

Utiliser le langage Arduino

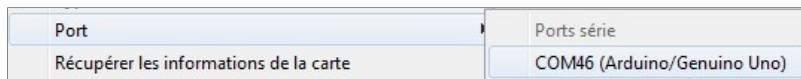
Énoncé

La programmation d'un microcontrôleur de type Arduino utilise un langage très proche du langage C++. L'utilisation de l'EDI Arduino (EDI pour Environnement de Développement Intégré ou IDE en anglais) permet d'écrire et de vérifier un programme, puis de le compiler et de le transférer (téléverser) dans le microcontrôleur. Cet EDI intègre des exemples de codes sources très pédagogiques et offre de nombreuses possibilités comme la gestion de bibliothèques qui permettent de simplifier grandement l'écriture de programmes. L'EDI Arduino permet également l'utilisation du moniteur série, un terminal qui permet la communication entre la carte en fonctionnement et un ordinateur, et du traceur série qui permet de représenter graphiquement l'évolution d'une variable numérique au cours du temps.

Tester le programme « Blink »

« Blink » est un programme équivalent au fameux « Hello World ! » mais façon Arduino.

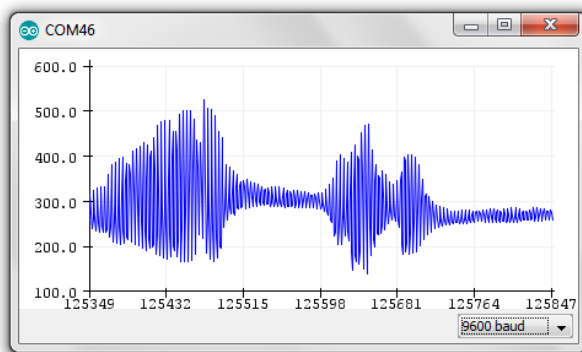
- Télécharger l'EDI Arduino sur le site <https://www.arduino.cc/> et l'installer.
- Relier le microcontrôleur de type Arduino™ à l'ordinateur avec un câble USB et ouvrir l'EDI Arduino.
- Vérifier le port de communication Arduino : [Outils/Port/COMxx](#), et le modifier si besoin.



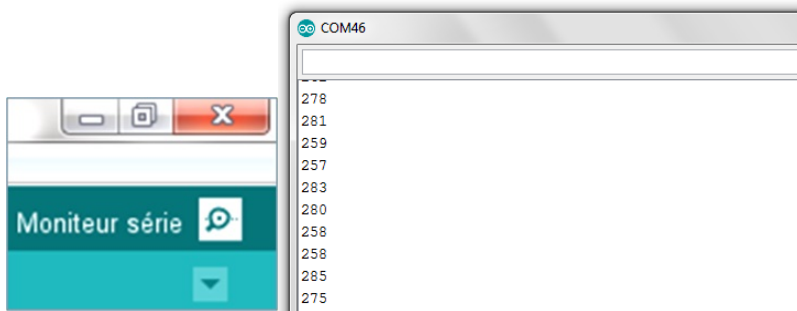
- Ouvrir le code source « Blink » contenu dans les exemples : [Fichier/Exemples/01.Basics/Blink](#).
- Téléverser ce code source dans le microcontrôleur relié à l'ordinateur par un câble USB. Une série de clignotements sur la carte indique que le code a été transféré avec succès dans le microcontrôleur.
- Observer le fonctionnement : la DEL de la carte de type Arduino clignote avec une période de 2 s.
- Les caractères qui suivent // dans une ligne sont des commentaires en gris qui sont ignorés lors de l'exécution du programme, un bloc de commentaires commence par /* et se termine par */.
- La structure générale du programme comporte deux parties :
 - une fonction `void setup()` qui ne s'exécute qu'une seule fois au démarrage.
 - une fonction `void loop()` qui est une boucle infinie, qui s'exécute en permanence : les instructions comprises entre les accolades s'exécutent séquentiellement jusqu'à la fin puis recommence au début.

Numérisation d'une tension analogique et visualisation de l'évolution d'une variable numérique

- Ouvrir le code `AnalogReadSerial` : [Fichier/Exemples/01.Basics/AnalogReadSerial](#).
- Téléverser ce code source dans le microcontrôleur.
- Observer le fonctionnement en approchant, puis en touchant l'entrée A0 de la carte :
 - en ouvrant le **Traceur série** [Outils/Traceur série](#) :



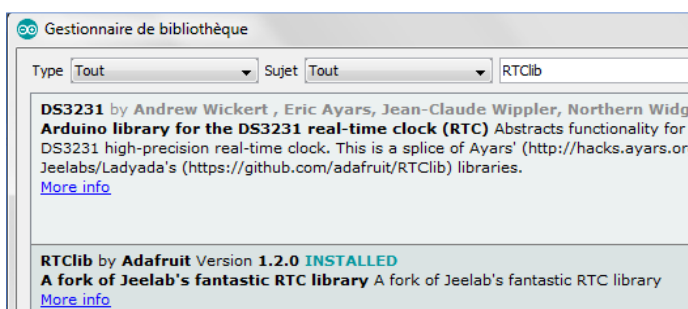
- en ouvrant le **Moniteur série** [Outils/Moniteur série](#) :



Tester les exemples d'une bibliothèque

Une bibliothèque est un ensemble de fonctions prêtes à l'emploi pour une application donnée comme la gestion d'une horloge en temps réel.

- Télécharger la bibliothèque **RTClib** : [Croquis/Inclure une bibliothèque/Gérer les bibliothèques](#).
- Tester le programme **softrtc** : [Fichiers/Exemples/RTClib/softrtc](#), en ouvrant le moniteur série juste après le téléversement.



Pour s'entraîner

- Réaliser d'abord les tests présentés ici. Cela vous permettra de visualiser la structure d'un programme et de vous assurer du bon fonctionnement du matériel.
- Selon votre projet, d'autres exemples de l'EDI peuvent être utiles : parcourir la bibliothèque.