ESP32 - BMP180

Objectif(s)

Créer un objet connecté en wifi qui enverra via **MQTT** la température et la pression atmosphérique vers un serveur qui stockera ces informations en base de données **InfluxDB** avant de les restituer au travers de la solution **Grafana**.

> Répartissez vous bien les tâches : le travail est divisible et il y a pas mal de choses à faire ...

Arduino

Ajout de l'ESP 32 à Arduino si ce n'est pas déjà fait : https://doc.creatronic.fr/books/i2---conception-dun-syst%C3%A8me-embarqu%C3%A9-temps-r%C3%A9el/page/installation-arduino

> Utiliser la board **Wemos LOLIN32**.

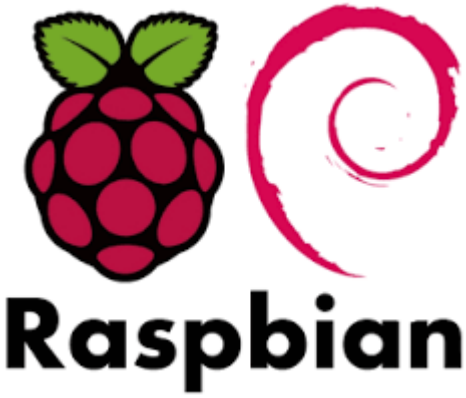
Librairie conseillé(e) : Adafruit_BMP085

Pour l'installer il faudra ajouter l'URL suivante dans le menu préférences :

```
https://adafruit.github.io/arduino-board-index/package_adafruit_index.json
```

Rpi

Préparation du Raspberry Pi (installation Raspian)



Si Raspbian est déjà installé vous pouvez sauté cette étape.

Télécharger la dernière version de Raspbian : https://www.raspberrypi.org/downloads/raspbian/

Flasher l'image sur la carte SD avec Balena Etcher : https://www.balena.io/etcher/

> Avant de débrancher la carte SD, sur la partition boot ajouter un fichier **ssh** (pas un .ssh, un fichier ssh tout court **sans extension**) afin que le système autorise le ssh bloqué par défaut.

Si vous souhaitez travailler en Wifi, il faut le configurer avant de mettre la carte SD dans le Raspberry : https://www.raspberrypi-spy.co.uk/2017/04/manually-setting-up-pi-wifi-using-wpa_supplicant-conf/

Mise à jour du système :

```
sudo apt update && sudo apt dist-upgrade && sudo apt upgrade
```

Installation de NodeRed



```
cd ~
bash <(curl -sL https://raw.githubusercontent.com/node-red/raspbian-deb-package/master/resources/update-nodejs-and-nodered)
sudo systemctl enable nodered.service
sudo node-red-start
```

Installation de InfluxDB



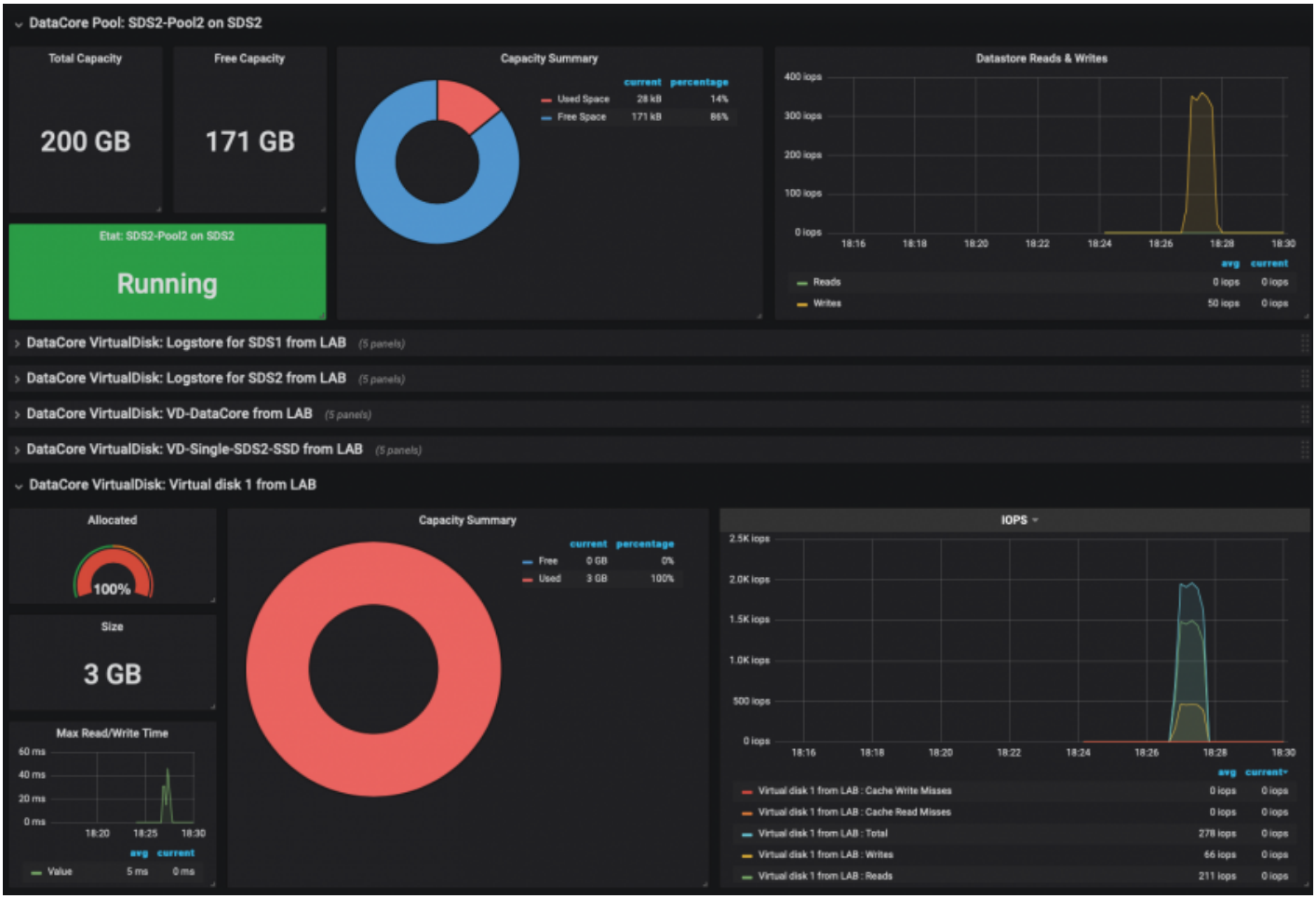

```
curl -sL https://repos.influxdata.com/influxdb.key | sudo apt-key add -
echo "deb https://repos.influxdata.com/debian buster stable" | sudo tee
/etc/apt/sources.list.d/influxdb.list
sudo apt-get update && sudo apt-get install influxdb
sudo service influxdb start
```

Si tout va bien, taper "influx" pour accéder à la console influxdb.

Créer une base de donnée : `create database lenomquevousvoulez;`

Installation de Grafana






```
sudo apt update && sudo apt-get install -y software-properties-common apt-transport-https
sudo nano /etc/apt/sources.list.d/grafana.list
```

Ajouter le dépôt suivant puis enregistrer.

```
deb https://packages.grafana.com/oss/deb stable main
```

Puis :

```
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
sudo apt-get update && sudo apt-get install grafana
sudo service grafana-server start
```

Doc officielle en cas de problème : https://grafana.com/docs/installation/debian/

Accessible sous http://ip:3000

Login/mdp par défaut : admin/admin

Ajout des nodes nécessaires dans node-red

Ajouter les nodes suivants :

node-red-contrib-influxdb => va nous permettre de dialoguer avec les bases influxDB

Travail à réaliser

Sur **nodeRED**, créer un flow permettant de récupérer et stocker les données de l'objet connecté et de l'insérer dans un *measurement* influxDB.

Sur Grafana : créer un nouveau dashboard qui permettra d'afficher un graph et une gauge de la charge CPU (penser à activer l'auto refresh).

Revision #5
Created 7 October 2020 14:59:59 by Admin
Updated 8 October 2020 12:12:54 by Admin