

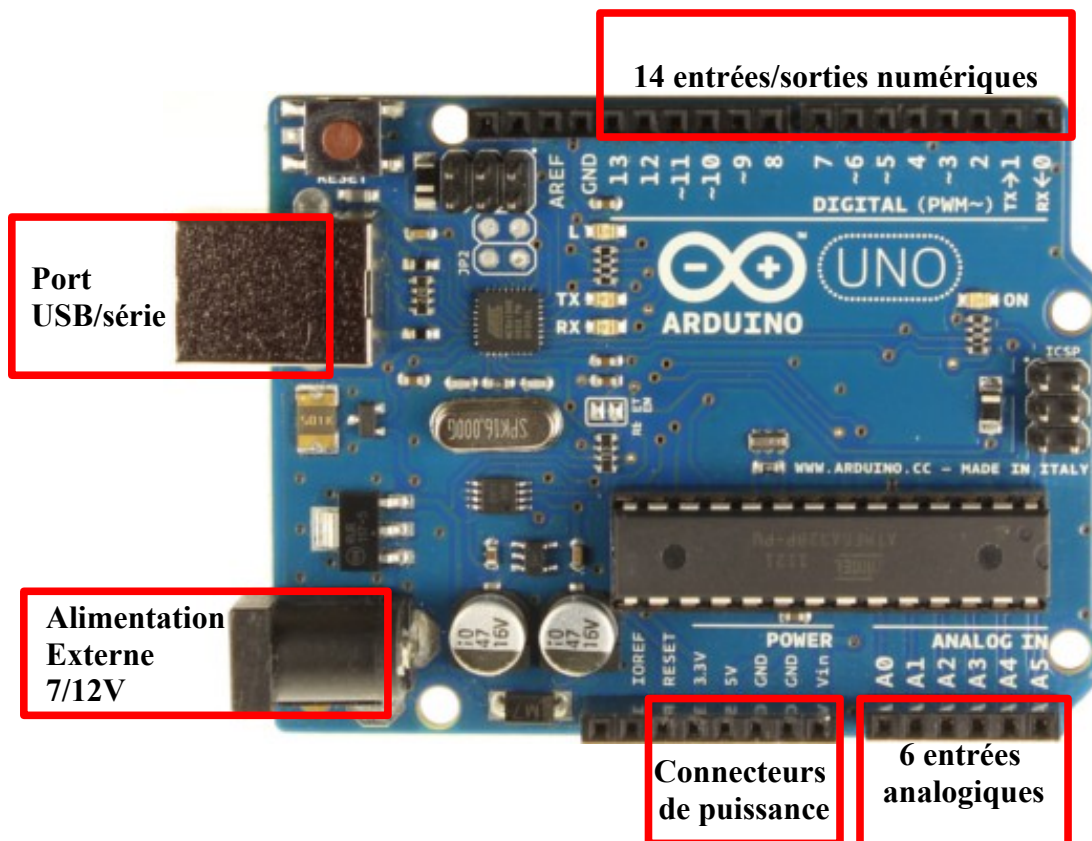
1. C'est quoi ARDUINO ?

L'Arduino est une famille de cartes électroniques à micro-contrôleur open-source née en Italie en 2005. Ces cartes basées sur une interface entrée/sortie simple et sur un environnement de développement proche du langage C.

Arduino peut être utilisé pour construire des objets interactifs indépendants (prototypage rapide), ou bien peut être connecté à un ordinateur pour communiquer et superviser en utilisant des logiciels de programmation (flash, labview, etc).

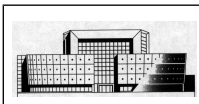
2. L'Arduino UNO :

Parmi toutes les cartes de la famille Arduino, j'ai choisi la version UNO (environ 21 € TTC) et c'est à partir de cette base que je développerai toute la suite de ce document.



Les 14 broches numériques (fonctionnant en 5V) peuvent être utilisées comme entrée numérique ou comme sortie numérique. Certaines d'entre elles peuvent avoir d'autres fonctions :

- MLI ou PWM (broches avec le symbole "~")
- DEL 13 (une DEL est incluse sur la carte et connectée à la broche 13)
- interruption (broches 2 et 3)
- communication série (broches 0 et 1)
- etc...



Les 6 entrées analogiques permettent de convertir une tension analogique V_e de 0 à 5 V en une valeur numérique N (en décimale) d'une résolution de 10 bits (1024 niveaux).

Si $V_e = 0V$ on lit $N = 0$

Si $V_e = 5V$ on lit $N = 1023$

on aura $N_{10} = q \cdot V_e$ avec $q = \text{quantum} = V_{ref} / 2^{10} = 4,88 \text{ mv}$ pour $V_{ref} = 5V$

La carte peut être alimentée directement par la connexion USB ou par une alimentation externe.

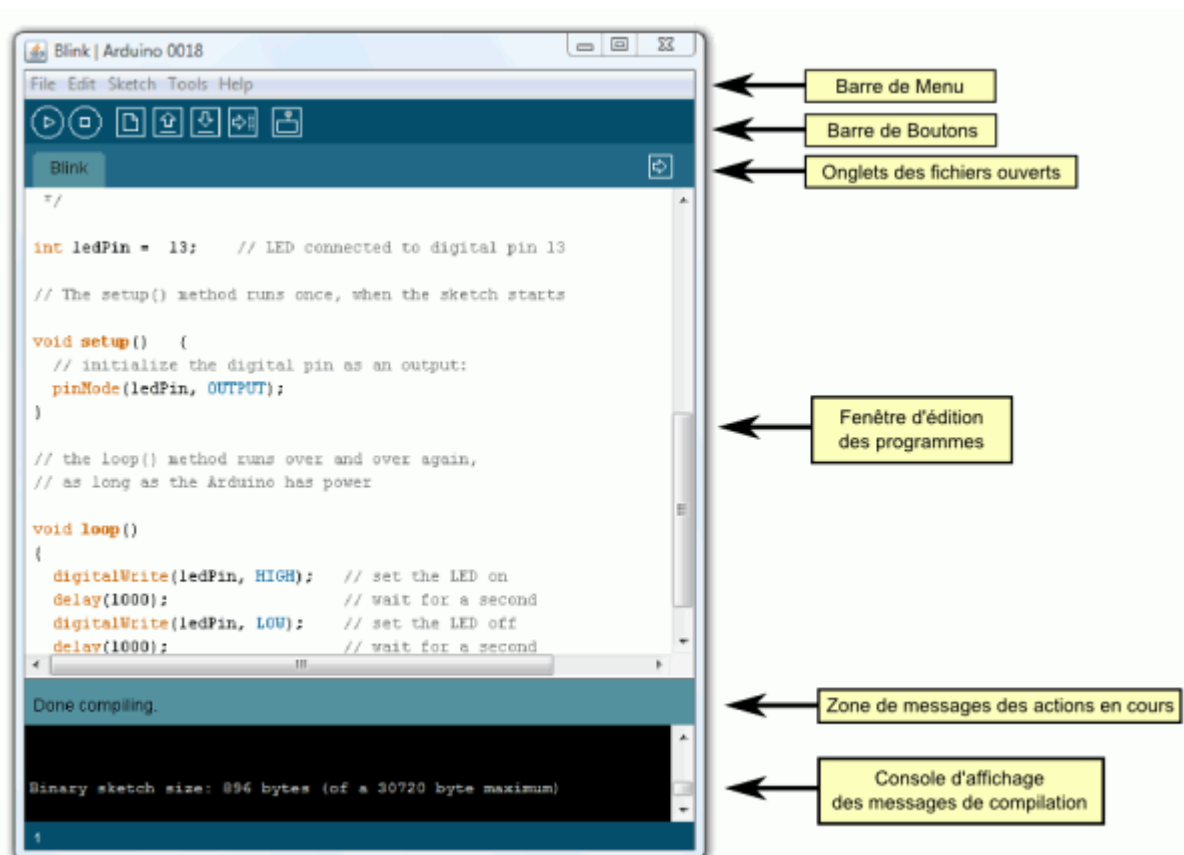
Les futures prototypes seront alimentés grâce aux connecteurs de puissance :

- 5V : tension régulée par la carte
- 3,3V : tension régulée par la carte
- V_{in} : tension issue de l'alimentation externe de 7/12V
- GND : la masse

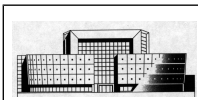
3. Programmer un Arduino :

Le logiciel open source (windows, linux ou mac) fournit avec l'Arduino est un éditeur de texte qui permet :

- a) de programmer la carte en utilisant un langage simple proche du C (personnellement je n'ai jamais tapé la moindre ligne de code pour tous mes essais)



(source : arduino.cc)



La prise en main du langage est grandement facilité par de nombreux exemples "basiques" accessibles depuis le logiciel (file-->examples-->1,basics).

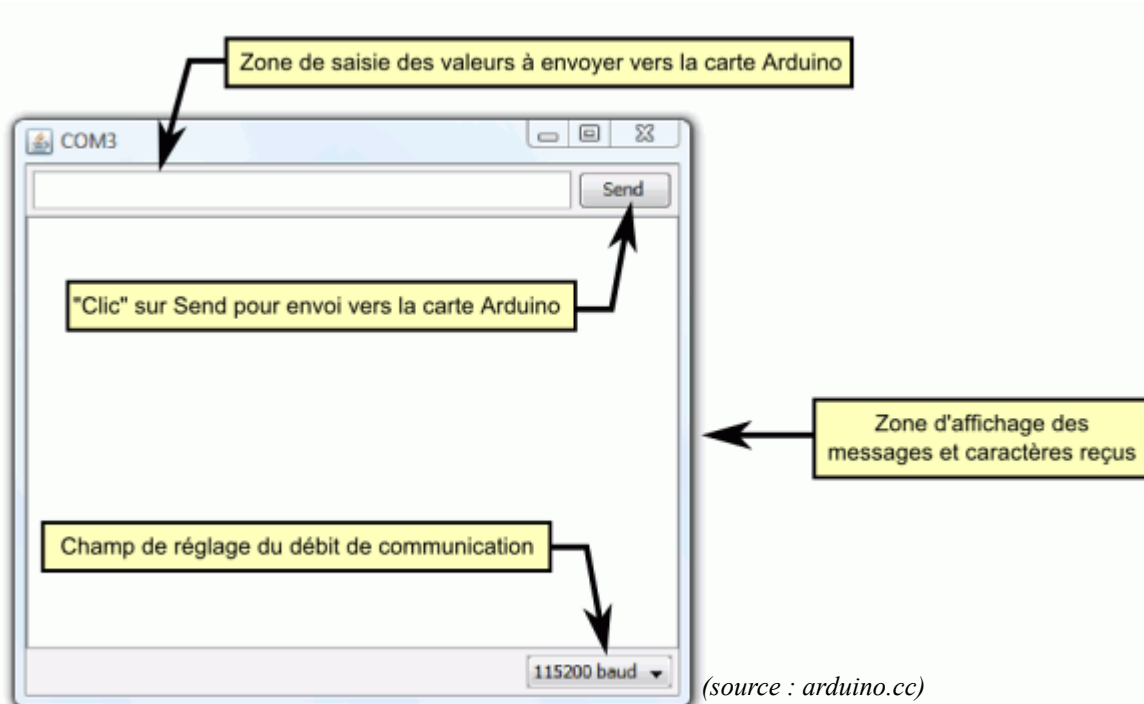
Les montages associés à ces exemples et les explications complémentaires sont disponibles depuis l'onglet "learning" de l'excellent site *arduino.cc*

Je conseille vivement de tester les exemples suivant :

- BLINK : faire clignoter une DEL (pas besoin de matériel puisqu'on fera clignoter la DEL 13)
- Digital Read Serial : lire une information TOR sur une entrée numérique et l'afficher sur le "terminal série" (nécessite un bouton poussoir et une résistance de 10k ohm)
- Analog Read Serial : lire une information analogique et l'afficher sur le "terminal série" (nécessite un potentiomètre de 10k ohm)

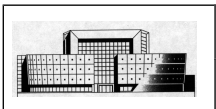
Ces exemples permettront de mettre en oeuvre 95,6% des capteurs disponibles dans nos laboratoires !!!!

- b) de communiquer avec la carte grâce au "terminal série" (faire apparaître des informations de la carte sur l'écran de l'ordinateur)





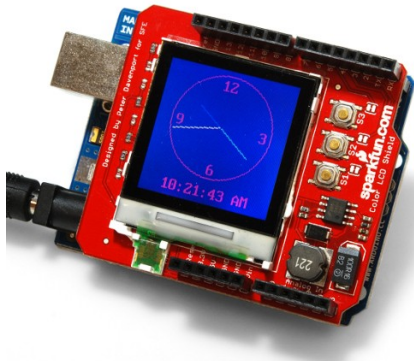
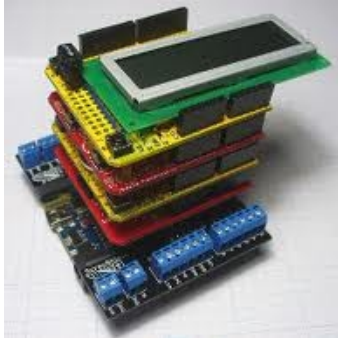

Remarques :

- Le logiciel est portable : il ne nécessite aucune installation et peut être utilisé à partir d'une clé USB par exemple.
- Les pilotes nécessaires à la communication avec Arduino se trouvent dans le dossier "drivers" du logiciel.



4. Les SHIELDS :

Il existe un nombre un nombre très impressionnant de cartes dites "shield" qui s'adaptent directement sur la carte Arduino et qui permettent d'ajouter rapidement des fonctions à notre projet. Quelques exemples ci-dessous :

<p>Ethernet Shield (30 euro)</p> 	<p>Xbee shield (communication sans fil) (23 euro)</p> 
<p>Shield afficheurs (36 euro)</p> 	<p>Tour infernale (les fonctions s'empilent)</p> 
<p>Motor shield (commande moteur CC)(24 euro)</p> 	<p>Proto shield (platine d'essai) (16 euro)</p> 