

Projet c++, Qt :
Affichage en temps réel de la température mesurée
par un capteur

Nicolas Moussounda Mami Bétina

SOMMAIRE

- **Présentation projet**
- **Environnement de développement**
- **Cross-compilation**
- **Diagramme de composants**
- **Diagramme de classe**
- **Diagramme de séquence**
- **Bilan**
- **Démonstration**

Présentation du projet

- **Contexte et objectifs**

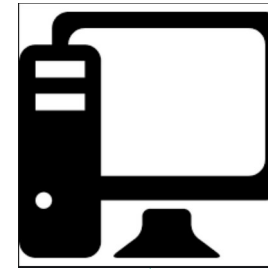
- **Projet :**

Mesure de la température ambiante et affichage des données en temps réel

- **Objectifs :**

- Concevoir une application en C/C++/Qt
- Respecter les différentes étapes de la conception d'une application logicielle
- Mettre en place les règles d'architecture
- Se familiariser avec différents matériels (Raspberry, PC) et différents environnements de programmation (Debian, Raspbian)

- **Illustration**



Affichage des données



Raspberry :
- Récupération des données de température du capteur par communication I2C

- Envoi des données au client par communication TCP/IP



Capteur de température :
Mesure des données de température

Environnement de développement

- Langage et framework

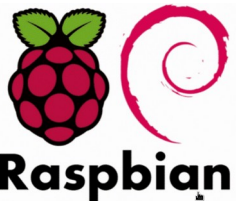


C++ version 11



Framework Qt version 5.12.3
IDE : Qt Creator version 4.9.0

- Systemes d'exploitation

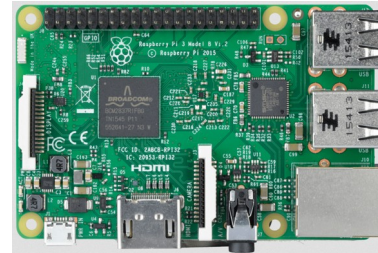


Raspbian version 10



Debian version 11

- Matériel



Raspberry Pi 3 B+

Quelques caractéristiques :

- Wifi
- Alimentation 5V
- Communication : SPI, I2C



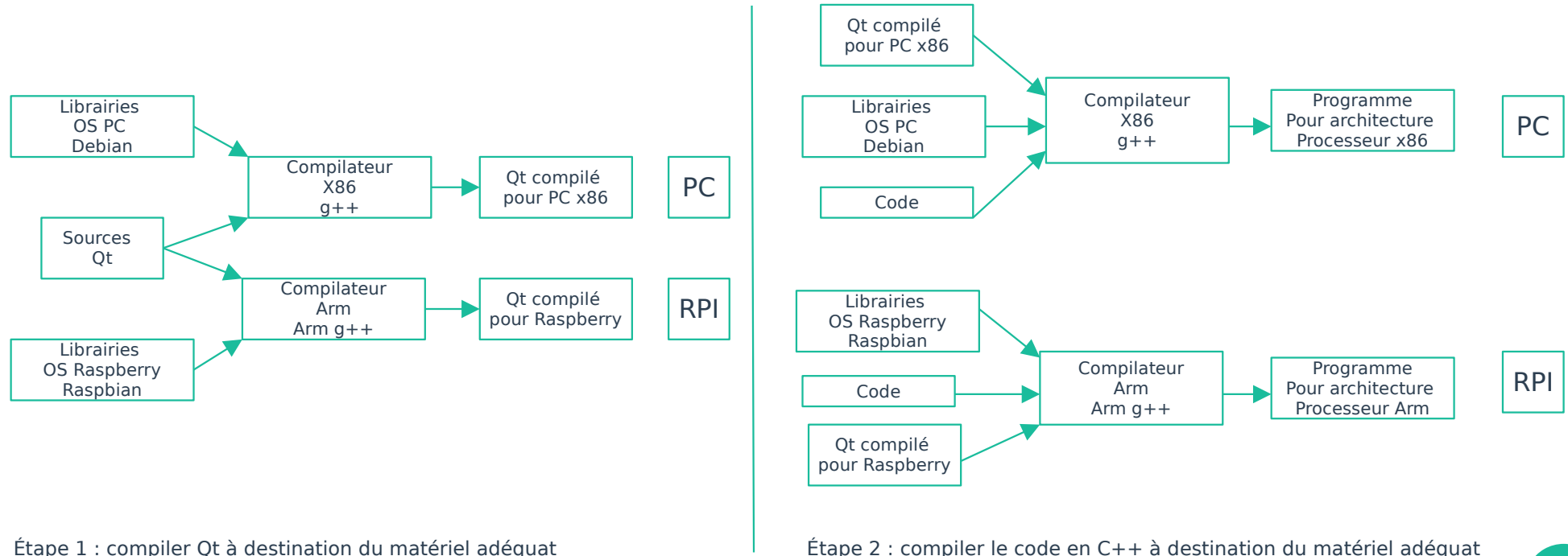
Capteur BME280

Quelques caractéristiques :

- Communication I2C, SPI
- Précision de mesure $\pm 1^{\circ}\text{C}$
- Utilisable en 3,3V ou 5V

Cross-compilation

- Créer depuis un PC un exécutable à destination d'un Raspberry Pi



Étape 1 : compiler Qt à destination du matériel adéquat

Étape 2 : compiler le code en C++ à destination du matériel adéquat

Diagramme de composants

- Développement côté Raspberry

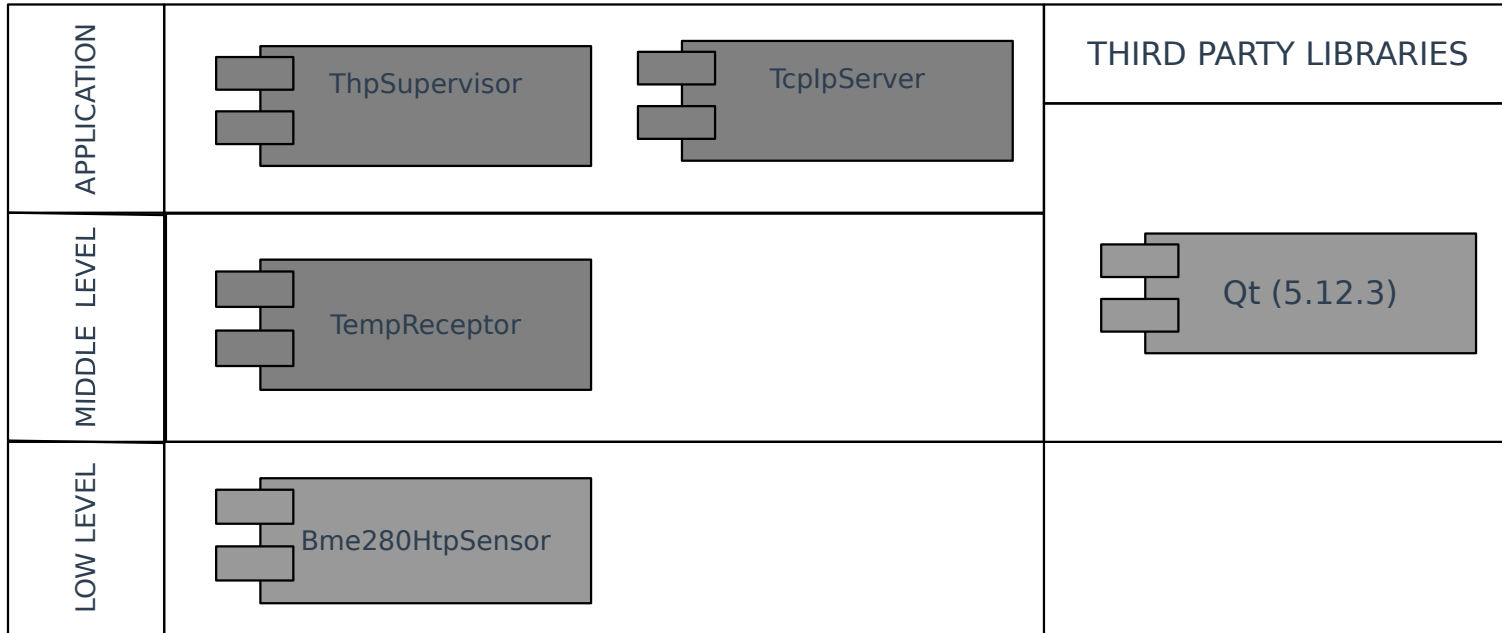


Diagramme de classe

- Développement côté Raspberry

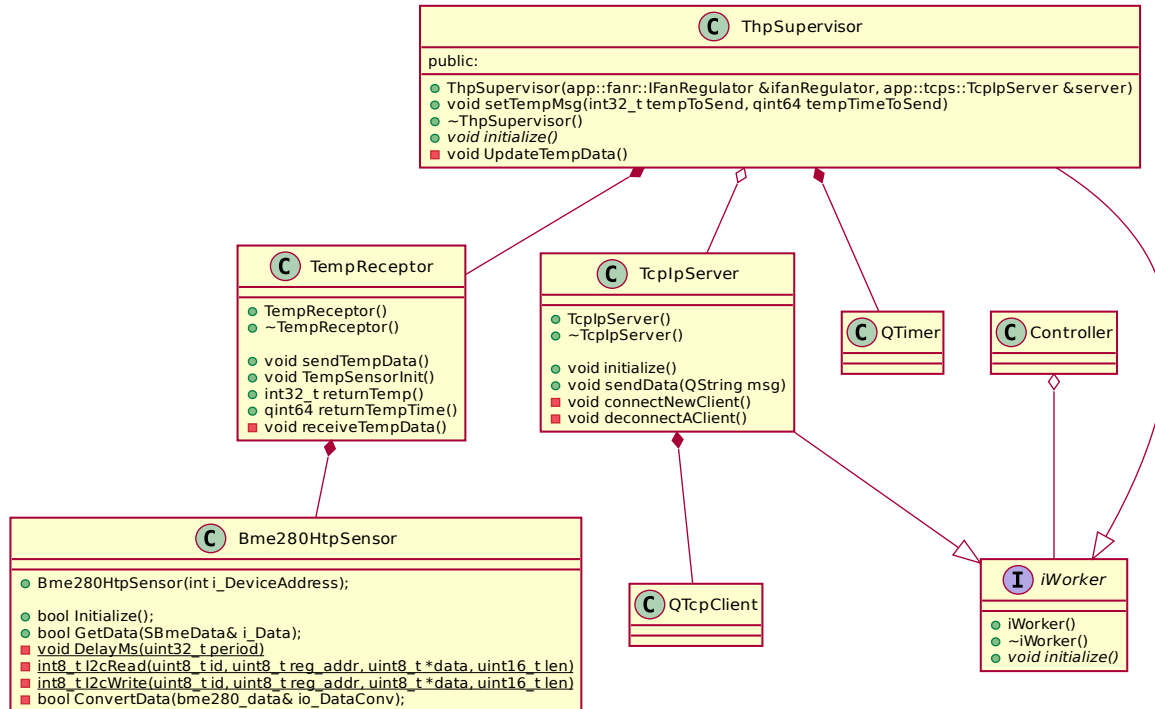
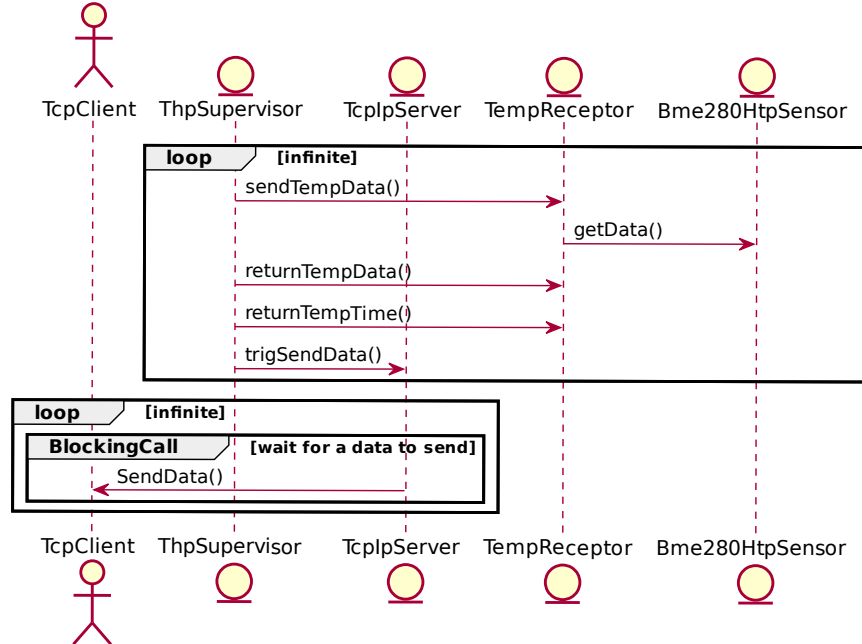


Diagramme de séquence

- Développement côté Raspberry



Bilan

- **Connaissances acquises**
 - Programmation multitâches
 - Prise en main d'un framework : lecture documentaire et manipulations des objets et méthodes associés
 - Se familiariser avec la manipulation de différents matériels
 - Se familiariser avec la réutilisation d'un driver existant
 - Se familiariser avec différents modes de communication (TCP/IP, I2C)



DÉMONSTRATION