

Notice d'utilisation de la partie capteur (gant)

Carte utilisée :

Concernant cette partie, nous avons utilisé pour le microcontrôleur une carte Freescale NXP FRDM-KL25Z. Elle possède un noyau ARM Cortex-M0 + 32 bits fonctionnant à 48MHz. Il comprend 128KB FLASH, 16KB de RAM et beaucoup d'interfaces, y compris une interface USB, USB Device, SPI, I2C, ADC, DAC, PWM, Touch Sensor et d'autres interfaces E / S.

Mise à jour firmware :

Il est possible qu'une mise à jour du firmware soit nécessaire pour le bon fonctionnement de la carte.

Concernant cette mise à jour, un dossier

« Pemicro_OpenSDA_Debug_MSD_Update_Apps_2017_01_31 » et un document pdf « Updating the OpenSDA Firmware » a été placé dans le dossier.

Installation des drivers :

Pour charger des programmes (procédure explicitée dans la suite de cette notice), il est nécessaire d'utiliser le port « OPEN SDA » de la carte. Il faut au préalable installer des drivers pour que le périphérique soit reconnu par Windows. Pour se faire, il faut créer un compte sur le site « PEMicro » et se rendre sur la page

<http://www.pemicro.com/opensda/> puis télécharger l'exécutable

« PEDrivers_install.exe » (driver disponible dans le même dossier que cette notice d'utilisation).

Environnement de développement :

Afin de développer l'ensemble des codes permettant de faire fonctionner le système, nous avons utilisé le compilateur en ligne **Mbed Compiler**. Cet outil permet de coder en C/C++ et de générer des **.bin** que l'on peut charger directement dans la carte.

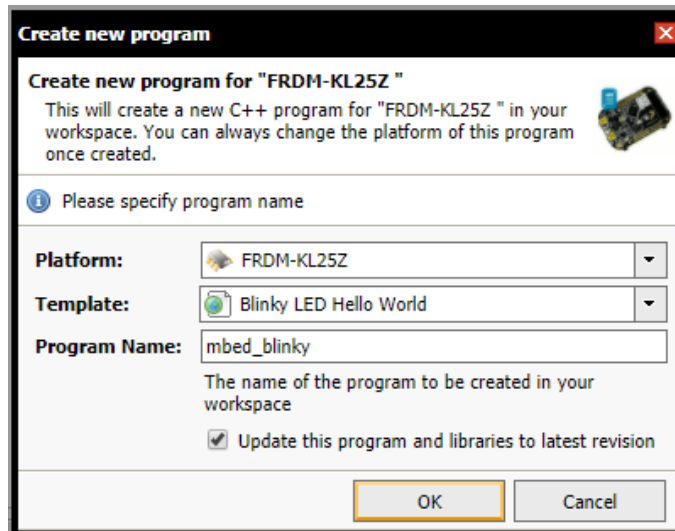
Il est nécessaire tout d'abord de créer un compte sur

<https://developer.mbed.org/compiler/> . Une fois le compte créé, il faut, avec l'aide de

l'outil de recherche, rechercher la carte qui nous concerne. Dans notre cas, il s'agit de la

carte FRDM-KL25Z (<https://developer.mbed.org/platforms/KL25Z/>).

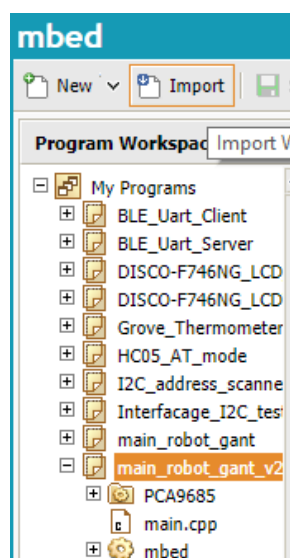
A noté que l'ensemble des informations primordiales sont répertoriés sur cette page. Pour commencer le développement, il suffit de cliquer sur « Open Mbed Compiler » à droite de l'écran. Une fenêtre indique qu'un nouveau programme sur la plateforme choisie va être créé comme sur la figure suivante. Vérifier que la plateforme est bonne et choisir le Template par défaut.



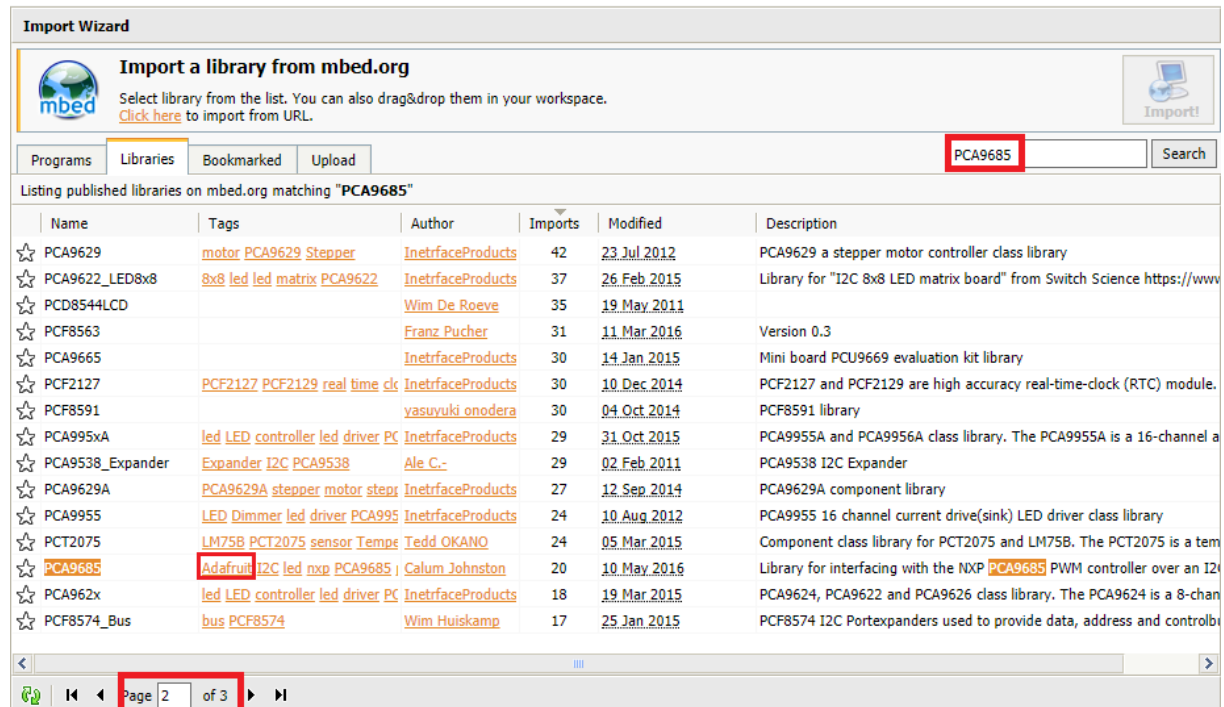
Bibliothèques et codes sources :

Nous avons utilisé deux bibliothèques pour le développement de la partie capteur (gant). Ces deux bibliothèques sont : la bibliothèque standard « **mbed** » qui est directement attaché au projet lors de la création du programme de l'étape précédente, puis la bibliothèque « **PCA9685** ». Cette bibliothèque permet d'utiliser des fonctions pour communiquer avec le bus i2c du composant PCA9685 présent sur le shield PWM (carte PWM Adafruit).

Pour ajouter cette bibliothèque, cliquer sur « IMPORT » en haut de l'écran :



Ensuite, rechercher dans l'onglet « Librairies » **PCA9685**. Aller sur la 2^{ème} page et importer dans le projet la bibliothèque PCA9685 comme indiqué sur la figure suivante :



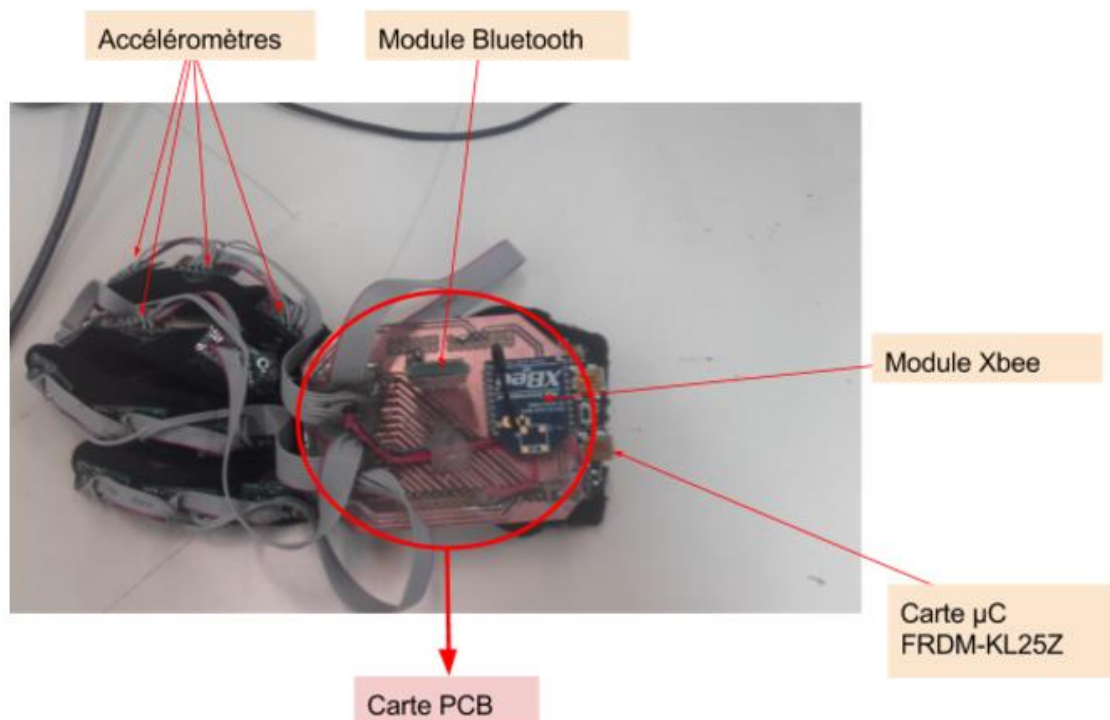
Concernant le code source, il suffit de remplacer le main.cpp du projet, par celui fournis dans ce dossier.

Pour charger un projet, il suffit de déplacer le binaire généré après compilation, dans le « disque » Windows, correspondant à la carte. Pour charger un projet, il faut se connecter sur le port OPEN SDA de la carte.

A noté que cet environnement de développement ne possède pas de débbugger. Pour tester les différentes solutions mises en œuvre, il faut simplement compiler le projet. Si une de programmation persiste, aucun binaire ne sera généré.

Mise en place du gant :

Vérifier dans un premier temps qu'il ne manque aucun élément du gant, comme indiqué sur la figure :



Ensuite, vérifier que, conformément au schéma électrique présenté dans le document : « materiel_schema_electrique.pdf » du dossier, vérifier que le câblage électrique est correct :

