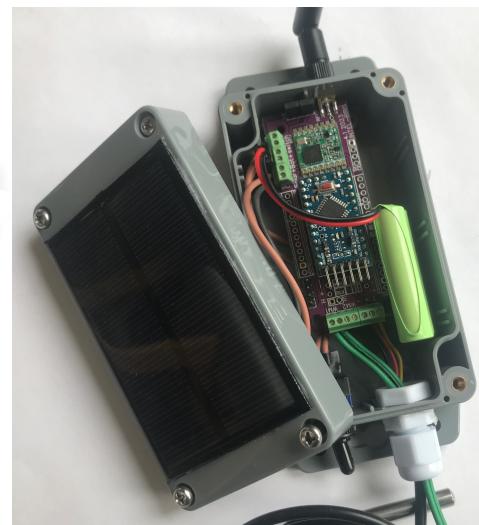


```
//put here the resistor value, in Ohms
#define WM_RESISTOR 10000
//we defined WM_MAX_RESISTOR=32760 because the transmitted value would be 32760/10=3276
//and currently a bug in WaziGate XLPP decoding code will limit the maximum value to 3276
#define WM_MAX_RESISTOR 32760
```

- ```
///////////
// uncomment to have a soil tensiometer watermark sensor
#define WITH_WATERMARK
#define WM_REF_TEMPERATURE 28.0
/////////
// uncomment to force the watermark to have default device address for WaziGate
//#define WM_AS_PRIMARY_SENSOR
```

# Transmission with the WM sensor

- Devices with a WM sensor will have a different default address
  - 26011DB1 instead of 26011DAA for capacitive
- 2 values are transmitted
  - Centibars converted from resistance value
  - Raw resistance value measured for WM sensor, but scaled down by a factor of 10 → 300 = 3000 Ohms



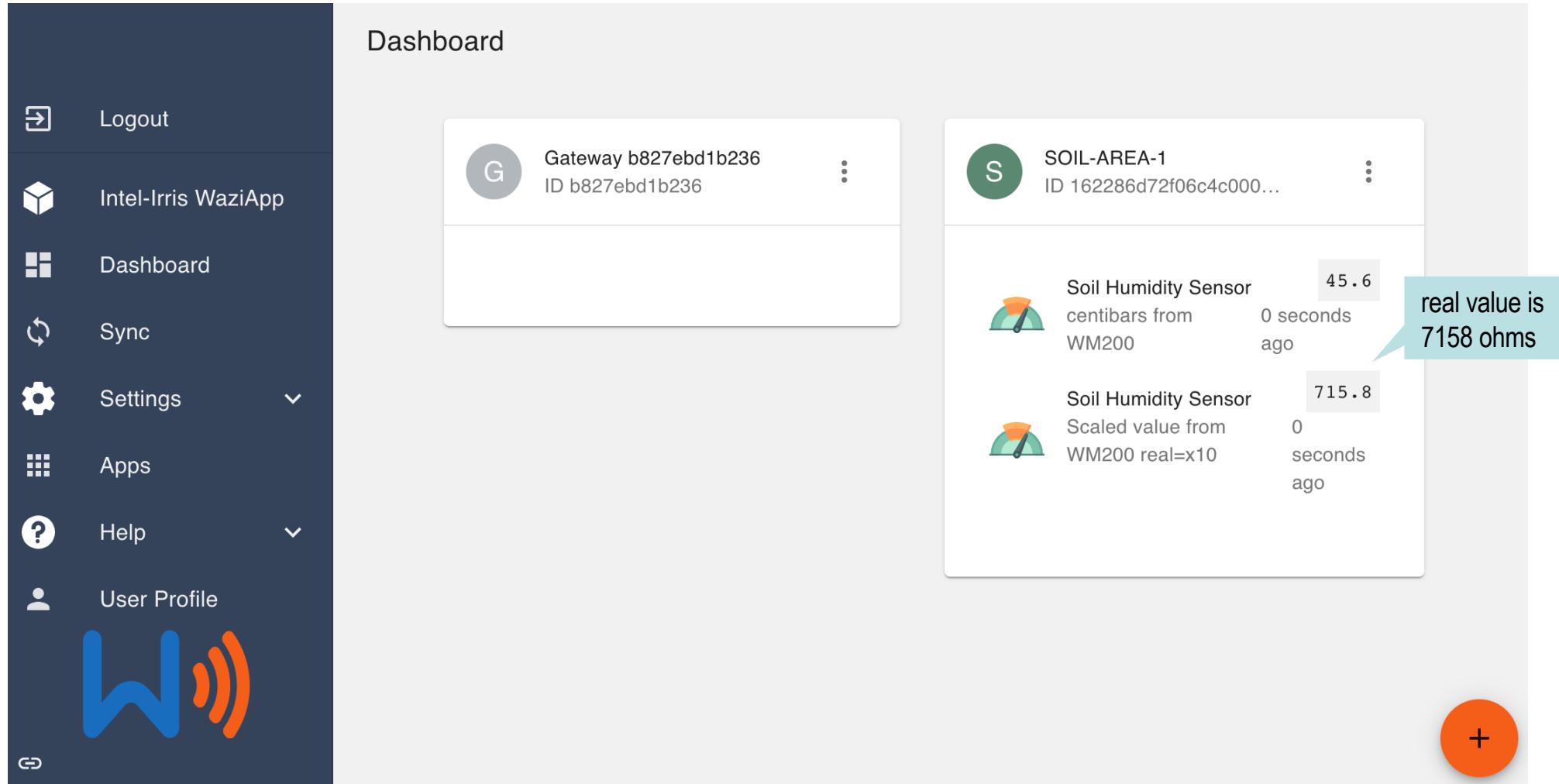
Parameters for  
INTEL-IRRIS WaziGate

 **LoRaWAN™**

SF12BW125  
 868.1MHz | 433.175MHz  
 Node id is 26011DB1  
 1 msg/60mins  
 1 sensor  
 XLPP data



# See it on the dashboard



The screenshot shows the Intel-Irris WaziApp dashboard. On the left is a sidebar with the following menu items:

- Logout
- Intel-Irris WaziApp
- Dashboard
- Sync
- Settings
- Apps
- Help
- User Profile

At the bottom of the sidebar is a blue icon consisting of a stylized 'W' and three orange arcs.

The main area is titled "Dashboard". It displays two main components:

- A card for a "Gateway" with ID b827ebd1b236, represented by a grey circle with a white letter "G".
- A card for "SOIL-AREA-1" with ID 162286d72f06c4c000..., represented by a green circle with a white letter "S".

Under the "SOIL-AREA-1" card, there are two entries for a "Soil Humidity Sensor":

- centibars from WM200: 45.6 (0 seconds ago)
- Scaled value from WM200 real=x10: 715.8 (0 seconds ago)

A light blue callout bubble points to the value "715.8" with the text "real value is 7158 ohms".

# 2 versions of the soil device



# Soil temperature with WM sensor (1)

- Without a real temperature sensor, the INTEL-IRRIS code uses a default soil temperature set to 28°C to convert the resistance value into centibars (see <https://www.irrometer.com/200ss.html>)

```
//////////  
// uncomment to have a soil tensiometer watermark sensor  
#define WITH_WATERMARK  
#define WM_REF_TEMPERATURE 28.0  
//////////  
// uncomment to force the watermark to have default device address for WaziGate  
//#define WM_AS_PRIMARY_SENSOR
```

- You can change this setting in the code for testing purposes
- The final objective is to use a real temperature sensor to dynamically get the soil temperature

# Soil temperature with WM sensor (2)

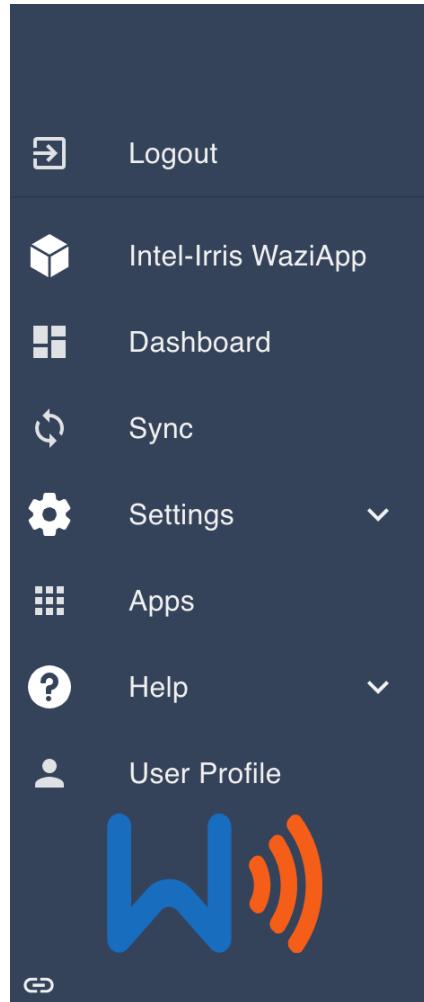
- A DS18B20 temperature sensor can be connected to the device



```
///////////////////////////////  
// uncomment to have 1 soil temperature sensor ST  
// using a one-wire DS18B20 sensor  
#define SOIL_TEMP_SENSOR  
//#define LINK_SOIL_TEMP_TO_CENTIBAR
```

- In this case, the soil temperature will also be sent to the gateway
- It is also possible to link the measured soil temperature to the centibar calculation performed by the device itself

# See it on the dashboard



- [!\[\]\(d1572bfe46d45c861fd8d8d40ba1c337\_img.jpg\) Logout](#)
- [!\[\]\(a23a5d20409a08093440697492a7dc02\_img.jpg\) Intel-Irris WaziApp](#)
- [!\[\]\(a55ca8cfea2c43ba6e0afedbc40e939e\_img.jpg\) Dashboard](#)
- [!\[\]\(1d255a6a05f1608f7bdc321158e98929\_img.jpg\) Sync](#)
- [!\[\]\(551071d6fdf8ce437e81add21a90ce9e\_img.jpg\) Settings](#)
- [!\[\]\(92357c0da7753295516f5c4f7cdbdfad\_img.jpg\) Apps](#)
- [!\[\]\(786ae2d7d63650af84a5e5e8f283d55f\_img.jpg\) Help](#)
- [!\[\]\(397b2887bd4884aaaea34cf6252d3497\_img.jpg\) User Profile](#)



## Dashboard

 **Gateway b827ebd1b236**  
ID b827ebd1b236

 **SOIL-AREA-1**  
ID 162286d72f06c4c000...

 Soil Humidity Sensor  
centibars from WM200      45 . 6  
6 seconds ago

 Soil Humidity Sensor  
Scaled value from WM200 real=x10      714 . 5  
6 seconds ago

 Soil Temperature Sensor  
degree Celsius      26 . 3  
6 seconds ago

Here, real soil temperature is 26.3°C but the device is not linking the centibar conversion with the temperature sensor and uses the default 28°C value. The final centibar conversion could be realized by a dedicated irrigation application run on the WaziGate gateway.

+

Prof. Congduc Pham  
<http://www.univ-pau.fr/~cpham>

70

# Low battery voltage indication

- The INTEL-IRRIS code has low battery voltage indication feature, by default, low battery voltage threshold is set to 2.85V

```
#ifdef WITH_WATERMARK
//the ATMega328P reboots at about 2.65 - 2.75
#define VCC_LOW 2.85
#else
//capacitive sensors can be impacted by low voltage, especially for very dry soil conditions
#define VCC_LOW 2.85
#endif
```

- The battery voltage will be transmitted to the gateway as well and will appear on the WaziGate's dashboard to warn end-user
- When low battery voltage is detected, the measure & transmission time interval will increase from 1h to 4h to preserve battery

# See it on the dashboard

**G** Gateway b827ebd1b236  
ID b827ebd1b236

**S** SOIL-AREA-1  
ID 6314f8f4127dbd0001...

**S** SOIL-AREA-2  
ID 63160162127dbd000...

 Soil Humidity Sensor 402.5  
Raw value from SEN0308 8 minutes ago

 Battery voltage 3.48  
volt, low battery when lower than 2.85V 8 minutes ago

 Soil Humidity Sensor 132.9  
centibars from WM200 42 minutes ago

 Soil Humidity Sensor 2045.9  
scaled value from WM200 real=x10 42 minutes ago

 Soil Temperature Sensor 22.3  
degree Celsius 42 minutes ago

 Battery voltage 2.85  
volt, low battery when lower than 2.85V 42 minutes ago

# Software configuration for 2 WM

- Uncomment in main program TWO\_WATERMARK

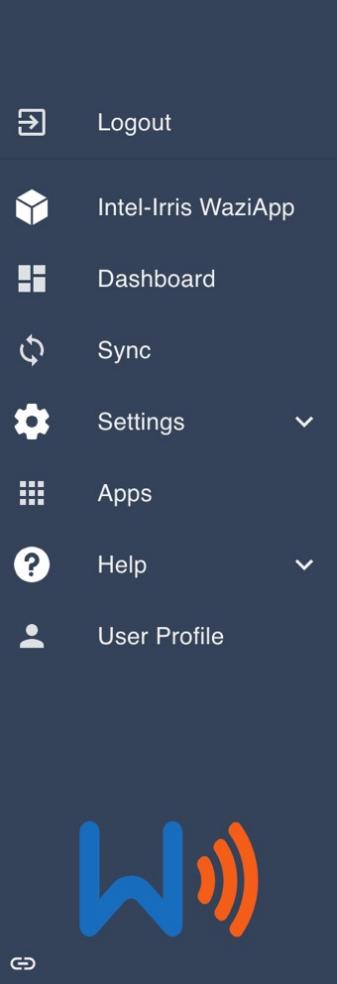
```
///////////////////////////////  
// uncomment to have a soil tensiometer watermark sensor  
#define WITH_WATERMARK  
#define WM_REF_TEMPERATURE 28.0  
  
///////////////////////////////  
// uncomment to force the watermark to have default device address for WaziGate  
//#define WM_AS_PRIMARY_SENSOR  
  
///////////////////////////////  
// uncomment to have 2 tensiometer watermark sensor on the same device  
#define TWO_WATERMARK
```



- You can still have a soil temperature sensor



# See it on the dashboard



- Logout
- Intel-Irris WaziApp
- Dashboard
- Sync
- Settings
- Apps
- Help
- User Profile

**Dashboard**

G Gateway b827ebd1b236  
ID b827ebd1b236

S SOIL-AREA-1  
ID 162286d72f06c4c000...

Watermark 1  
centibars from WM200

Watermark 1  
Scaled value from WM200 real=x10

Watermark 2  
centibars from WM200

Watermark 2  
Scaled value from WM200 real=x10

Soil Temperature Sensor  
degree Celsius

+

Here, the device has been compiled to link the centibar conversion with the real value measured by the temperature sensor

If the default 28°C value was used, centibar for WM1 with 1374 ohms would have been 13.1 and centibar for WM2 with 685 ohms would have been 2.0