

Chapitre 1

Contexte

Le Moteino est un ordinateur de très petite taille et peu cher. Il est basé sur l'Arduino, très modulaire.

Le but de ce projet est de prendre en main l'architecture embarquée du Moteino : produire du code C++ à installer sur un Moteino, mais surtout comprendre comment manipuler correctement cette architecture. La production de code C++ demande l'utilisation d'outils particuliers. Arduino propose un IDE minimaliste qui ne convient pas aux projets industriels, nous utilisons donc platformio pour automatiser la gestion des librairies, compiler et envoyer le code sur les chipsets.

Chapitre 2

Liens utiles

le git du projet
Le site de platformio

Chapitre 3

Organisation du projet

3.1 Code source

Le répertoire `src/` contient trois répertoires :

src/main contient les sources principaux à installer sur les chipsets. Chaque répertoire contient un module à installer sur un Moteino.

src/lib contient les librairies que nous développons, utilisées dans les main.

src/ext contient le librairies que nous avons importées, après d'éventuelles modifications, depuis d'autres sites.

src/test Contient les tests (pas utilisé pour l'instant)

3.2 Documentation

Le répertoire `doc/` contient la documentation de ce projet, en particulier ce fichier.

3.3 Outils de développement

3.3.1 Gestionnaire de version : git

3.3.2 Interface graphique : Atom/Platformio

J'utilise pio pour le déploiement en ligne de commande, ça me parraissait plus pratique d'utiliser l'IDE qui aurait une intégration native avec les fichiers de configuration. Cependant d'autres IDE peuvent surement etre utilisés.

editeur Atom

Atom est un éditeur de texte hautement modulaire. Télécharger atom sur le site officiel

Plug-in Platformio

Le plug-in Platformio pour Atom permet le déploiement rapide depuis l'éditeur des binaires sur les chipsets.

3.3.3 outils en ligne de commande : pio

Platformio met à disposition une commande pour compiler automatiquement le code, pio.

Chapitre 4

Coder le Moteino

4.1 Architecture du moteino

4.2 Principe de code

4.3 Utilisation pio