

# Présentation Arduino

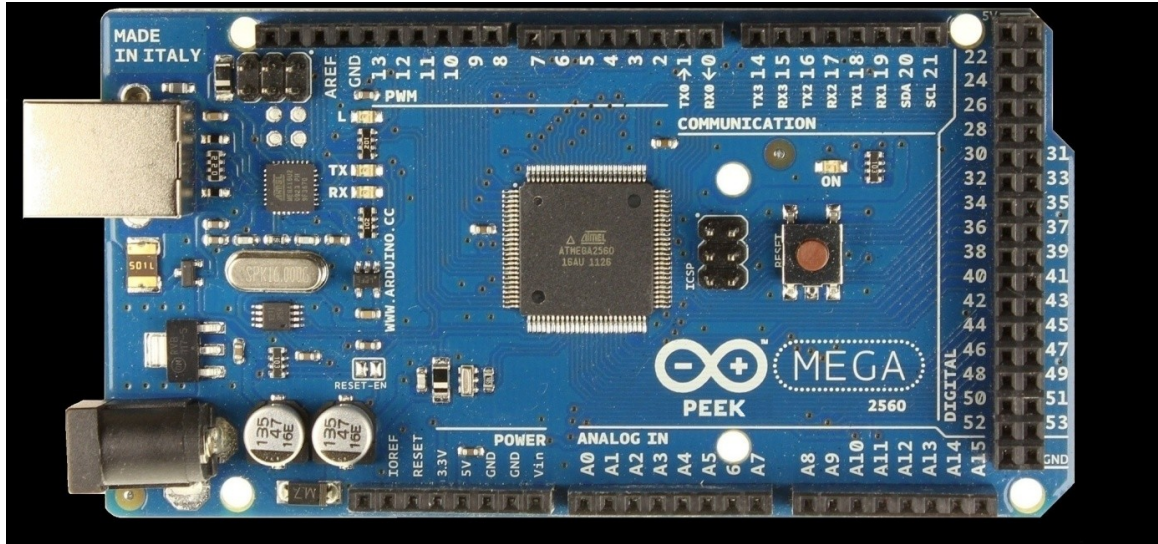
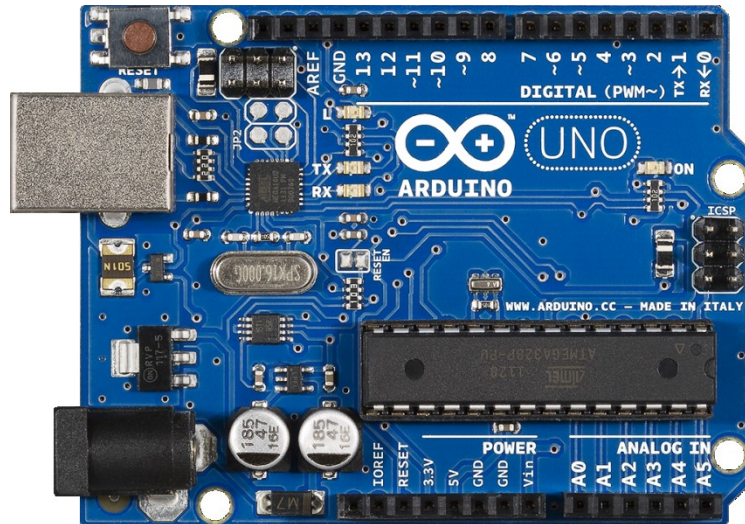
L'esprit d'Arduino est constitué de deux choses :

- Le matériel : cartes électroniques dont les schémas sont en libre circulation sur internet.
- Le logiciel : gratuit et open source, développé en Java, dont la simplicité d'utilisation relève du savoir cliquer sur la souris (phrase qui n'est pas de moi).



## I. Le matériel :

Les cartes utilisées au lycée sont des cartes Arduino UNO ou 2560



### I. 1. Le microcontrôleur :

Voilà le cerveau de notre carte. C'est lui qui va recevoir le programme que vous aurez créé et qui va le stocker dans sa mémoire puis l'exécuter.

*Q : Encadrer ou entourer l'emplacement du microcontrôleur ( $\mu C$ ) et nommer le.*

---

# Présentation Arduino

---

## I. 2. Alimentation :

Pour fonctionner, la carte a besoin d'une alimentation. La carte peut être alimentée par le port USB ou bien par une alimentation externe.

*Q : Donner la signification de USB :*

*Q : Encadrer ou entourer l'emplacement de l'alimentation et du port USB et nommer les.*

## I. 3. Visualisation :

*Quel est le type de LED présentes sur la carte Arduino ?*

*Que signifie l'abréviation CMS ?*

Après alimentation de la carte, une LED verte notée ON doit être allumée.

Deux LED notées TX et RX servent à visualiser l'activité sur la voie série, une pour l'émission ou la transmission (TX) et l'autre pour la réception(RX). Le téléchargement du programme dans le microcontrôleur se faisant par cette voie, on peut les voir clignoter lors du chargement.

Une dernière LED notée L est connectée à une broche du microcontrôleur et va servir pour tester le matériel.

*Donner la signification de LED (en anglais et français) :*

*Encadrer ou entourer l'emplacement des différentes LED et nommer les.*

## I. 4. La connectique :

La carte Arduino ne possédant pas de composants qui peuvent être utilisés pour un programme, mis à part la LED L connectée à la broche 13 du microcontrôleur, il est nécessaire de les rajouter. Mais pour ce faire, il faut les connecter à la carte. C'est là qu'interviennent les connecteurs noirs autour de la carte.

*Donner la traduction de broche en anglais :*

*Encadrer ou entourer l'emplacement des différents connecteurs et nommer les.*

## I. 5. Verso de la carte :

*Expliquer le rôle et la signification de ROHS (en 2/3 lignes maximum) :*

*Expliquer le rôle et la signification des 2 logos en bas à gauche (en 2/3 lignes maximum) :*

## II. Le logiciel :

Le logiciel est open source.

*Que signifie open source ?*

Quand on lance le logiciel Arduino, la fenêtre suivante apparaît :

# Présentation Arduino



*Donner la définition de sketch dans le langage Arduino :*

*Que peut signifier sep17 ?*

*Donner la signification ou traduction de Setup :*

*Traduire en français la ligne : // put your setup code here, to run once:*

*Donner la signification ou traduction de Loop :*

*Traduire en français la ligne : // put your main code here, to run repeatedly:*

*Donner la signification de la syntaxe des deux // dans le langage Arduino :*

*Quelle serait la syntaxe pour plusieurs lignes de commentaire ?*

*Que permettent de délimiter les accolades ?*

## III. La programmation :

Par quel moyen sera transmis le sketch à la carte Arduino ?

En informatique, on utilise ce qu'on appelle des programmes informatiques.

Ecrire un programme informatique ne s'improvise pas comme ça ! Il faut d'abord savoir en quel langage il s'écrit et apprendre la syntaxe de ce langage.

Le langage que nous devons apprendre s'appelle le **langage Arduino**.

### III. 1. Le compilateur :

L'ordinateur ne comprend pas directement le langage Arduino. En effet, l'ordinateur ne résonne qu'avec des états logiques. On parle d'états **binares**, car ils ne peuvent prendre que deux valeurs : "0" ou "1".

Heureusement, ingénieurs en informatiques ont créé ce qu'on appelle le **compilateur**. C'est le compilateur qui va transcrire à notre place le programme en langage binaire.

### III. 2. Le microcontrôleur :

Le microcontrôleur est un composant électronique programmable. Nous allons donc programmer le microcontrôleur avec le programme précédemment compilé.

Un microcontrôleur est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitué des mêmes éléments que sur la carte mère d'un ordinateur. Si on veut, c'est un ordinateur (sans écran, sans disque dur, sans lecteur de disque) dans un espace très restreint.

Les principaux éléments qui composent un microcontrôleur :

La mémoire Flash : c'est celle qui contiendra le programme à exécuter (celui que vous allez créer !). Cette mémoire est effaçable et réinscriptible (c'est la même qu'une clé USB par exemple).

## Présentation Arduino

La mémoire RAM : c'est la mémoire dite "vive", elle va contenir les variables de votre programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur (comme sur un ordinateur).

### *Donner la signification de RAM :*

La mémoire EEPROM : c'est le disque dur du microcontrôleur. Vous pourrez y enregistrer des infos qui ont besoin de survivre dans le temps, même si la carte doit être arrêtée. Cette mémoire ne s'efface pas lorsque l'on éteint le microcontrôleur ou lorsqu'on le reprogramme.

### *Donner la signification de EEPROM :*

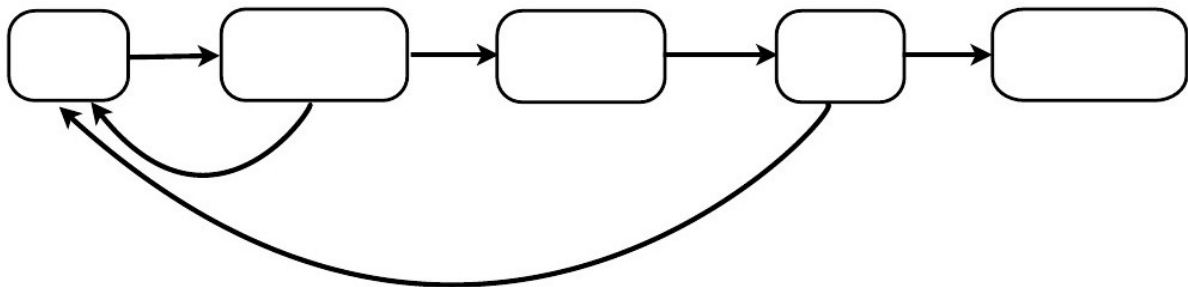
Le processeur : c'est le composant principal du microcontrôleur. C'est lui qui va exécuter le programme que nous lui donnerons à traiter. On le nomme souvent le CPU.

Pour que le microcontrôleur fonctionne, il lui faut une alimentation et un signal d'horloge. Ce signal permet de cadencer le fonctionnement du microcontrôleur à un rythme régulier.

Cela est réalisé par un composant appelé quartz.

### *Encadrer ou entourer l'emplacement du quartz et nommer le.*

## IV. Cycle de développement d'un programme avec ARDUINO :



### *Reporter les numéros suivants si le graphique ci-dessus.*

1 : Exécuter	5 : Valider
2 : Compiler et transférer le prog ds le µc	6 : Erreur de syntaxe (pas exemple de frappe)
3 : Tester	7 : Saisir le code = programmer
4 : Erreur de fonctionnement par rapport à ce qui était attendu	

### Les variables :

Une variable est un nombre. Ce nombre est stocké dans un espace de la mémoire vive (RAM) du microcontrôleur.

Ce nombre a la particularité de changer de valeur (d'où son appellation de variable).

## Présentation Arduino

Il existe une multitude de nombres : les nombres entiers, les nombres décimaux, ... C'est pour cela qu'il faut assigner une variable à un type de variable.

*Rappeler la valeur max possible en décimal sur 8 bits :*

*Donner la valeur max possible en décimal sur 16 bits :*

Donner la valeur max possible en décimal sur 32 bits :

*Donner la signification de float :*

*Donner le nom entier des abréviations char et int :*

*Compléter le tableau suivant pour les différents types de variables :*

Type de variable	Taille en mémoire (8 ou 16 ou 32 bits)	Type de nombre	Valeurs min/max
boolean		binaire	
char		entier	
unsigned char ou byte		entier	
int		entier	
unsigned int ou word		entier	
long		entier	
unsigned long		entier	
float		réel	

Si une variable ne prend jamais une valeur négative alors on utilisera un type **non-signé**. On repère ces types par le mot **unsigned** (de l'anglais : non-signé).

*Pour se simplifier pourquoi ne pas mettre toutes les variables en type "long" ?*