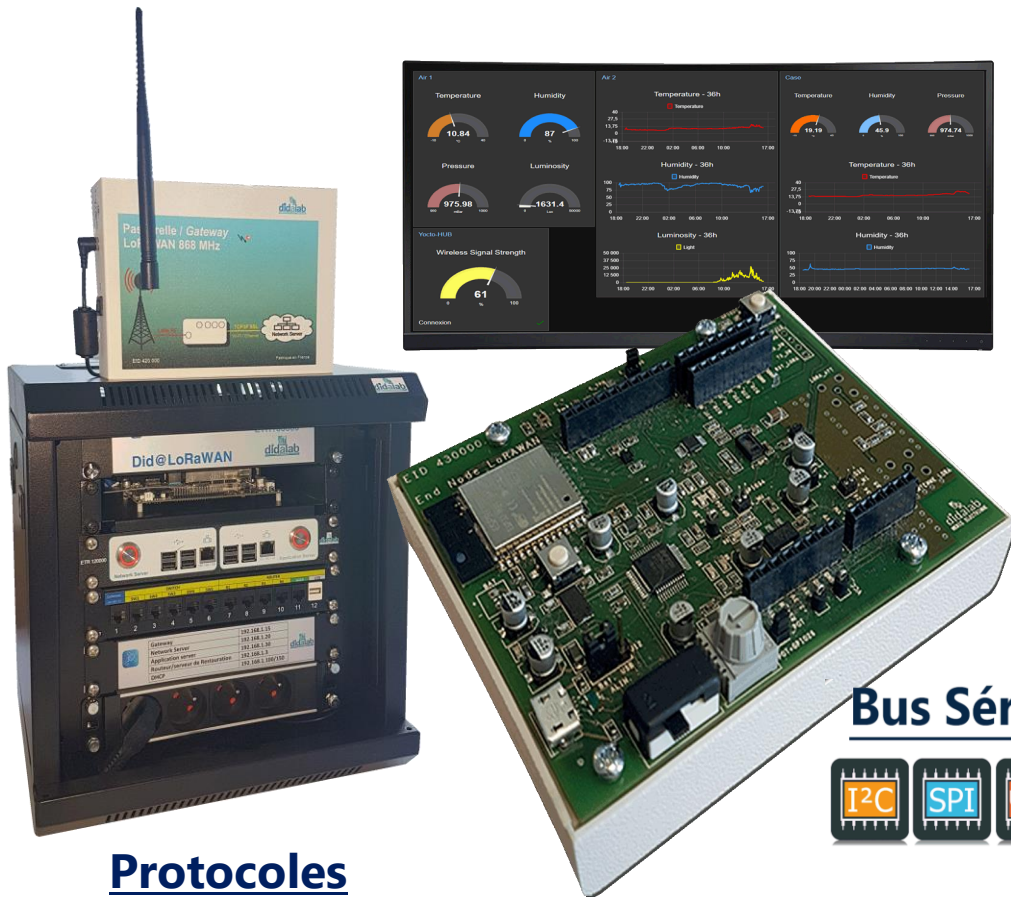


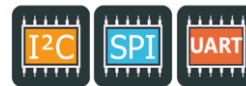
I.o.T. End Device



Programmations



Bus Série



Protocoles



Mise en oeuvre du système Did@LoRaWAN

3. *Outils*

- Créer un répertoire didalab sur le disque C :

- Installer Putty dans c:\didalab\Putty

Dans le raccourci qui se crée sur le PC, sauvegarder les liaisons sshnotée au Chapitre 1

- Installer pspad dans c:\didalab\Pspad

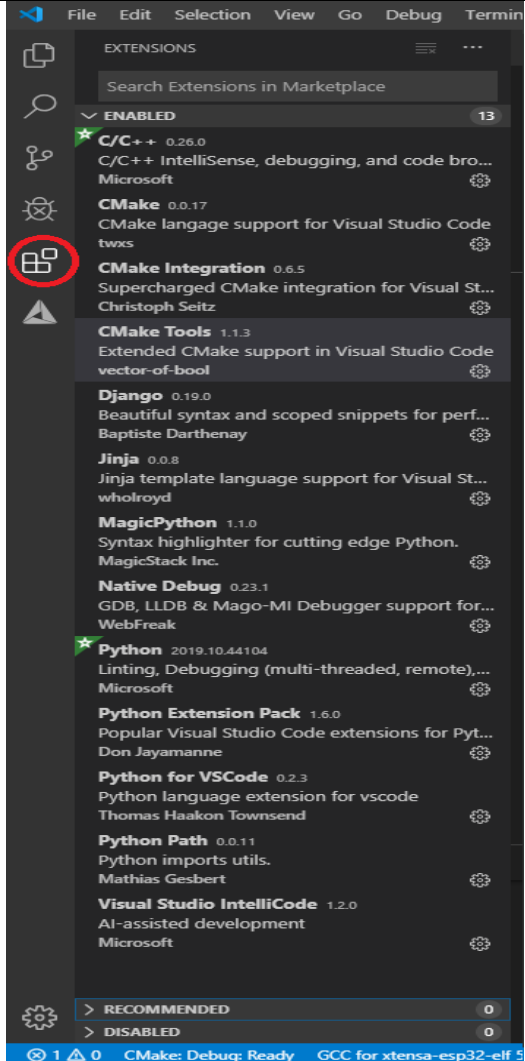
- Installer les drivers FTDI

Dans le sous répertoire DriverUSB, sélectionner le fichier CDM v2.12.00 WHQL Certified
Effectuer un clic droit, installer en tant qu'administrateur.

4. Visual Studio Code et Compilateur

- Installer Visual Studio Code dans c:\didalab\

Une fois l'installation finie, lancer VSCode, puis cliquer à gauche sur l'icône extensions
En haut, dans la barre de recherche, entrer les noms suivants et installer les plug-ins nécessaires.

<p>c/C++ 0.26.1 CMake 0.0.17 CMakeintegration 0.6.5 CMake Tools 1.2.2</p> <p>Django 0.19.0</p> <p>Jinja 0.0.8</p> <p>Native Debug 0.23.1</p> <p>MagicPython 1.1.10 Python 2019.10.44104 Python Extension Pack 1.6.0 Python for VSCode 0.2.3 Python Path 0.0.11</p> <p>Visual Studio IntelliCode 1.2.0</p> <p>Puis refermer VSCode</p>	
---	---

- Installation du Compilateur esp-idf-tools-setup-1.1
Lancer l'exécutable, et conserver les paramètres par défaut

5. Finalisation de l'environnement

- Importation des programmes

Copier le contenu du répertoire Programmes vers c:\didalab

- Variables d'environnement

Lancer un terminal **Administrateur**, se placer dans C:\didalab\esp-idf-master\esp_idf

Lancer le script setenv.bat (Retour attendu : 2 * Réussite + affichage des variables path)

En cas d'erreur sur le retour de la commande (limite 1024 caractères)

Vérifiez la présence des variables d'environnement suivantes :

Variable Utilisateur : Path

C:\Python27\

C:\Python27\Scripts

C:\Program Files\Esspressif\ESP-IDF Tools\toolchain\bin

C:\Program Files\Esspressif\ESP-IDF Tools\mconf-idf

C:\Program Files\Esspressif\ESP-IDF Tools

C:\didalab\Microsoft VS Code\bin

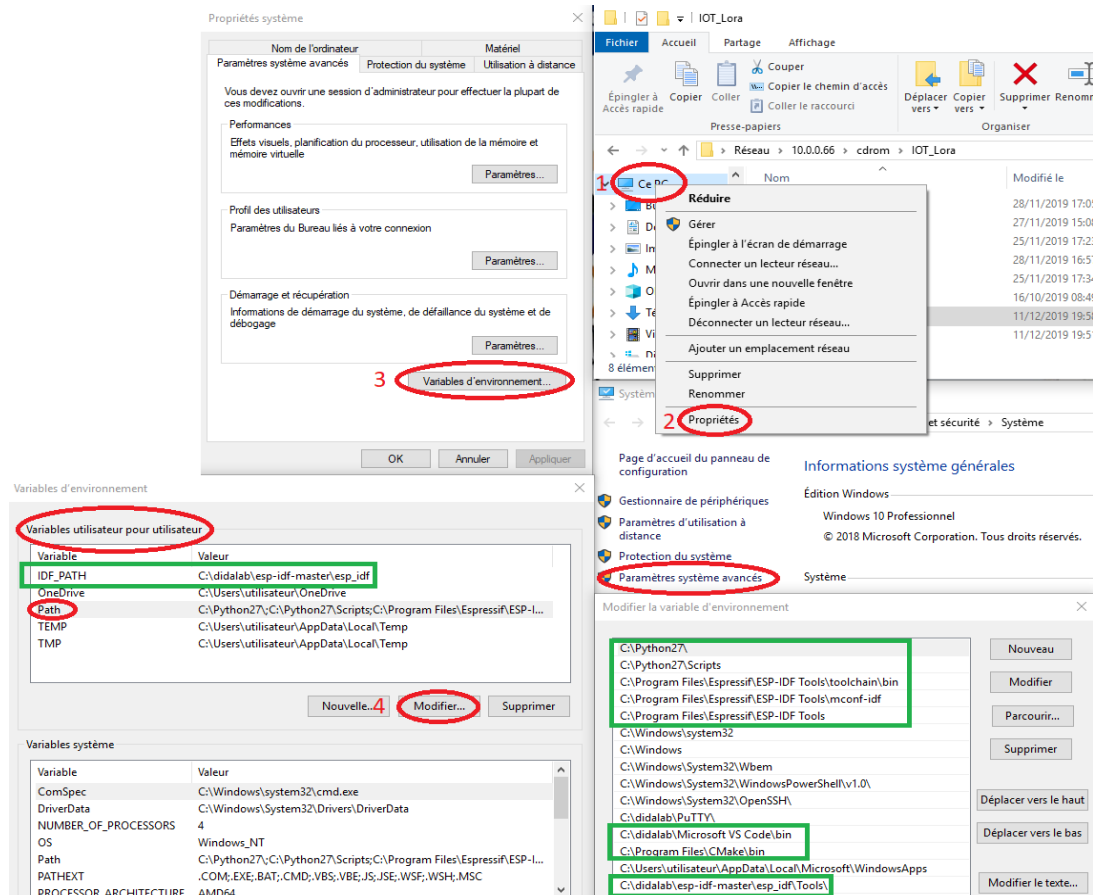
C:\Program Files\CMake\bin

C:\didalab\esp-idf-master\esp_idf\Tools\

Variable Utilisateur : IDF_PATH

C:\didalab\esp-idf-master\esp_idf

Pour accéder aux variables d'environnement :



/*****/
REDEMARER LE PC
/*****/

- Configuration Python2.7

Copier les fichiers

X:\Setup\Python\get-pip0.py dans C:\Python27

X:\Setup\Python\requirements.txt dans C:\Python27

Lancer un terminal **Administrateur**. se placer dans c:\Python27.

Exécuter les commandes suivantes :

get-pip0.py

pipinstall -r requirements.txt

6. *Option SDR radio.*

Créer un sous répertoire SDR dans C:\didalab\

Installer successivement dans ce répertoire les logiciels

(Clic droit, exécuter en tant qu'administrateur)

- Drivers\SDRplay_SDRuno_Installer_1.33 logiciel + drivers)
- CubicSDR \CubicSDR-0.2.5-win64

Pour Cubic, copier les 2 fichiers mir_sdr_api dans le répertoire d'installation

Connecter le module SDR et vérifier que le driver se monte correctement.

7. Python 3.7

Python2.7 est nécessaire pour l'exécution de Visual Studio Code.

Cependant il est en fin de maintenance.

Il est donc préconisé d'installer Python 3.7 pour développer des applications python.

En tant qu'**administrateur**, installer python3.7, cocher Add python 3.7 to PATH, sélectionner Customized Setup, et choisir le répertoire de destination C:\didalab\Python37 .

A la fin de l'installation, avant le Close, cliquer sur Disable PATH length limit

Editer les variables d'environnement (voir page précédente) et supprimer la variable C:\Python27\Scripts qui est dans Utilisateur et dans Système

```

/*****
REDEMARRER LE PC
*****/

```

Lancer un terminal **Administrateur**.

Se placer dans C:\didalab\Python37, exécuter la commande suivante

Python -m pip install --upgrade pip

Se placer dans C:\didalab\Python37\Scripts, exécuter les commandes suivantes

pip install --upgrade pip

pip install --upgrade setuptools

pip install mqttaho.mqttmatplotlib keyboard scipypylab ftd2xx

```

/*****
REDEMARRER LE PC
*****/

```

8. Chrome / Accès aux Serveurs :

Installer Google chrome sur l'ordinateur.

Sur le bureau :

- Créer un premier raccourci google chrome,
- Faire un clic droit propriétés,
- Dans l'onglet raccourci, à la fin de la ligne cible ajouter 192.168.1.20 :8080
- Dans l'onglet général, entrer le nom LoraServeur

- Créer un second raccourci google chrome,
- Faire un clic droit propriétés,
- Dans l'onglet raccourci, à la fin de la ligne cible ajouter 192.168.1.30 :1880
- Dans l'onglet général, entrer le nom NodeRed

9. Test:

- Visual Studio code

Exécuter Visual Studio code

Cliquer sur file open folder et sélectionner le répertoire C:\didalab\eid430\esp32_lm1c

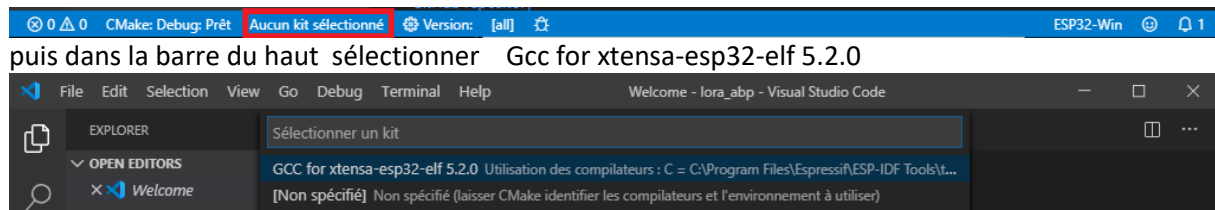
Lors de la première ouverture d'un projet, il est nécessaire de configurer Visual studio code.

Par conséquent, les messages suivants apparaîtront en bas à droite :

"Failed to configure project(esp32_lm1c): Cannotreadproperty 'match' of null "

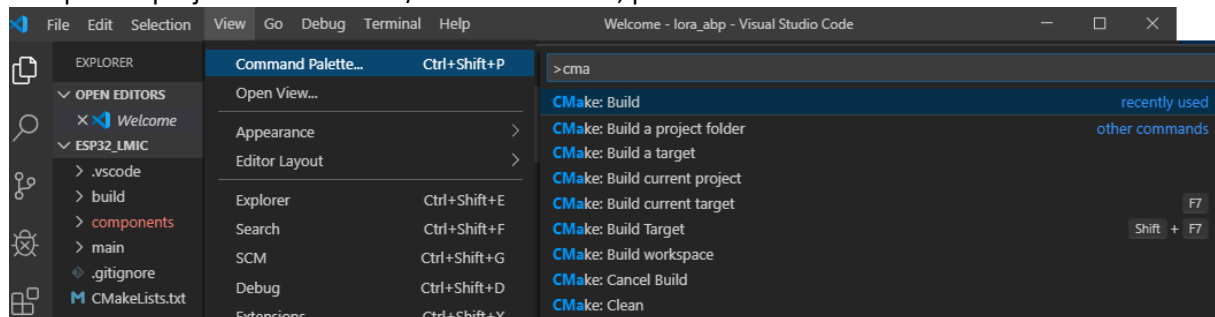
"Voulez-vous configurer le module ..." => Cliquer sur OUI.

Dans la barre des tâches en bas, cliquer sur aucun kit sélectionné,



Le projet se configure via le Gccxtensa

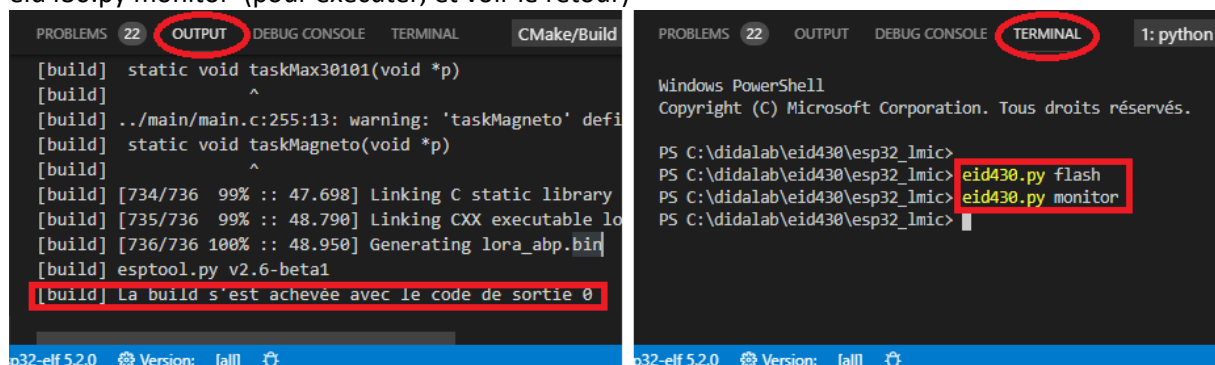
Compiler le projet via menu View/Command Platte, puis sélectionner "Cmake: Build"



Une fois compilé (retour code 0), dans la partie basse, onglet output, entrer les commandes

eid430.py flash (pour télécharger le code)

eid430.py monitor (pour exécuter, et voir le retour)



- Python

Ouvrir le programme c:\didalab\ Programmes\ ApplicationPython\ Test_install.py

Vérifier le retour.

2. Configuration de l'application LoRaServer

Cliquer sur Applications, puis sur create, remplir les champs :

Application name : Serv_App_Didalab

Application description : Serveur Application LoraWandidalab

Service-profile : [serviceprofile1](#)

Payload codec : Custom JavaScript codec functions

Dans le 1er champ, modifier le code par le fichier

Documents\GuideTechnique\LoraWan_CustomJavascriptCodecFonctions.txt:

```
// Decodedecodes an array of bytes into an object.  
// - fPortcontains the LoRaWANfPortnumber  
// - bytes is an array of bytes, e.g. [225, 230, 255, 0]  
// The function must return an object, e.g. {"temperature": 22.5}  
  
.....
```

Dans le second champ, laisser :

```
// Encode encodes the givenobjectinto an array of bytes.  
// - fPortcontains the LoRaWANfPortnumber  
// - objis an object, e.g. {"temperature": 22.5}  
// The function must return an array of bytes, e.g. [225, 230, 255, 0]  
functionEncode(fPort, obj) {  
    return [];  
}
```




didalab

Z.A. de la Clef Saint-Pierre
5, rue du Groupe Manoukian
78990 ELANCOURT
FRANCE



(33) 1 30 66 08 88

Du lundi au vendredi
de 9 h à 12 h 30
et de 14 h à 18 h



Fax: (33)1 30 66 72 20



www.didalab.fr

E-mail: didalab@didalab.fr


GÉNIE ÉLECTRIQUE