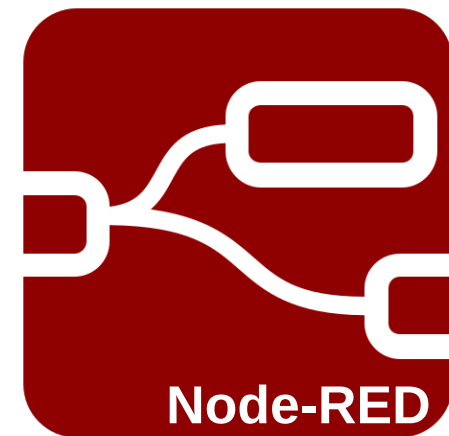
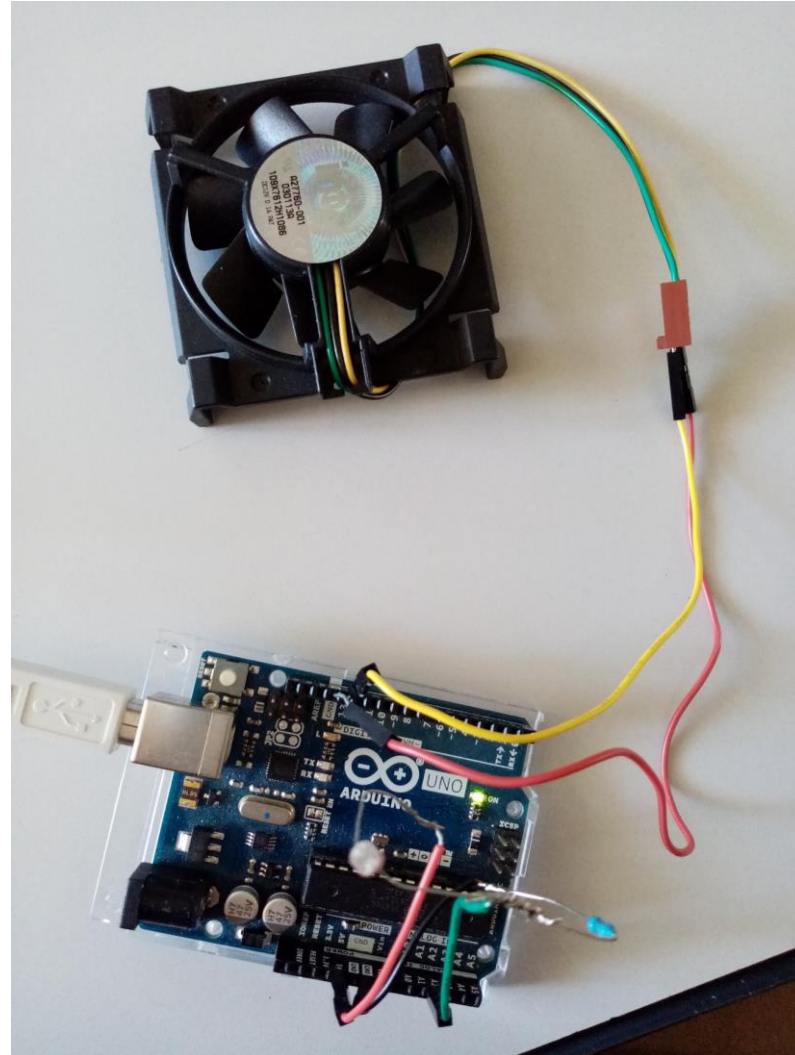


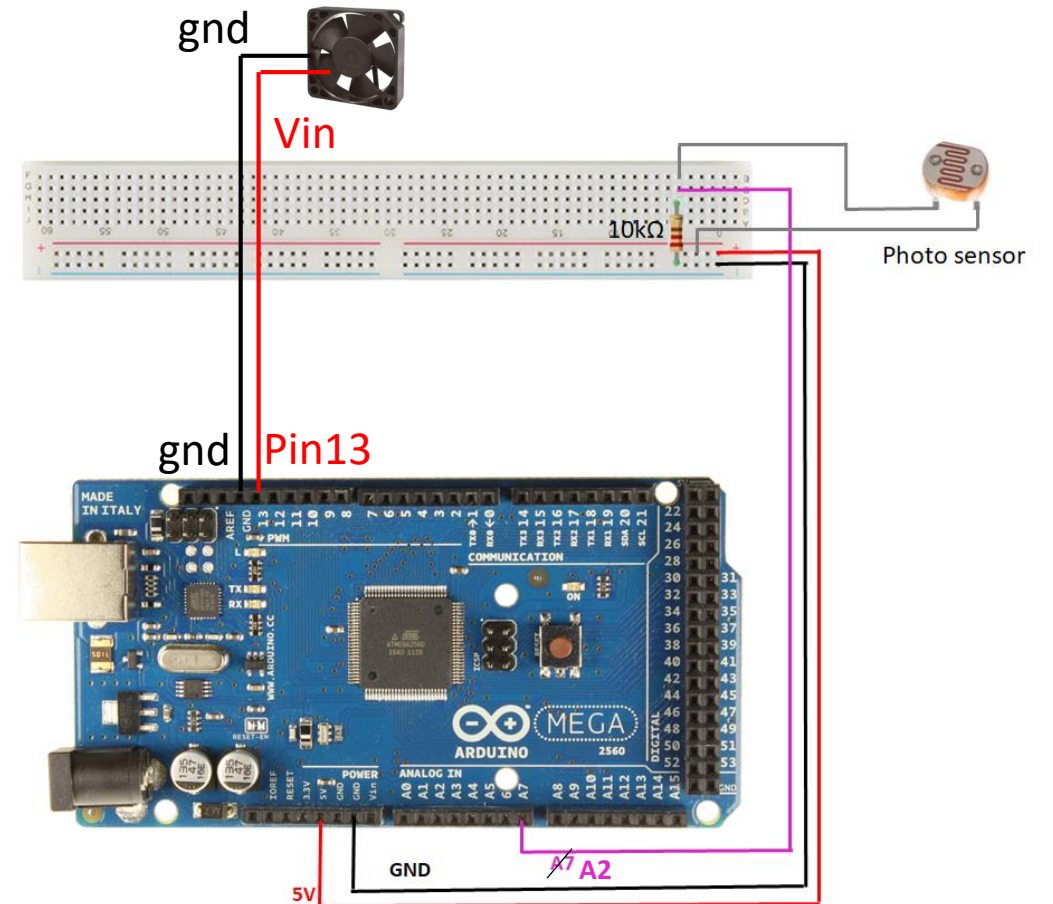
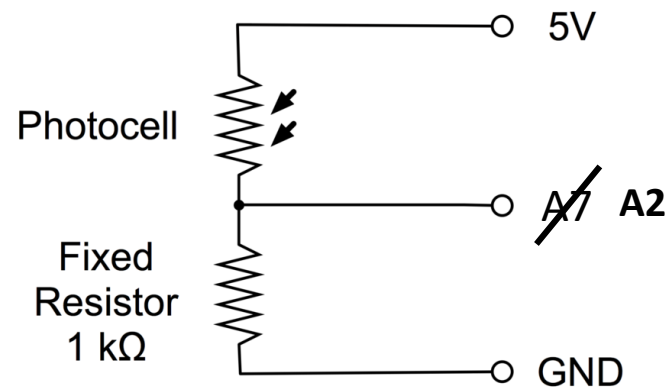
# TP Arduino

Julien Zoubian & Pierre Barrillon



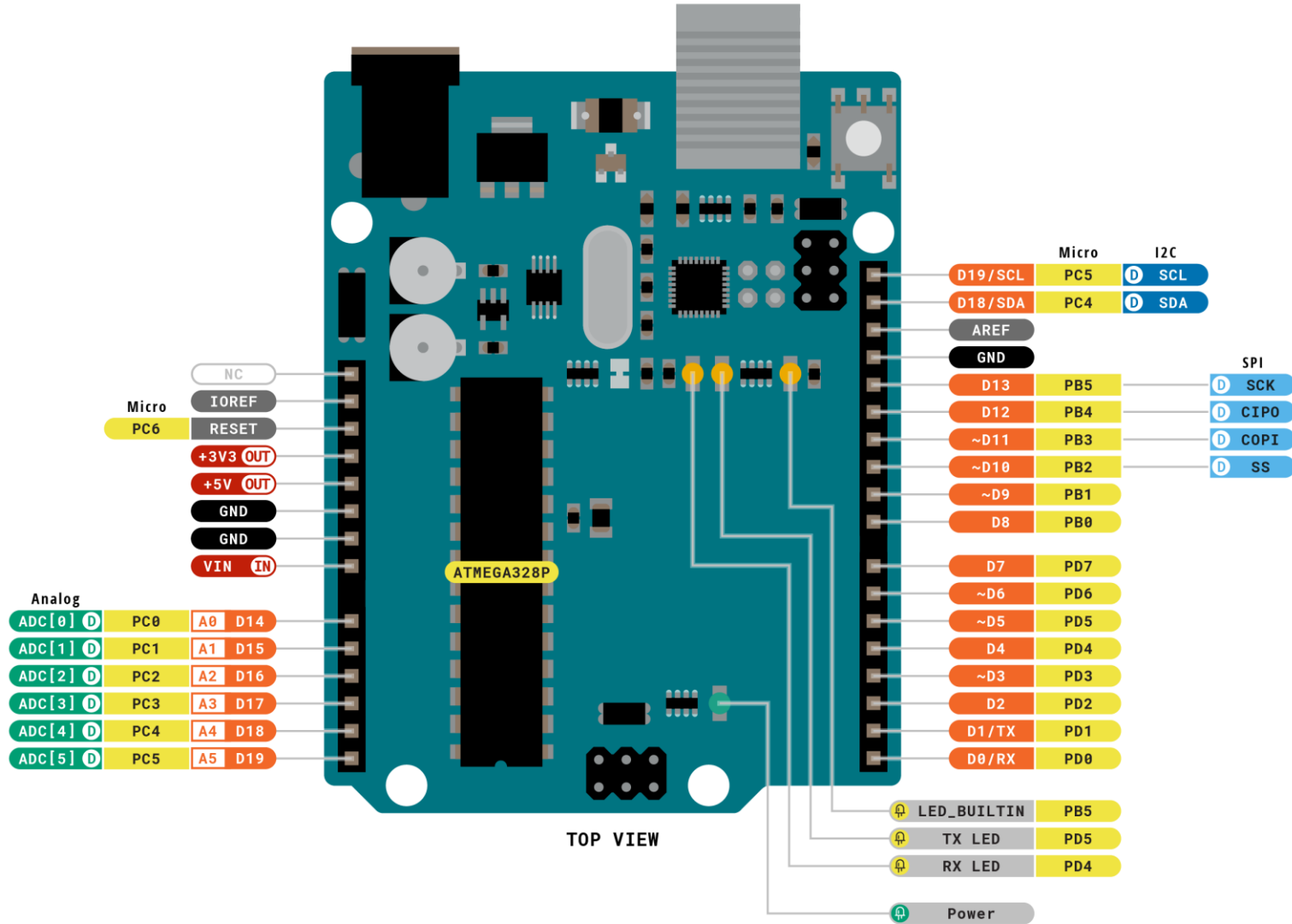
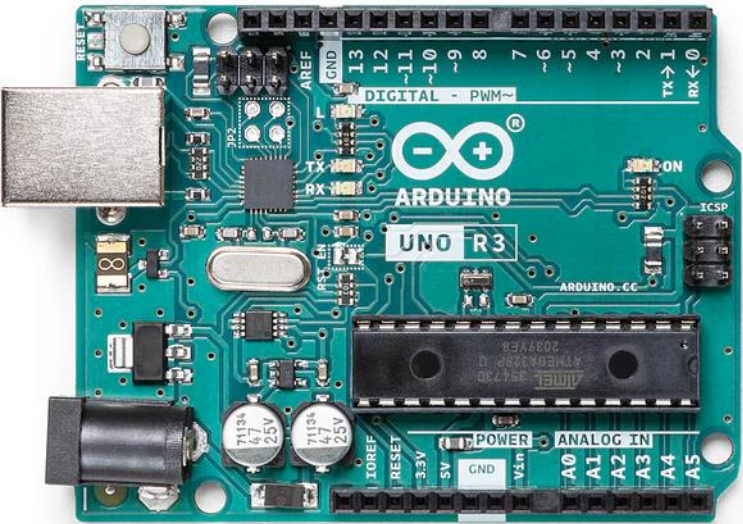
# Introduction

- **But:**
  - Lecture d'un capteur de lumière « photo – resistor » et visualisation
  - Commande d'un ventilateur
- **Matériel:**
  - Arduino UNO
  - Photo-resistor + résistance de 1kOhm
  - Ventilateur 5V



# Arduino UNO

- Carte microcontrôleur ATmega328P.
- 14 pins I/O digital, 6 pins analog Input
- Connection USB
- Produit 5V et 3.3V
- Communications I2C et SPI disponibles



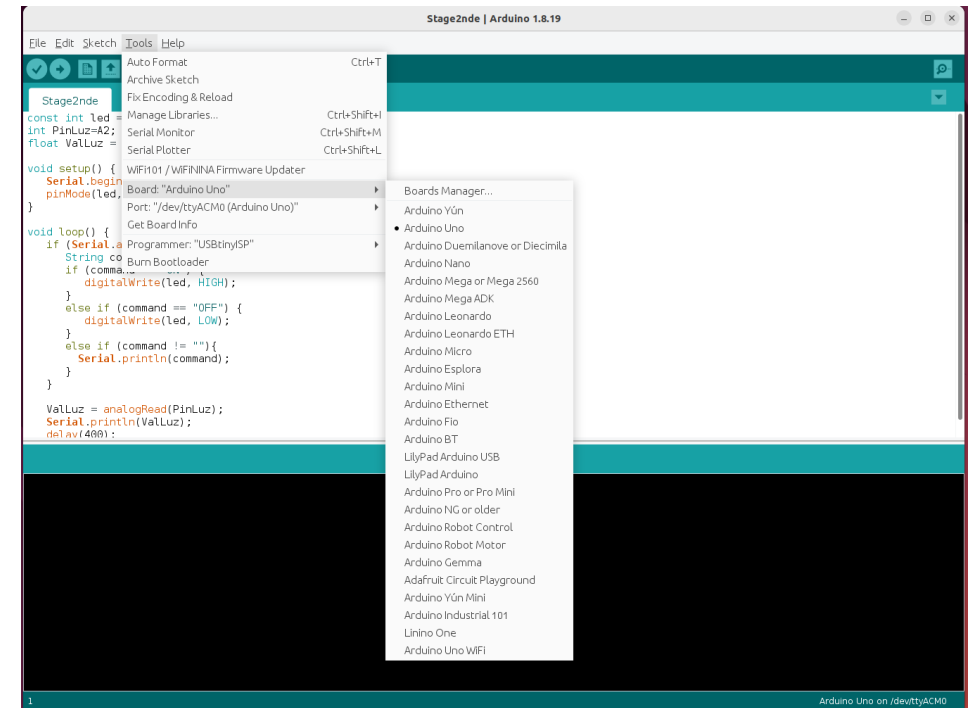
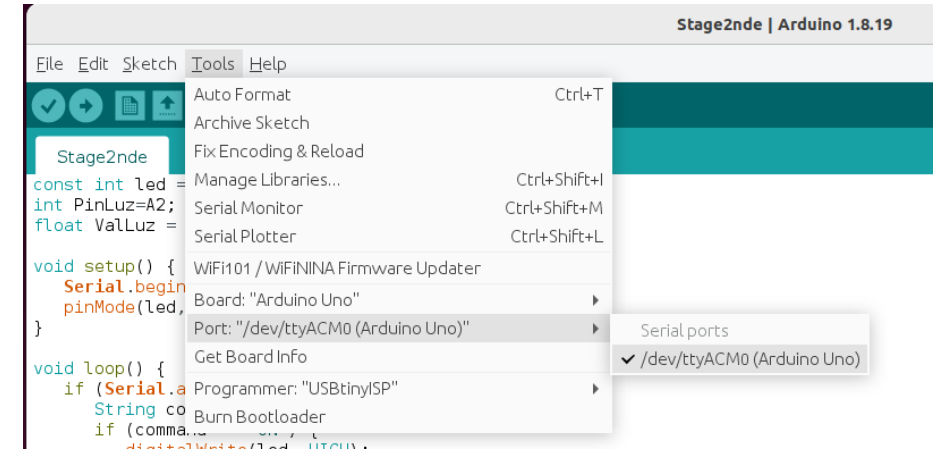
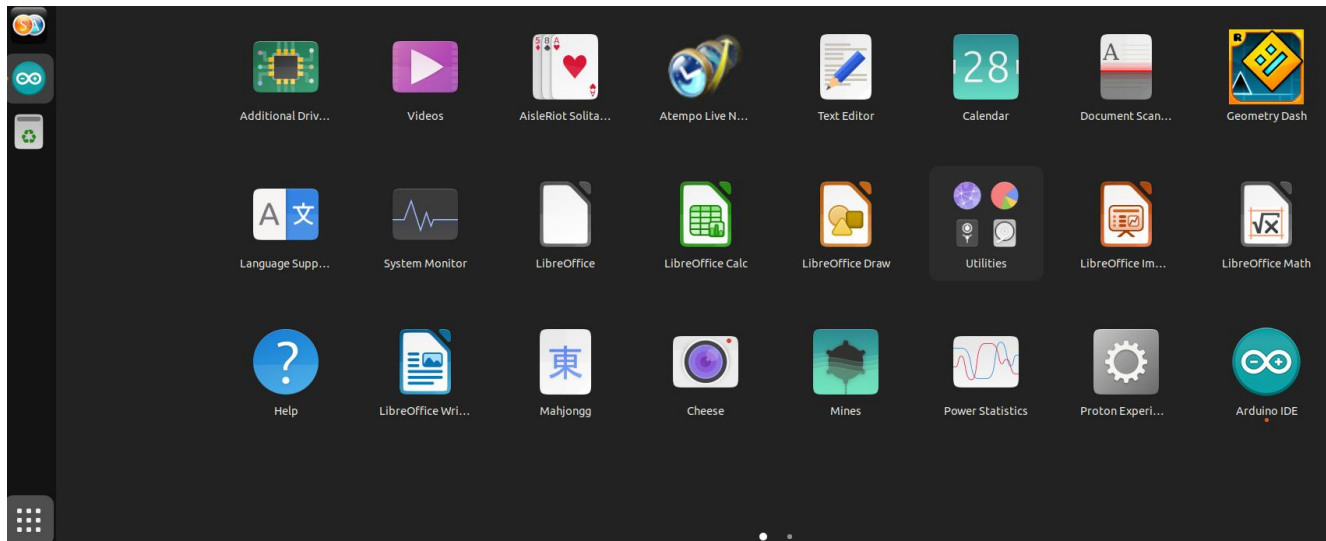
|  |   |  |
|--|---|--|
| <b>Legend:</b>   | <span style="color: orange;">■</span> Digital   | <span style="color: blue;">■</span> I2C      |
| <span style="color: red;">■</span> Power                             | <span style="border: 1px solid black; display: inline-block; width: 10px; height: 10px;"></span> Analog | <span style="color: lightblue;">■</span> SPI |
| <span style="background-color: black; color: black;">■</span> Ground | <span style="background-color: yellow;">■</span> Main Part  | <span style="color: green;">■</span> Analog  |



ARDUINO UNO REV3  
SKU code: A000066  
Pinout  
Last update: 6 Oct, 2022

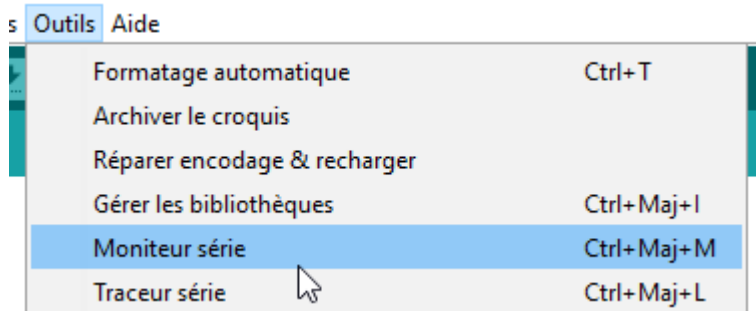
# Démarrage

- Connection au PC. Demander le mdp.
- Rechercher et lancer le logiciel IDE Arduino (show applications) ou ouvrir le fichier Stage2nde.ino (demandez où il se trouve).
- Le code est décrit juste après.
- Il faudra le « téléverser » dans la carte Arduino en sélectionnant le bon type de carte et le bon port.

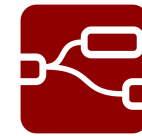


# Code Arduino

- Définition des Pins utilisées
- Définition du paramètre mesuré (intensité lumineuse)
- Démarrage de la liaison série
- Lecture en boucle de la valeur du paramètre sur la pin
- Ecriture sur la liaison série de cette valeur
- Attente de 500 ms (ou autre valeur) avant la prochaine mesure
- Lecture sur la liaison série pour détecter l'allumage/extinction du ventilateur via une commande ON/OFF
- Téléversement vers l'arduino
- Vérification sur le moniteur série



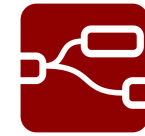
# Description rapide de node-red



<https://nodered.org/>

- *Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.*
- *It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.*
- Une fois installé et exécuté on accède à l'interface pour coder à cette adresse : <http://127.0.0.1:1880>. On accède au dashboard (visualisation) à celle ci : <http://127.0.0.1:1880/ui>.
- Utiliser pour du contrôle-commande ou du monitoring.
- Programmation graphique avec des nœuds à connecter + javascript pour rajouter des algorithmes.
- Entre chaque nœud connecté, le paramètre *msg.payload* est transmis

# Démarrage node-red



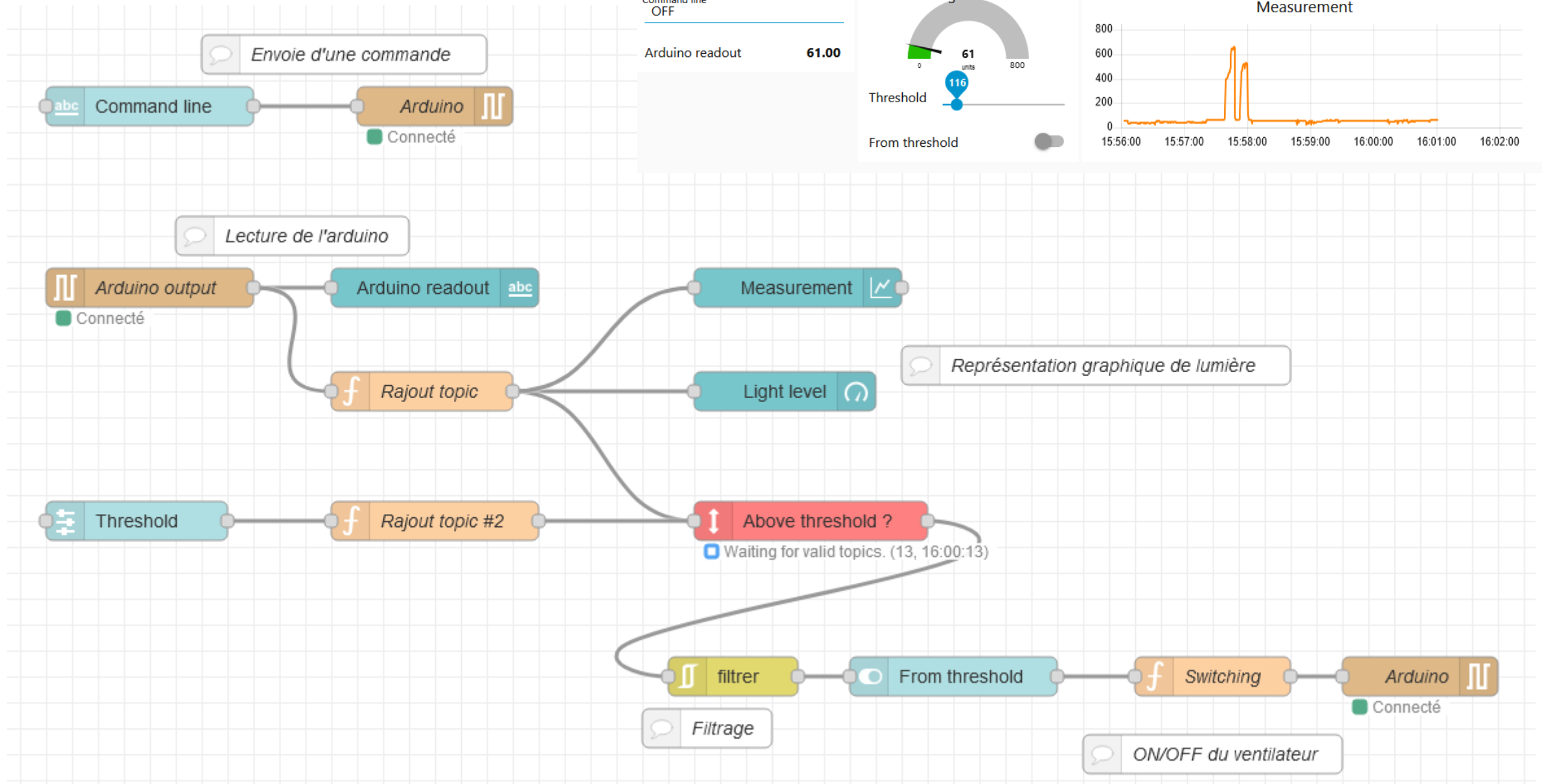
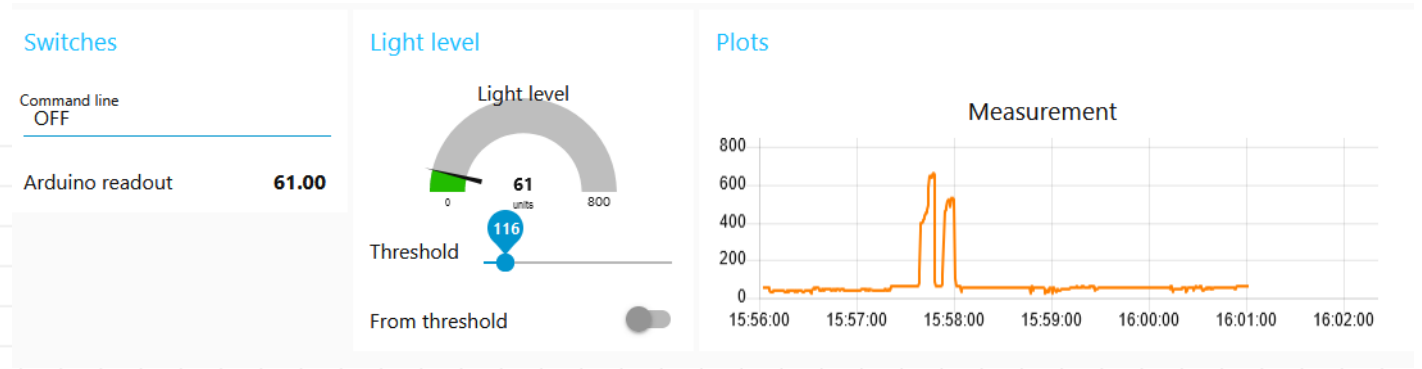
- Dans un terminal taper : *node-red*
- Dans un navigateur rentrer <http://127.0.0.1:1880>

```
barrillon@marmaille:~$ node-red
4 Dec 09:46:28 - [info]

Bienvenue sur Node-RED
=====

4 Dec 09:46:28 - [info] Node-RED version: v3.1.1
4 Dec 09:46:28 - [info] Node.js version: v20.10.0
4 Dec 09:46:28 - [info] Linux 6.2.0-37-generic x64 LE
4 Dec 09:46:28 - [info] Chargement des noeuds de la palette
4 Dec 09:46:29 - [info] Fichier de paramètres : /home/barrillon/.node-red/settings.js
```

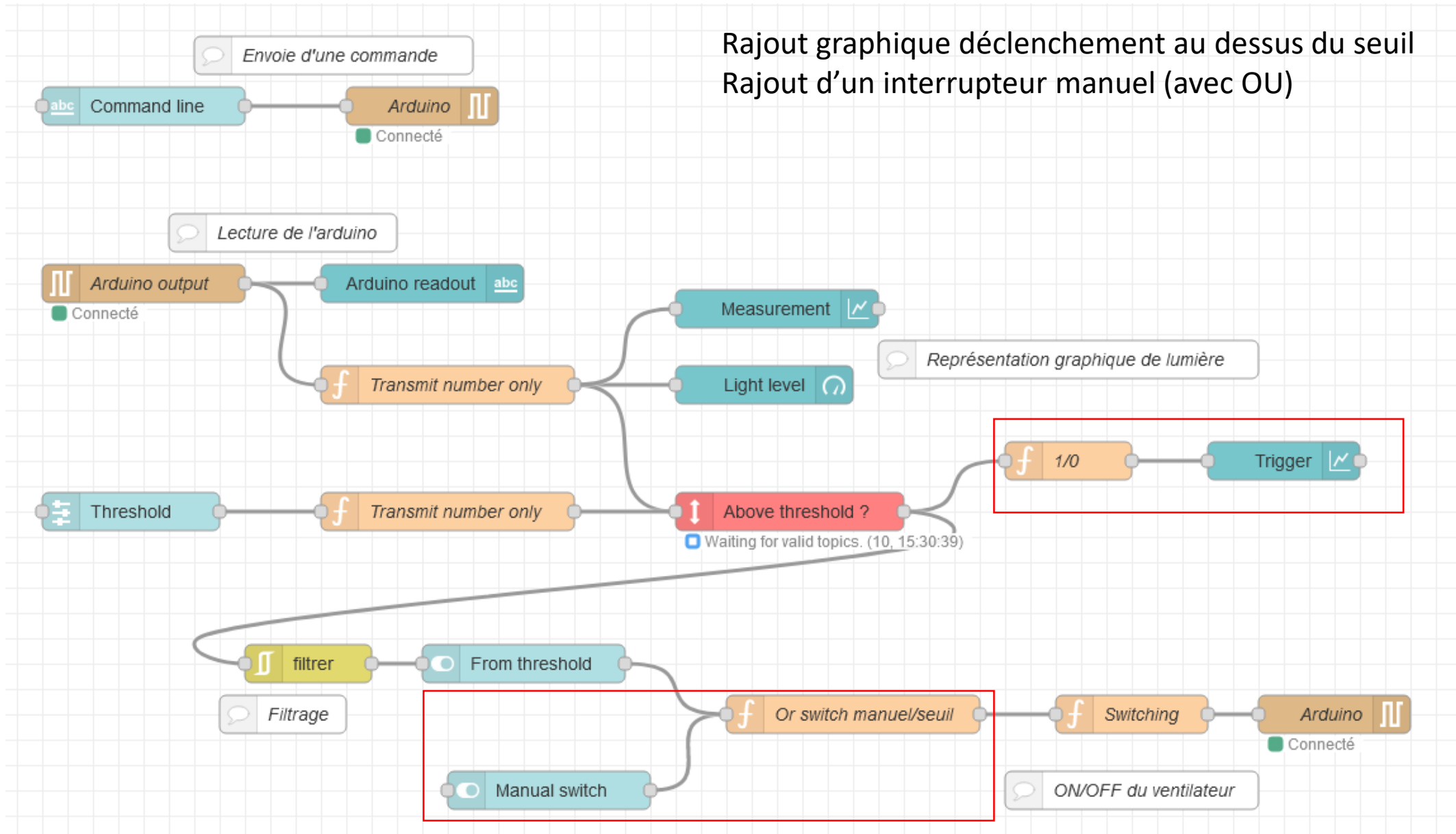
# Code final : lvl 1





# Code final : lvl2

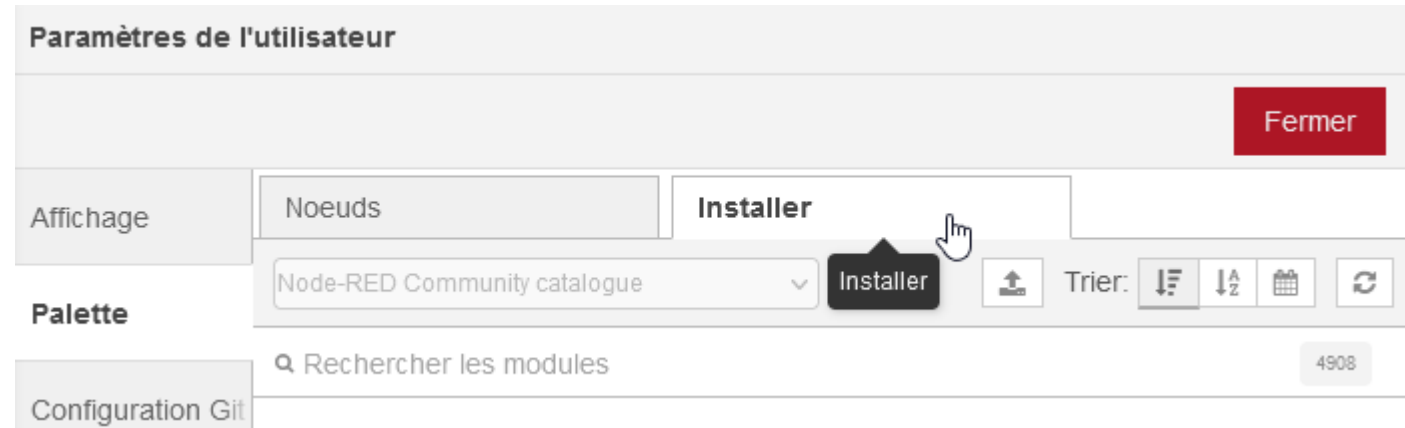
Rajout graphique déclenchement au dessus du seuil  
Rajout d'un interrupteur manuel (avec OU)



# Palette et installation de nœuds

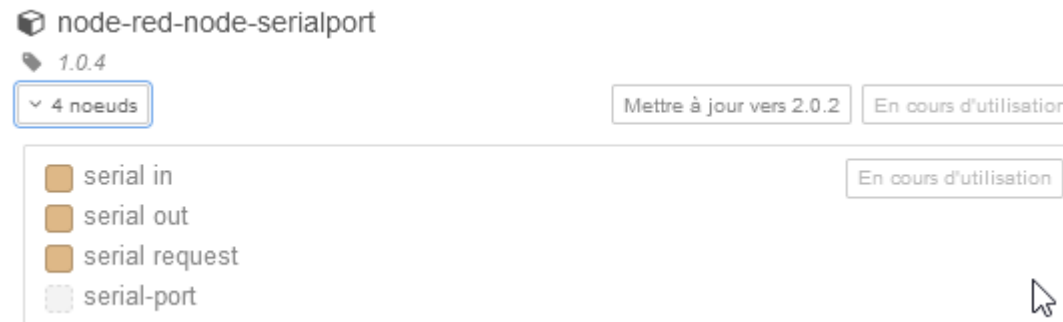


Certains nœuds font partie de bibliothèques à installer via la palette.



Pour nous :

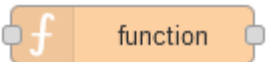
- **node-red-dashboard**
- **node-red-node-serialport**
- **node-red-node-ui-table**
- **node-red-contrib-ui-led**
- **node-red-contrib-boolean-logic-ultimate**



# Nœuds utilisés



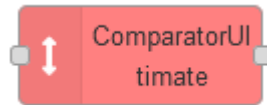
Lien série IN ou OUT



Fonction codée en js

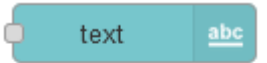


Filtre

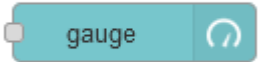


Comparaison (valeur > seuil)

Affichage sur dashboard



Texte



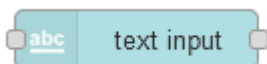
Jauge



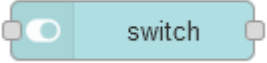
Graphique



Glissière (input)



Texte (input)



Interrupteur

Mode: bloquer sauf si la valeur change

Propriété: msg. payload

Appliquer le mode séparément pour chaque

msg. topic

Nom: Nom

Propriétés

Comparator Youtube Sample

Name: Above threshold ?

Input: msg. payload

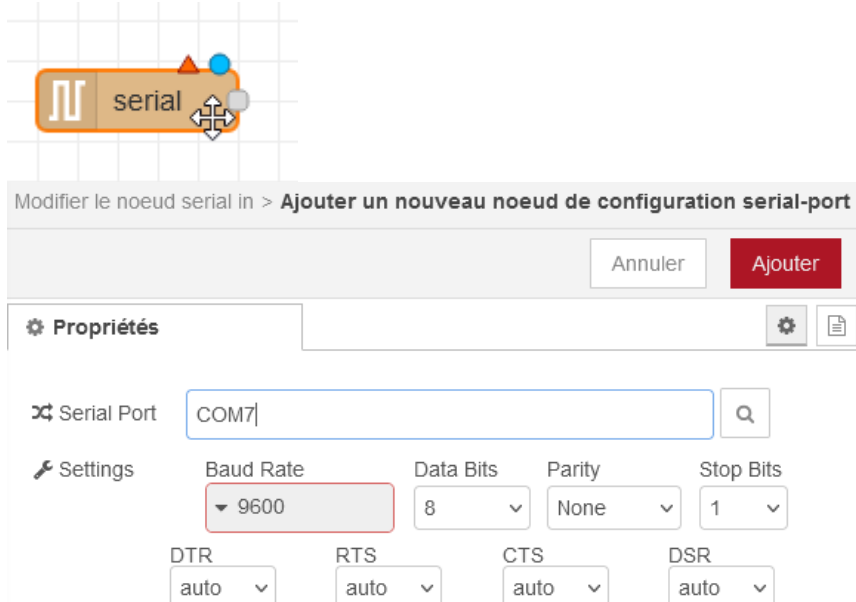
Topic 1: Measurement

Operation: >

Topic 2: Threshold

# Lecture de l'arduino et affichage

## Configuration sur NodeRed du lien série OUT

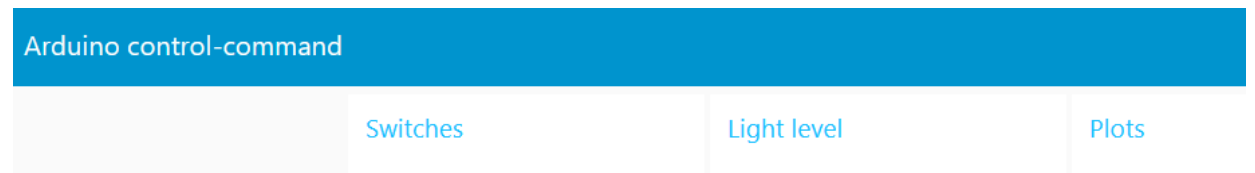
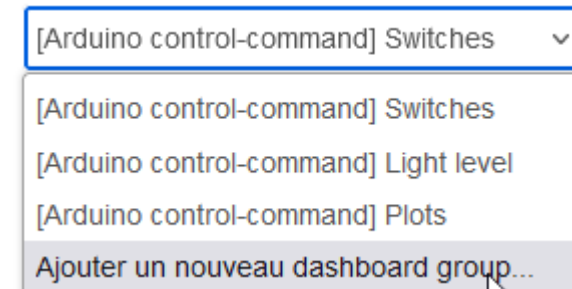


Sélectionner le bon serial port et configurer le lien.

Chaque élément de la dashboard doit être mis dans un groupe et un onglet

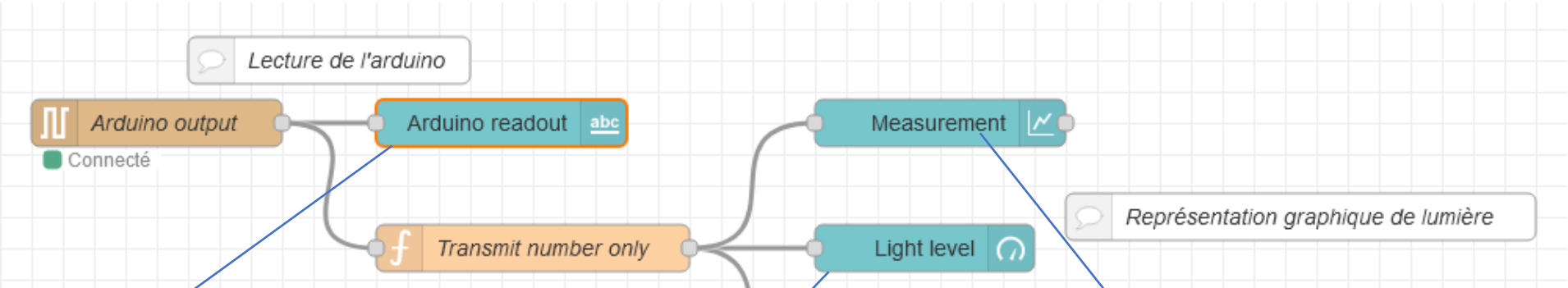


Il faut les créer si ils n'existent pas. Ils permettront l'affichage regroupé sur l'interface



# Affichage niveau de lumière

Chaque nœud doit être configuré et placé dans un groupe/onglet



**Modifier le noeud text**

Supprimer Annuler Terminer

**Propriétés**

Group: [Arduino control-command] Switches

Size: auto

Label: Arduino readout

Value format: {{msg.payload}}

Layout: label value labelvalue labelvalue

Class: Optional CSS class name(s) for widget

Name:

**Modifier le noeud gauge**

Supprimer Annuler Terminer

**Propriétés**

Group: [Arduino control-command] Light level

Size: 5 x 2

Type: Gauge

Label: Light level

Value format: {{value}}

Units: units

Range: min 0 max 800

Colour gradient:

**Modifier le noeud chart**

Supprimer Annuler Terminer

**Propriétés**

Group: [Arduino control-command] Plots

Size: 10 x 4

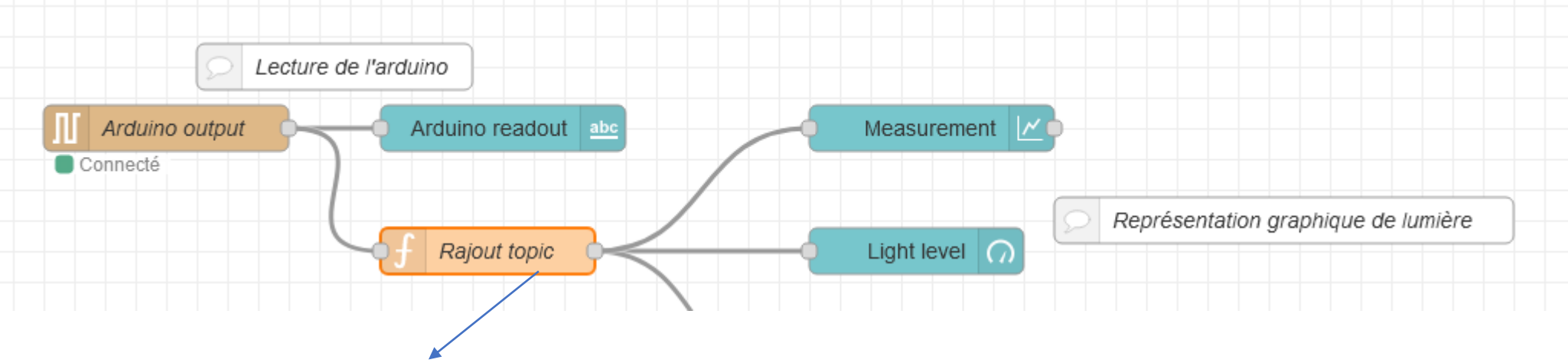
Label: Measurement

Type: Line chart  enlarge points

X-axis: last 5 minutes OR 1000 points

# Affichage niveau de lumière

Rajout d'un topic à un msg



**Modifier le noeud fonction**

Supprimer Annuler Terminer

**Propriétés**

Nom: Rajout topic

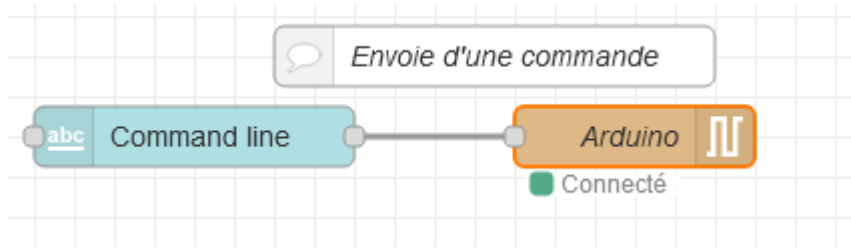
Configurations Au démarrage **Message reçu** À l'arrêt

```
1 msg.topic="Measurement";
2 return msg;
3
```

# Envoie d'une commande à l'arduino

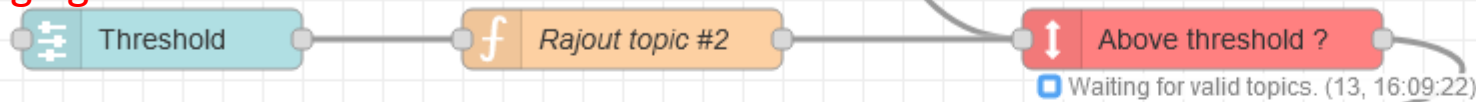
Lien avec le ON/OFF codé dans l'arduino. Plusieurs façons :

- Simple texte en input



- Détection d'un niveau de lumière supérieur à un seuil réglé par l'opérateur

## Réglage du seuil



## Filtrage si valeur ne change pas

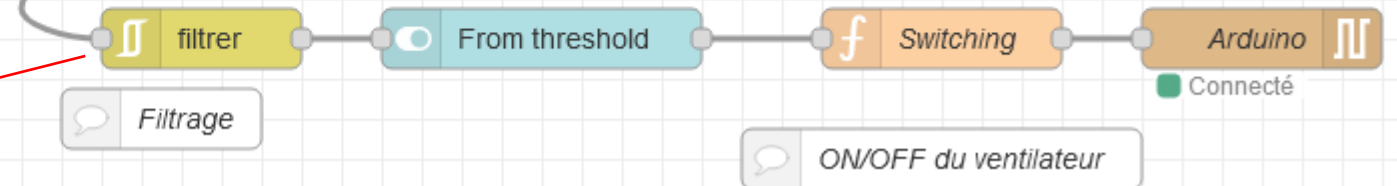
Mode: bloquer sauf si la valeur change

Propriété: msg. payload

Appliquer le mode séparément pour chaque

msg. topic

Nom: Nom



## Utilisation du topic pour comparer 2 payloads

Input: msg. payload

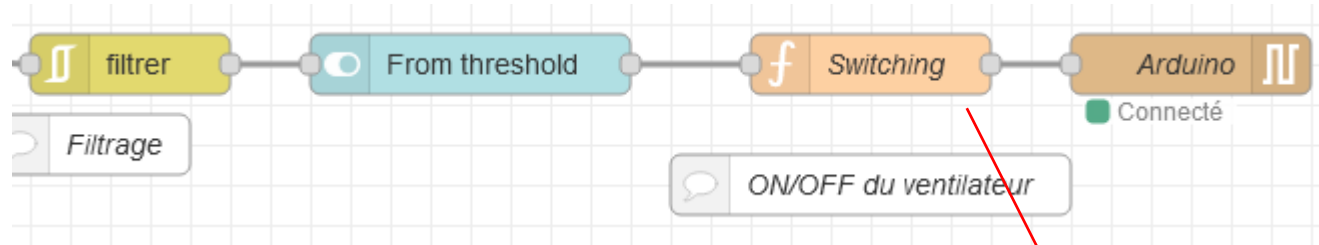
Topic 1: Measurement

Operation: >

Topic 2: Threshold

# Envoi de la commande à l'arduino

Commande ON/OFF sur la base du résultat du déclencheur



```
var Etat = msg.payload ;  
if (Etat === true )  
{  
  msg.payload= "ON";  
}  
else if (Etat === false )  
{  
  msg.payload= "OFF";  
}  
else  
{  
  msg.payload = "";  
}  
return msg;
```



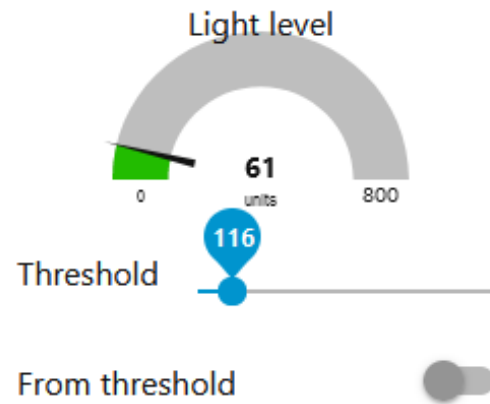
# Dashboard

## Switches

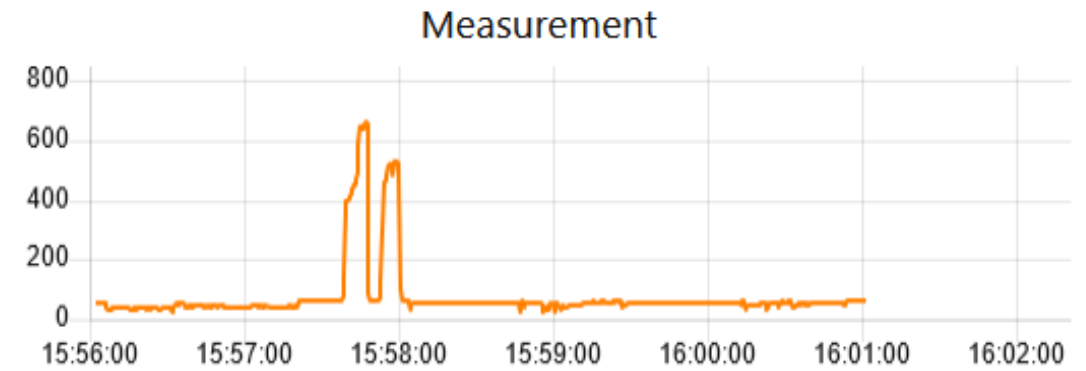
Command line  
OFF

Arduino readout **61.00**

## Light level



## Plots

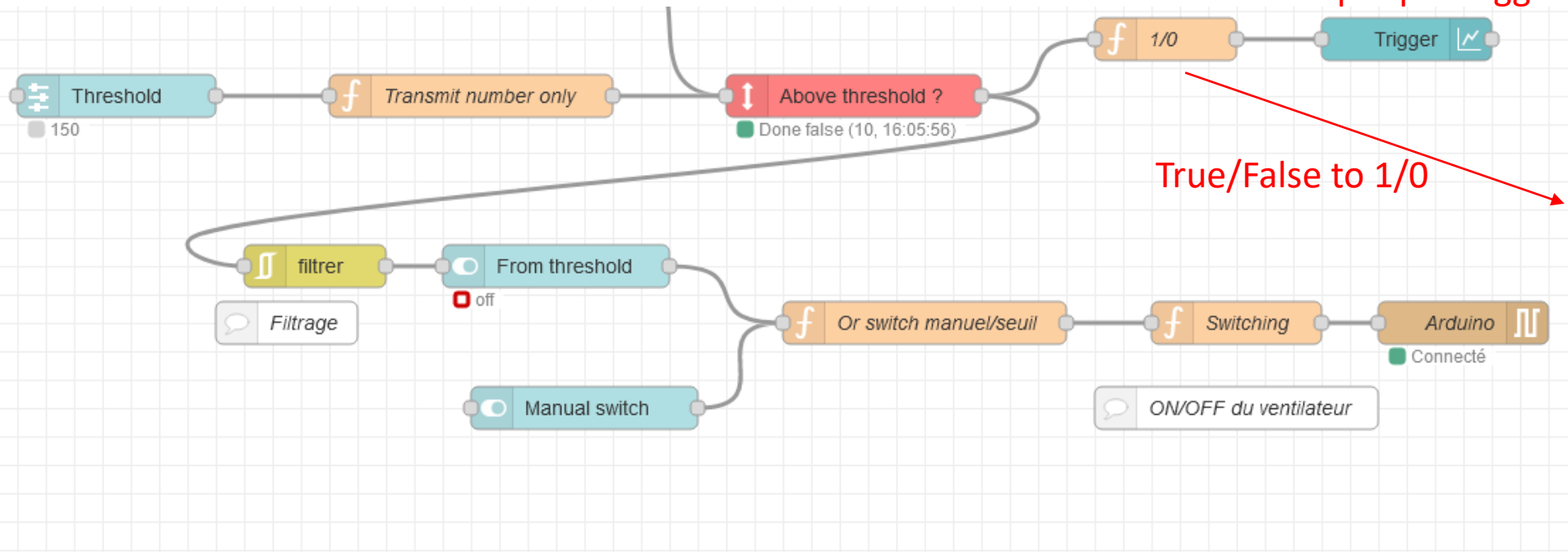




- Faire le lvl 2 ?
- Est-ce qu'il n'y aurait pas plus simple pour faire le OR ?
- Comment faire pour avoir une LED allumée sur le dashboard quand le seuil est dépassé ?
- Avez-vous trouvé comment faire un graphique plus grand que 5 de côté ?

# LVL 2

- Graphique déclencheur



Graphique trigger

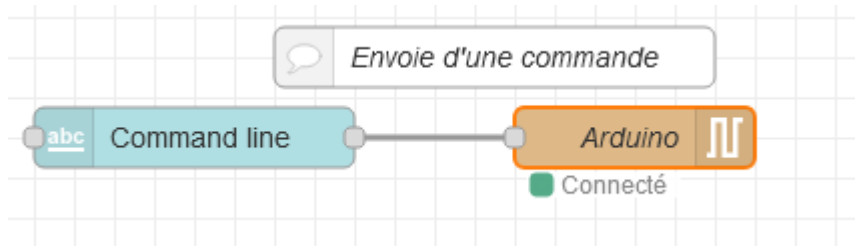
True/False to 1/0

```
var Etat = msg.payload ;
if (Etat === true )
{
  msg.payload= 1;
}
else if (Etat === false )
{
  msg.payload= 0;
}
else
{
  msg.payload = "";
}
return msg;
```

# LVL 2

Lien avec le ON/OFF codé dans l'arduino. Plusieurs façons :

- Simple texte en input



## Filtrage si valeur ne change pas

Mode: bloquer sauf si la valeur change

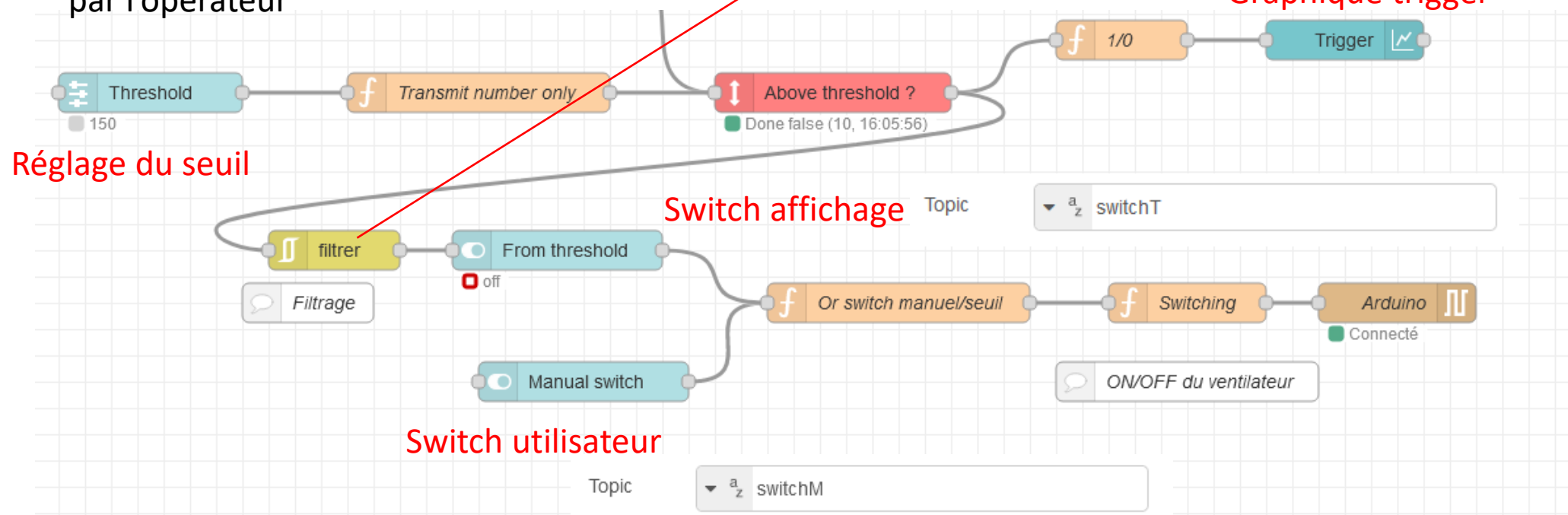
Propriété: msg. payload

Appliquer le mode séparément pour chaque

msg. topic

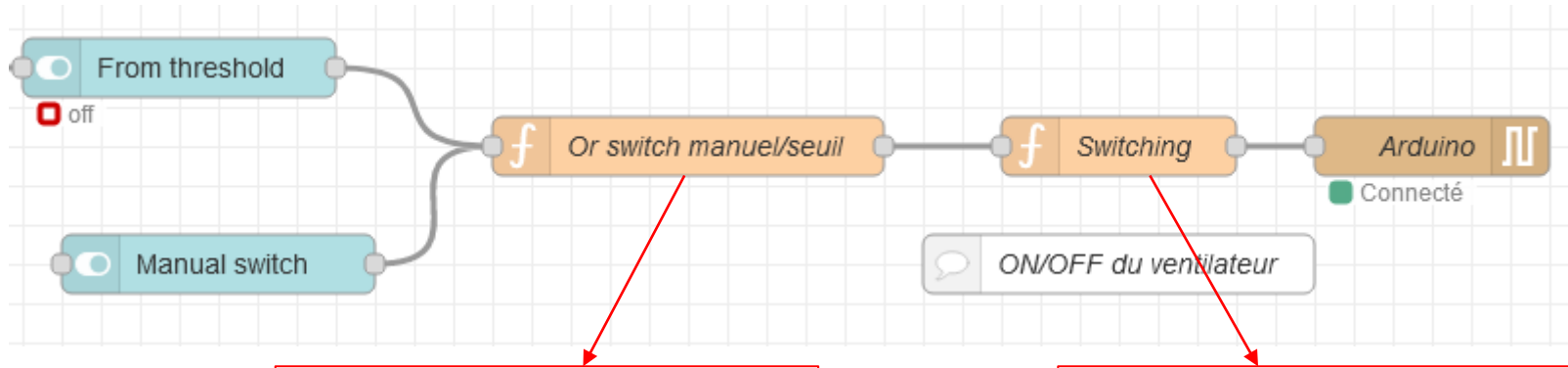
Nom: Nom

- Enclenchement d'un interrupteur (manual switch) ou détection d'un niveau de lumière supérieur à un seuil réglé par l'opérateur



# LVL 2

Commande ON/OFF sur la base des switchs



```
var s1=false;  
var s2= false;  
var topic = msg.topic;  
if (topic=="switchT")  
{  
  s1 = msg.payload;  
}  
if (topic=="switchM")  
{  
  s2 = msg.payload;  
}  
  
msg.payload=s1 || s2;  
  
return msg;
```

```
var Etat = msg.payload ;  
if (Etat === true )  
{  
  msg.payload= "ON";  
}  
else if (Etat === false )  
{  
  msg.payload= "OFF";  
}  
else  
{  
  msg.payload = "";  
}  
return msg;
```

# Dashboard

## Arduino control-command

### Switches

Command line

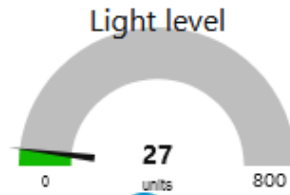
Manual switch



Arduino readout

**27.00**

### Light level



Threshold

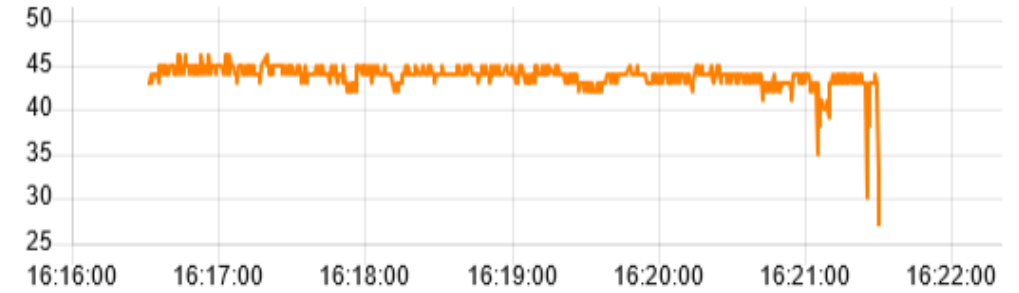


From threshold

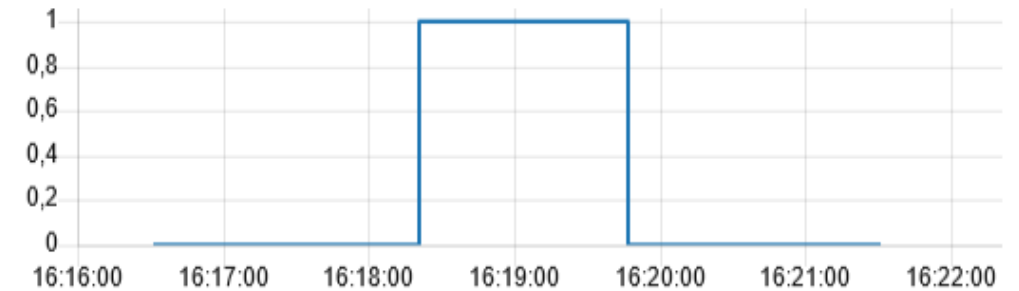


### Plots

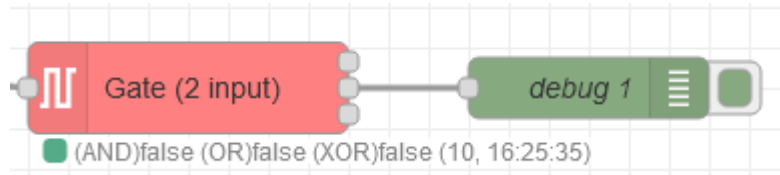
Measurement



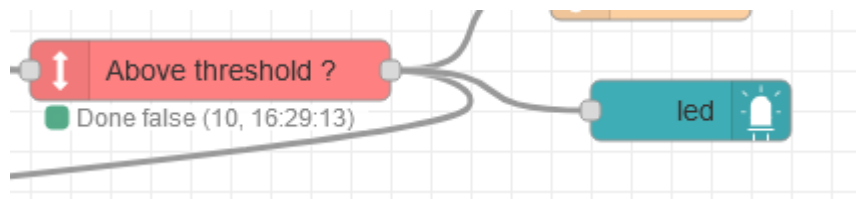
Trigger



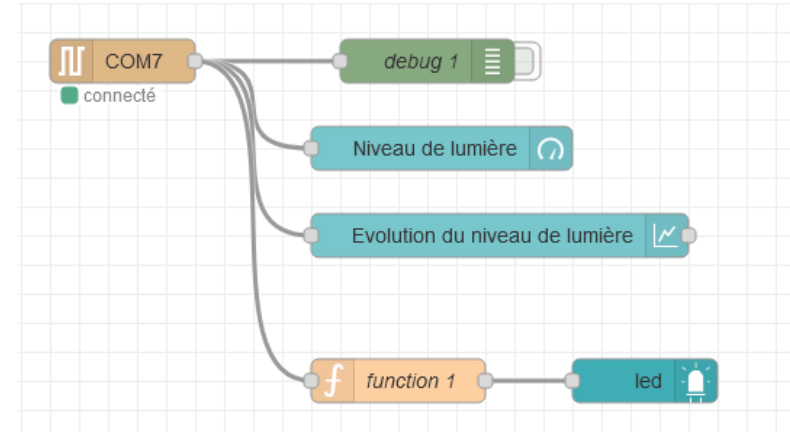
## Utilisation de la gate 2 inputs



## LED



## Rajout d'une fonction pour détecter un niveau trop élevé



## Taille

Modifier le noeud chart > **Modifier le noeud dashboard group**

Supprimer Annuler Sauver

**Propriétés**

Name Plots

Tab Arduino control-command

Class Optional CSS class name(s) for widget

Width 15

Display group name

Allow group to be collapsed

Modifier le noeud fonction

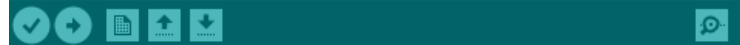
Supprimer Annuler Terminer

**Propriétés**

Nom fonction 1

Configurations Au démarrage Message reçu À l'arrêt

```
1 var luz = msg.payload;
2 if (luz > 300)
3 {
4     msg.payload = true;
5 }
6 else
7 {
8     msg.payload = false;
9 }
10 return msg;
```



NodeRedding

```
const int led = 13;
int PinLuz=A2;
float ValLuz = 0;

void setup() {
  Serial.begin(9600);
  pinMode(led, OUTPUT);
}

void loop() {
  if (Serial.available()) {
    String command = Serial.readString();
    if (command == "ON") {
      digitalWrite(led, HIGH);
    }
    else if (command == "OFF") {
      digitalWrite(led, LOW);
    }
    else if (command != ""){
      Serial.println(command);
    }
  }
  ValLuz = analogRead(PinLuz);
  Serial.println(ValLuz);
  delay(400);
}
```



Quelle fonctionnalité  
a-t-on rajouté ?



NodeRedding

```
const int led = 13;
const int switchIn = 8;
bool mySwitch = true;
int PinLuz=A2;
float ValLuz = 0;

void setup() {
  Serial.begin(9600);
  pinMode(led, OUTPUT);
  pinMode(switchIn, INPUT);
}

void loop() {
  if (Serial.available()) {
    String command = Serial.readString();
    if (command == "ON") {
      digitalWrite(led, HIGH);
    }
    else if (command == "OFF") {
      digitalWrite(led, LOW);
    }
    else if (command != ""){
      Serial.println(command);
    }
  }

  mySwitch = digitalRead(switchIn);
  Serial.println(mySwitch);

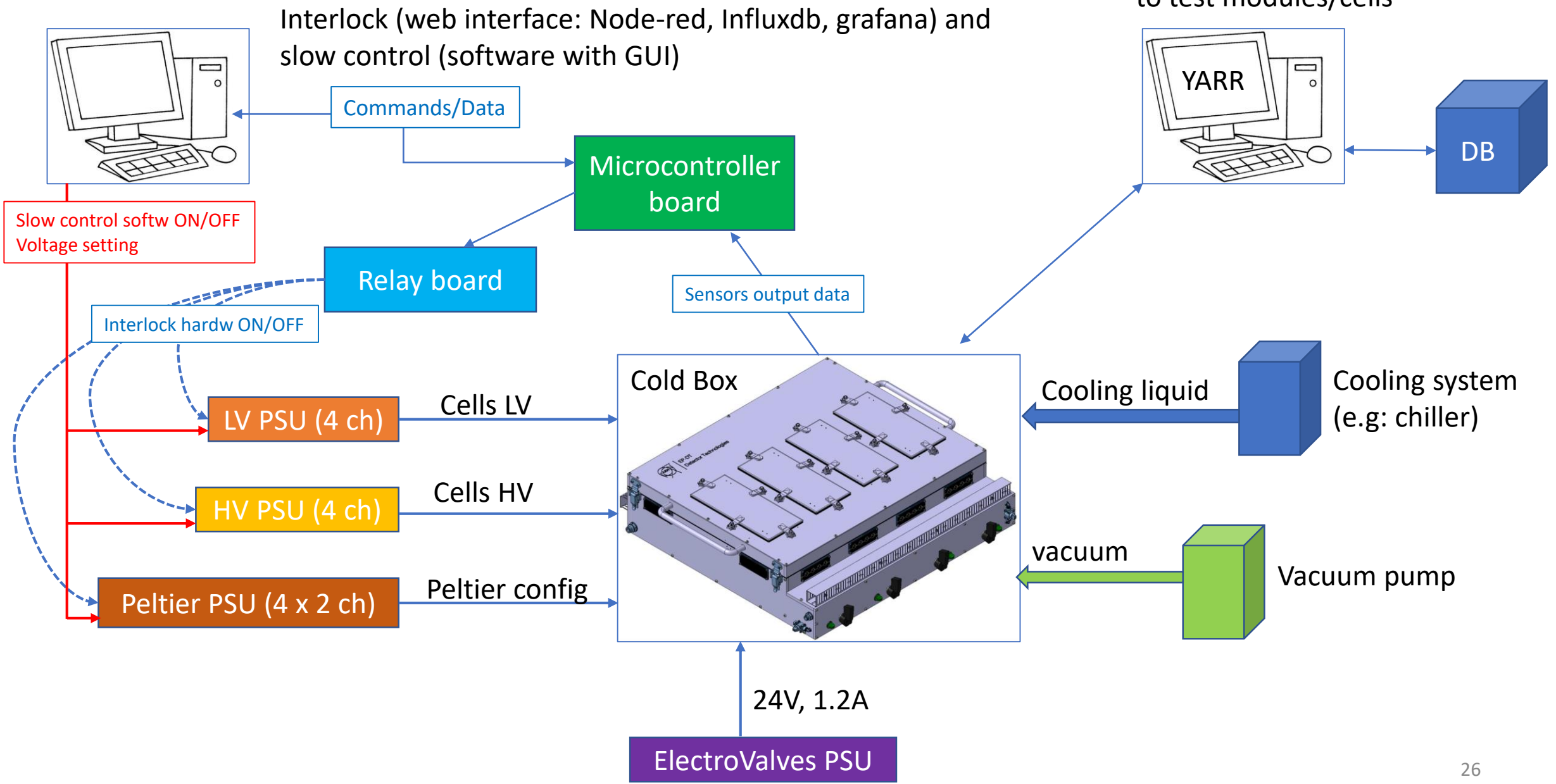
  if (mySwitch == false) {
    digitalWrite(led, HIGH);
  }
  else{
    digitalWrite(led, LOW);
  }

  ValLuz = analogRead(PinLuz);
  Serial.println(ValLuz);
  delay(400);
}
```



# EXEMPLE D'APPLICATION AU CPPM

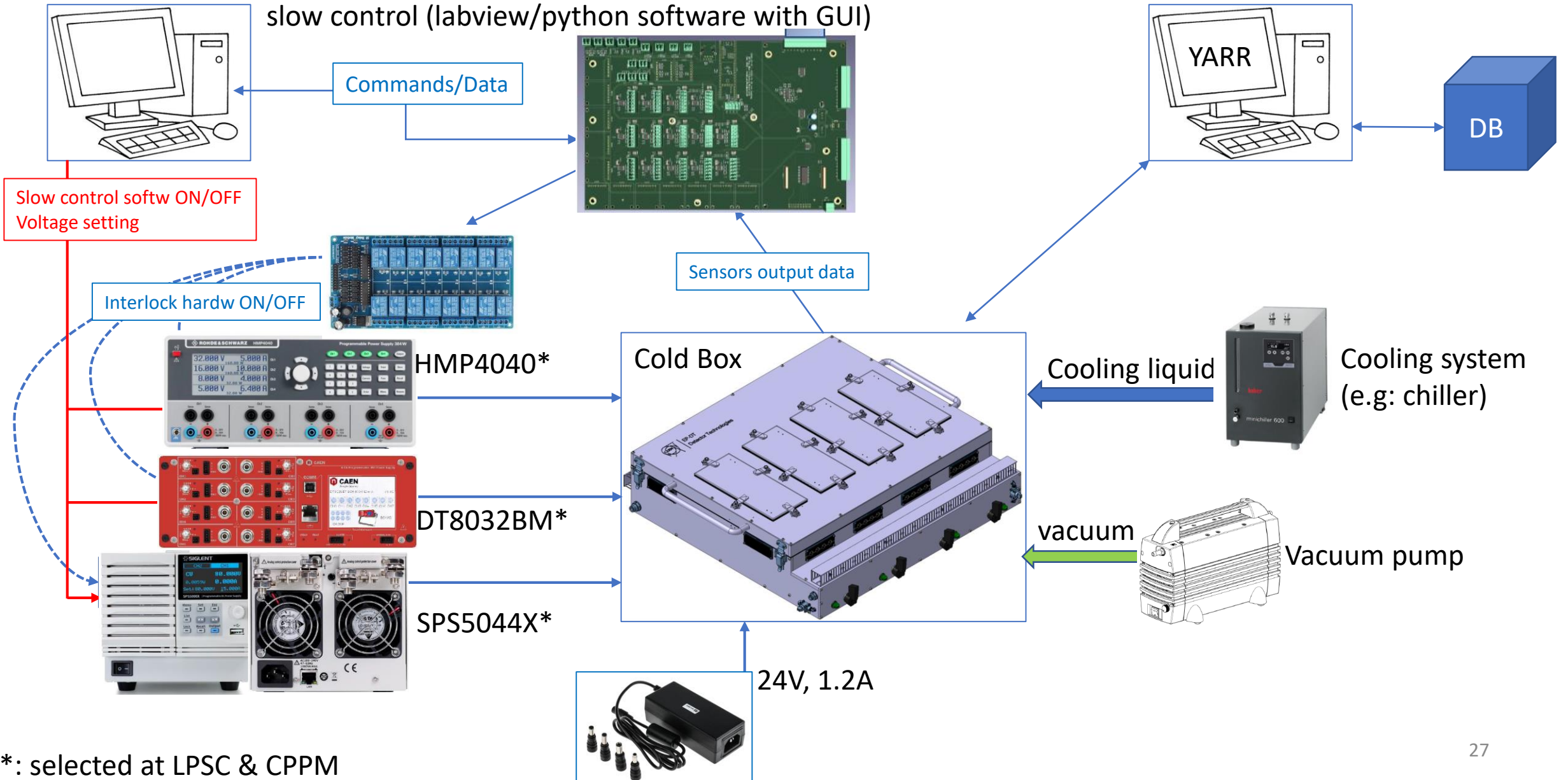
# Système de test Modules/Cells loaded



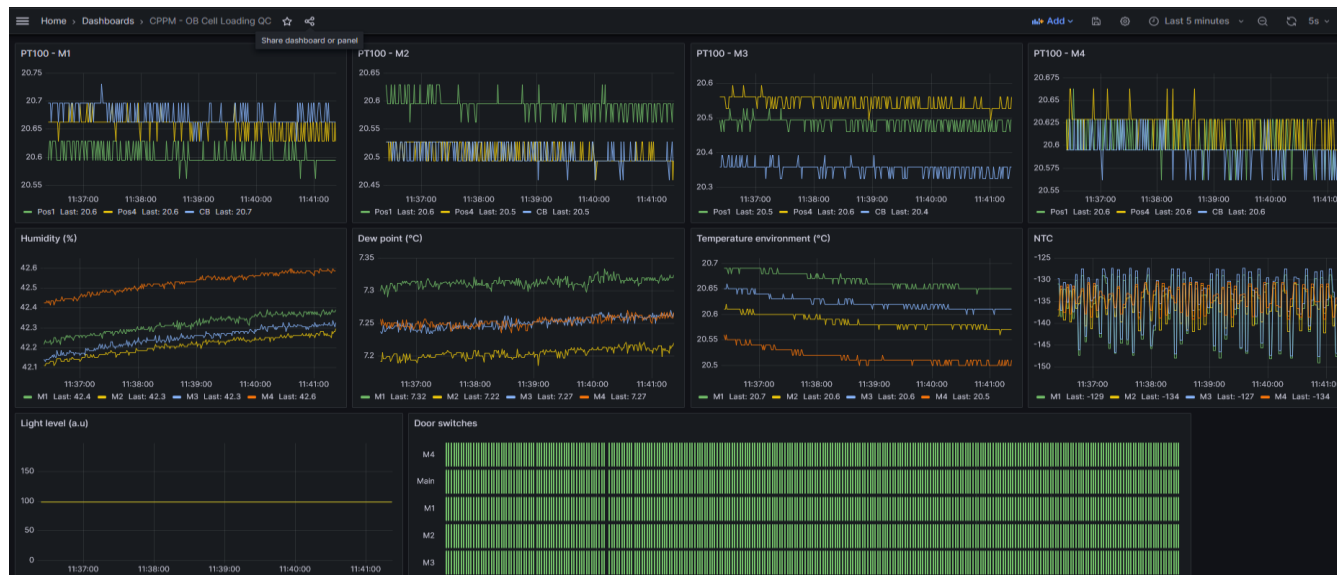
# Système de test Modules/Cells loaded

Interlock (web interface: Node-red, Influxdb, grafana) and slow control (labview/python software with GUI)

YARR PC running scripts to test modules/cells



# Systeme de test Modules/Cells loaded

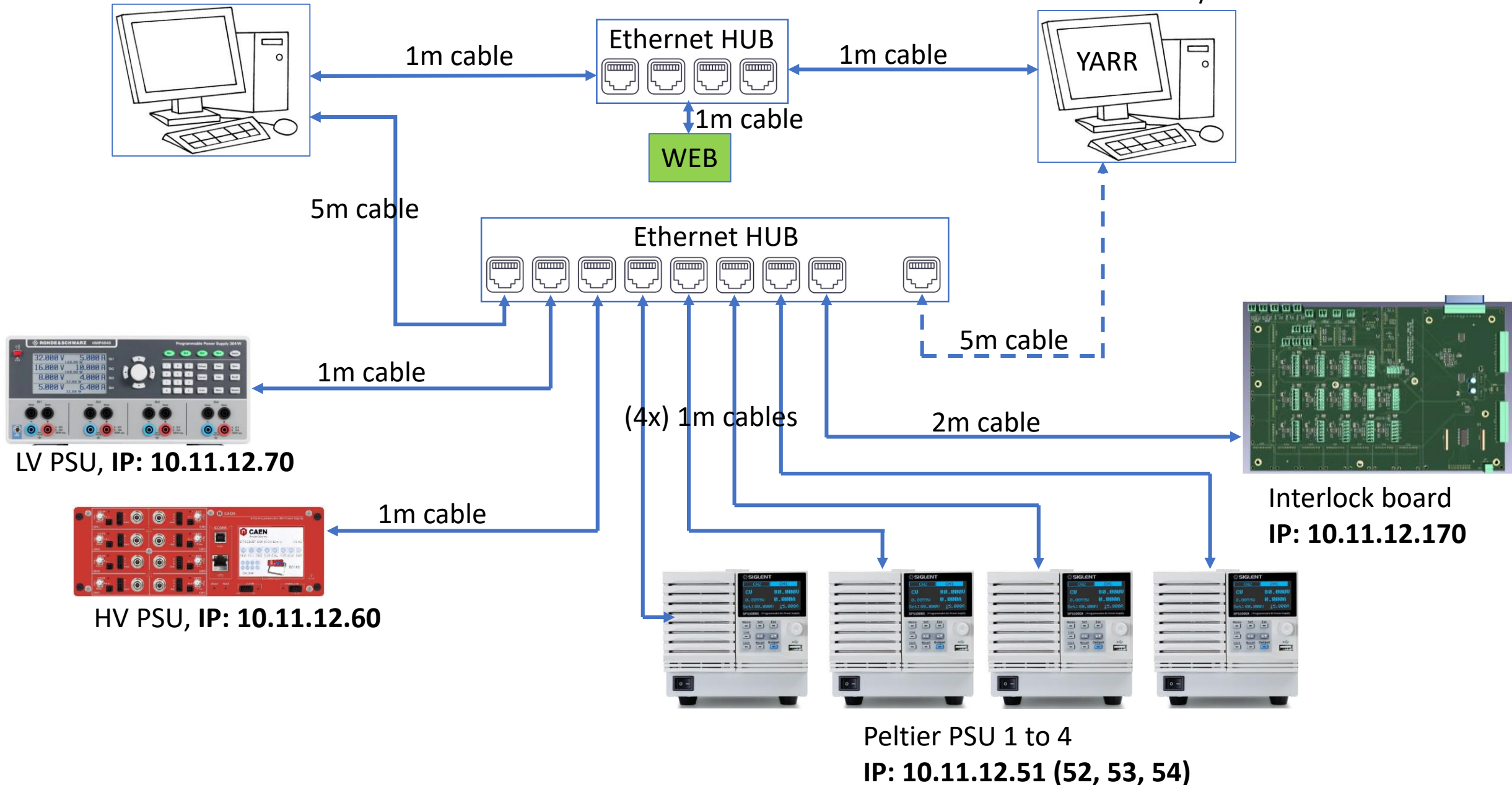


# IP addressing of loaded cells test set-up

Interlock & slow control PC

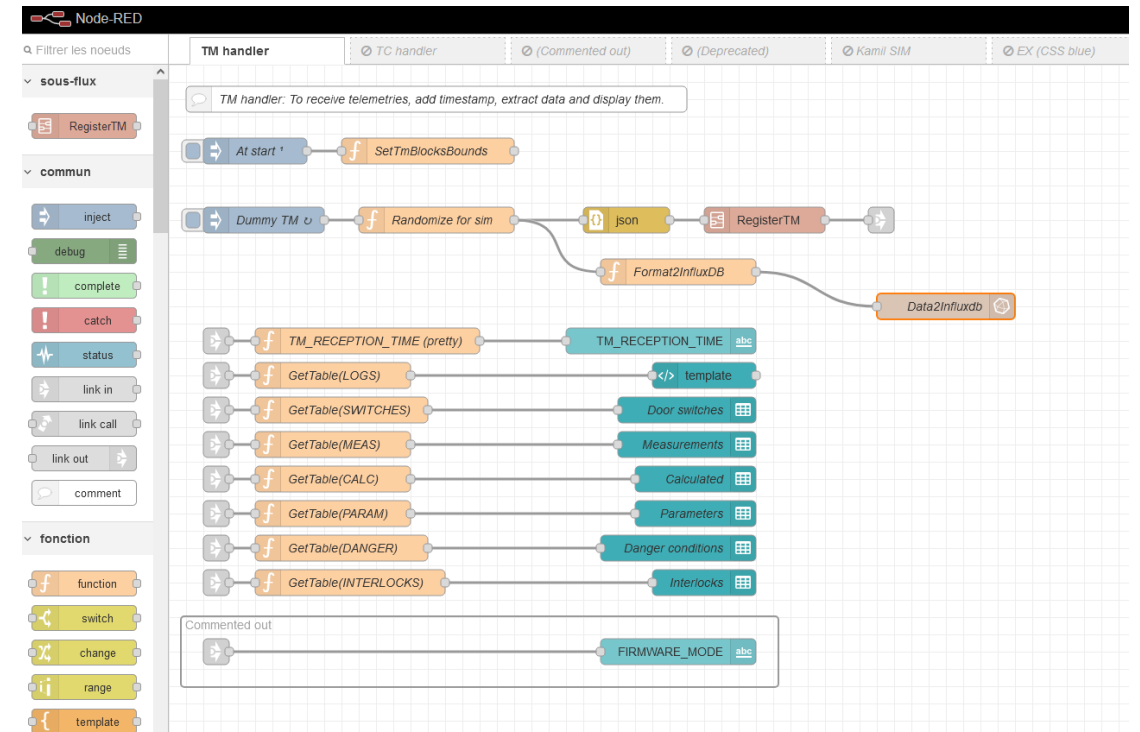
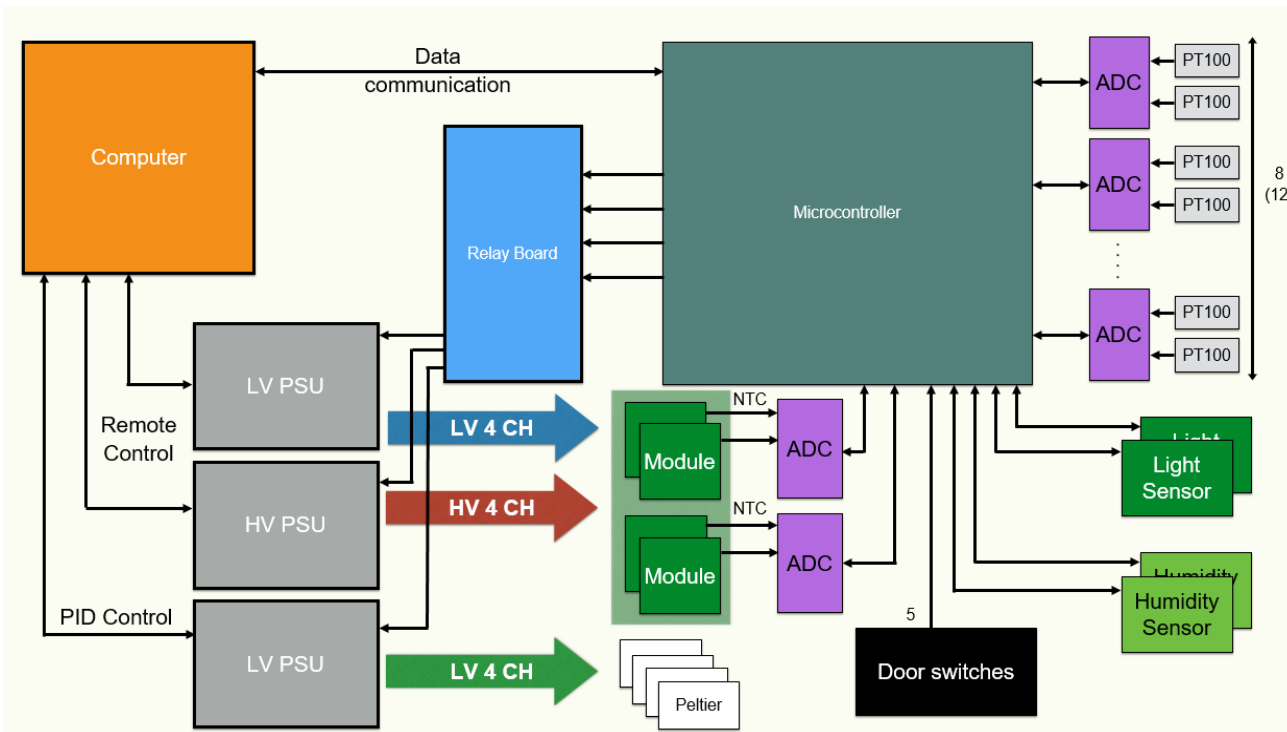
IP: 10.11.12.13 – Mask: 255.255.255.0

YARR PC running scripts to test modules/cells



# Pour itk

- Le code gère les données et logs fournis par le microcontrôleur
- Il intègre leur sauvegarde sur Influxdb



BACK-UP

## Partie interlock

| PT100                | Va...        | NTC / Light      | Va...        | TM_RECEPTION_TIME   | RESET LOCKED-IN INTERLOCKS |                             |               |                           |               |
|----------------------|--------------|------------------|--------------|---|----------------------------|-----------------------------|---------------|---------------------------|---------------|
| M1_PT100_11_T        | 20.427       | M1_NTC_T         | -136.6...    | 05/02/2024 16:41:42<br>f 1707147702007 1  |                            |                             |               |                           |               |
| M1_PT100_14_T        | 20.46        | M2_NTC_T         | -136.0...    | <b>Dangers (actual)</b>   | <b>Statds</b>              | <b>Interlocks (lock-in)</b> | <b>Statds</b> | <b>Controlled devices</b> | <b>Statds</b> |
| M2_PT100_11_T        | 20.46        | M3_NTC_T         | -134.3...    | 1_LIGHT_DETECTED  |                            | 1_LIGHT_DETECTED            |               | M1_PEL_OFF                |               |
| M2_PT100_14_T        | 20.359       | M4_NTC_T         | -133.82      | 2_DOOR_OPENED   |                            | 2_DOOR_OPENED               |               | M2_PEL_OFF                |               |
| M3_PT100_11_T        | 20.359       | M12_LIGHT_ADC    | 0            | 3_M1_PT100_TOO_HOT  |                            | 3_M1_PT100_TOO_HOT          |               | M3_PEL_OFF                |               |
| M3_PT100_14_T        | 20.393       | M34_LIGHT_ADC    | 0            | 3_M2_PT100_TOO_HOT  |                            | 3_M2_PT100_TOO_HOT          |               | M4_PEL_OFF                |               |
| M4_PT100_11_T        | 20.46        |                  |              | 3_M3_PT100_TOO_HOT  |                            | 3_M3_PT100_TOO_HOT          |               | M1_HV_OFF                 |               |
| M4_PT100_14_T        | 20.46        |                  |              | 3_M4_PT100_TOO_HOT  |                            | 3_M4_PT100_TOO_HOT          |               | M2_HV_OFF                 |               |
|                      |              |                  |              | 4_M1_NTC_SUPER_HOT  |                            | 4_M1_NTC_SUPER_HOT          |               | M3_HV_OFF                 |               |
| <b>Ambient</b>       | <b>Va...</b> | <b>Dewpoints</b> | <b>Va...</b> | 4_M2_NTC_SUPER_HOT  |                            | 4_M2_NTC_SUPER_HOT          |               | M4_HV_OFF                 |               |
| M1_ENV_T             | 20.44        | M1_ENV_DEWT      | 9.358        | 4_M3_NTC_SUPER_HOT  |                            | 4_M3_NTC_SUPER_HOT          |               | M1_LV_OFF                 |               |
| M1_ENV_RH            | 49.293       | M2_ENV_DEWT      | 9.277        | 4_M4_NTC_SUPER_HOT  |                            | 4_M4_NTC_SUPER_HOT          |               | M2_LV_OFF                 |               |
| M2_ENV_T             | 20.4         | M3_ENV_DEWT      | 9.294        | 5_DEWPOINT_REACHED  |                            | 5_DEWPOINT_REACHED          |               | M3_LV_OFF                 |               |
| M2_ENV_RH            | 49.145       | M4_ENV_DEWT      | 9.213        | 6_LOW_BOARD_VOLTAGE   |                            | 6_LOW_BOARD_VOLTAGE         |               | M4_LV_OFF                 |               |
| M3_ENV_T             | 20.47        |                  |              | 7_WATCHDOG_RESTART  |                            | 7_WATCHDOG_RESTART          |               |                           |               |
| M3_ENV_RH            | 48.979       |                  |              | 8_PC_COM_LOST   |                            | 8_PC_COM_LOST               |               |                           |               |
| M4_ENV_T             | 20.36        |                  |              | 9_NOT_ENOUGH_DRY_AIR  |                            | 9_NOT_ENOUGH_DRY_AIR        |               |                           |               |
| M4_ENV_RH            | 49.053       |                  |              | 10_NOT_ENOUGH_VACUUM  |                            | 10_NOT_ENOUGH_VACUUM        |               |                           |               |
| <b>Door switches</b> | <b>St...</b> |                  |              | <b>Teensy Notifications (Latest first)</b>  |                            |                             |               |                           |               |
| DSWITCH_1_OPEN       | Closed       |                  |              | <b>Time</b>   |                            |                             |               |                           |               |
| DSWITCH_2_OPEN       | Closed       |                  |              | <b>Message</b>  |                            |                             |               |                           |               |
| DSWITCH_3_OPEN       | Closed       |                  |              | 05/02/2024 16:39:28 Legal TC received. Resetting all interlocks.                                  |                            |                             |               |                           |               |
| DSWITCH_4_OPEN       | Closed       |                  |              | 05/02/2024 16:38:32 Starting. Setup complete. Now protecting. Custom parameters reset to default. |                            |                             |               |                           |               |
| DSWITCH_5_OPEN       | Closed       |                  |              | 05/02/2024 16:38:32 Starting. Setup complete. Now protecting. Custom parameters reset to default. |                            |                             |               |                           |               |
|                      |              |                  |              | 05/02/2024 16:38:32 Starting. Setup complete. Now protecting. Custom parameters reset to default. |                            |                             |               |                           |               |

## Partie slow control Pilotage des 3 types d'alimentation

The interface displays four modules (Module 1 to Module 4) and a common section. Each module has a 'SOM OK' indicator, 'FORCE LISTEN', 'POWER ON', and 'POWER OFF' buttons. Below these are 'Enable channel 1 (TOP)' and 'Enable channel 2 (BOTTOM)' toggle switches. The 'Temp. regulation' section for each module includes:
 

- Dew point: 6.8 °C (Module), 25.0 °C (Coolblock), 20.7 °C (Coolblock)
- PID regulation enable: toggle switch
- Use the Coolblock (PT100) like T° reference: dropdown menu
- T° Target: slider set to 20
- Use 12°C above dew point like set point: toggle switch
- PT100 (cooling block temp.): 0 °C

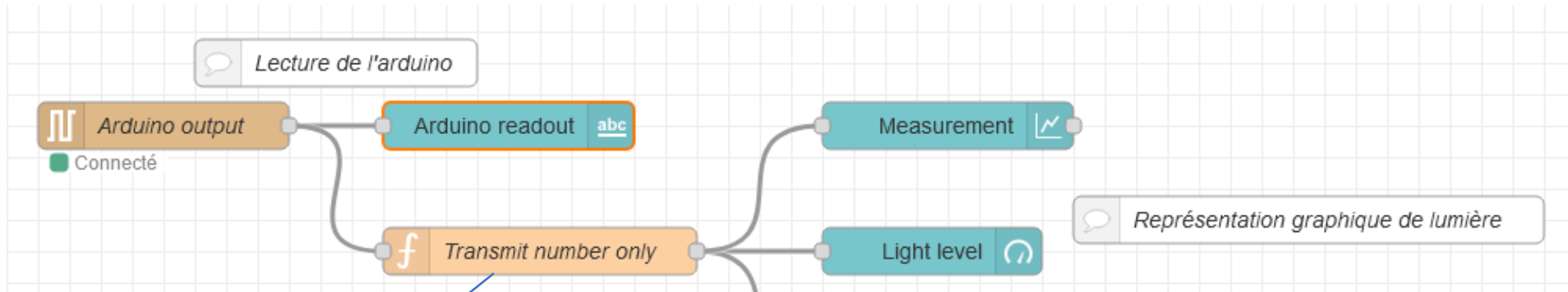
 The common section includes:
 

- Common: 'FORCE LISTEN' and 'CLEAR ALARM' buttons, 'Ramp Up' and 'Ramp Down' sliders, and 'Ramp on Power Down' toggle.
- Module 1 (PSU channel 4): On/Off status, FP Interlocked, Ramping Up/Down, KILL (qcxlock), FP Disabled, Local Ctrl, Over cur. Voltage Measured (0 V), Current Measured (101 µA).
- Module 2 (PSU channel 5): Similar status and measurements.
- Module 3 (PSU channel 6): Similar status and measurements.
- Module 4 (PSU channel 7): Similar status and measurements.



# Affichage niveau de lumière

Supprimer les NaN via une fonction en javascript



Modifier le noeud fonction

Supprimer Annuler Terminer

Propriétés

Nom Transmit number only

Configurations Au démarrage Message reçu À l'arrêt

```
1 var output = parseFloat(msg.payload) ;
2 if (!Number.isNaN(output))
3 {
4     msg.payload= output;
5     msg.topic="Measurement"
6 }
7 else
8 {
9     msg.payload = "";
10 }
11 return msg;
12
```

```
var output = parseFloat(msg.payload) ;
if (!Number.isNaN(output))
{
    msg.payload= output;
    msg.topic="Measurement"
}
else
{
    msg.payload = "";
}
return msg;
```