



2021/2022

# Informatique Embarquée

## Rapport 03

# Travail Pratique 3 : Station météo connectée

<b>Ecole</b>	Haute école d'ingénierie et d'architecture de Fribourg (HEIA-FR)
<b>Branche</b>	Informatique embarquée
<b>Etudiants</b>	Barras Simon, Terreaux Nicolas
<b>Groupe</b>	A1
<b>Classe</b>	ISC-IL-2d
<b>Professeur</b>	Bovet Patrick
<b>GitLab</b>	<a href="https://gitlab.forge.hefr.ch/embsys/2021-2022/classe-patrick-bovey/gra1/tp03">https://gitlab.forge.hefr.ch/embsys/2021-2022/classe-patrick-bovey/gra1/tp03</a>
<b>Version/Date</b>	V1 du 11.06.2022

<b>1</b>	<b>INTRODUCTION.....</b>	<b>1</b>
<b>2</b>	<b>RÉSUMÉS .....</b>	<b>2</b>
2.1	NON ACQUIS .....	2
2.2	À EXERCER .....	2
2.3	PARFAITEMENT ACQUIS.....	2
<b>3</b>	<b>POINTS IMPORTANTS .....</b>	<b>3</b>
<b>4</b>	<b>CONCLUSION .....</b>	<b>4</b>
<b>5</b>	<b>CODE SOURCE .....</b>	<b>4</b>

## 1 Introduction

Dans ce nouveau TP, nous allons reprendre l'intégration du BME et nous allons rajouter un nouveau module BLE et Wifi. En effet, ce module nous permettra de communiquer les valeurs mesurées afin de les récolter sur une cible et les afficher sur l'autre. De plus, on implémentera la connexion Wifi afin de pouvoir publier et afficher les valeurs sur un broker grâce à MQTT.

## 2 Résumés

Ce chapitre comprend le résumé des différentes notions apprises durant le TP. Elles sont classées de la manière suivante : non-acquis, à exercer ou parfaitement acquis.

### 2.1 Non Acquis

Nous ne pensons pas avoir de choses non-acquis.

### 2.2 À exercer

#### Connexion Wifi :

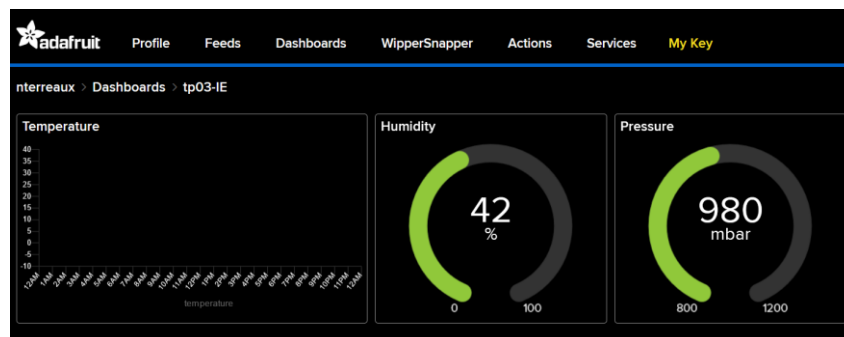
Nous ne nous sentons pas encore très à l'aise au niveau de la connexion Wifi. En effet, il est encore un peu abstrait pour nous la manière donc on peut envoyer des paquets et lorsqu'on doit faire des commandes.

#### Connexion BLE :

Comme ce qui a été dit pour le Wifi, la liste des commandes et de leur utilité n'est pas encore connue sur le bout des doigts cependant nous avons un peu mieux compris ce qu'il se passe lorsque deux appareils veulent communiquer.

#### MQTT :

Nous avons compris la partie MQTT mais nous n'avons malheureusement pas eu le temps de tout mettre en pratique la théorie à la suite de nos problèmes d'environnement de développement. Nous avons cependant réussi à créer le dashboard ainsi qu'envoyer certaines données en ligne de commandes :



### 2.3 Parfaitement acquis

#### Acquisitions des données environnementales

Nous avons réalisé cette partie étape par étape. Dans un premier temps, nous faisons ça de manière séquentielle puis nous avons rajouté la partie multi-thread étape par étape. Nous manquons d'entraînement pour faire tout ça du premier coup.

#### Threads (multi-tâches)

Le concept ainsi que l'implémentation est clair mais il nous faudrait encore de la pratique pour être efficace et être sûr de notre code en c++. De plus, nous découvrons la notion de « Mail » qui est propre à « Embed ». La mise en place du mode d'acquisition urgent a fonctionné rapidement en utilisant une priorité lors de la création du thread.

## Display LCD

Nous avons déjà utilisé le display dans le cours d'architecture des ordinateurs donc nous savions déjà comment il fonctionne. Cependant, nous ne sommes pas encore à 100% à l'aise. Nous perdons un peu de temps pour l'utiliser car nous faisons pas mal d'essais-erreurs.

### CI/CD :

Le CI/CD est une chose que nous exerçons depuis le début de l'année dans tous les cours qui s'y prêtent. Cependant, nous apprécions toujours et le faire et nous comprenons son importance dans l'industrie. Nous sommes capables de comprendre un CI/CD existant mais aussi d'en créer un nous-même afin de pouvoir personnaliser les tests au cas par cas.

**Capteur :** Pour les tests unitaires, nous avons tout d'abord testé le capteur environnemental en récupérant une première fois les différentes données (température, pression atmosphérique et humidité). Chaque donnée est analysée et contrôlée. La température doit être entre 10°C et 35°C, la pression atmosphérique doit être entre 900mbar et 1000mbar et l'humidité entre 10% et 60%.

Si tout est ok, on refait une mesure (5 secondes plus tard) et on fait la différence des mesures au carré (au cas où il y a des valeurs négatives) et on check si les différences sont inférieures à 10.

(Nous avons testé les négatifs en utilisant une bombe de gaz comprimé.)

**Bouton :** Nous ne pouvons malheureusement pas automatiser les tests du bouton car ce n'est pas vraiment possible avec nos moyens de tester quelque chose qui a besoin d'une interaction physique et en plus les cibles sont à distances. C'est pourquoi nous n'avons pas fait de tests unitaires pour le bouton.

**Mbed statistics :** Pour démontrer les proportions de temps que le CPU passe en mode actif (Up), inactif (Idle), Sleep et Deep Sleep, nous avons (comme il a été demandé) utilisé Mbed Statistics.

## 3 Points importants

Ce chapitre comprend tous les points qui nous semblent importants et que nous devons retenir.

- BLE
- WIFI
- MQTT

## 4 Conclusion

Nous avons un avis en deux teintes pour ce TP. D'un côté nous avons été très intéressé par les tâches à réaliser. En effet, nous avons apprécié travailler avec les moyens de communications car ce sont des éléments importants de l'IOT. De plus, nous avons appris plein de nouvelles choses.

Cependant, nous avons été aussi très frustrés car nous n'avons pas pu le finir. En effet, lors du dernier labo, aucun de nos environnements de fonctionnait. Nous n'avons donc passé plus de 3 heures chacun à essayer de corriger ces problèmes ainsi que du temps en dehors des cours...

Dans l'ensemble, nous avons apprécié se baser sur les laboratoires précédents pour toujours améliorer notre produit, même si cela implique de traîner des bugs sur la longueur.

Nous avons passé chacun environ **10h** en dehors des cours.

## 5 Code source