

## Investigation : l'environnement Arduino

Nom :

Prénom :

Classe

### Mes consignes de travail

1. Je me mets en groupe de 5 élèves ;
2. Je travaille sur un thème tiré au sort :
  1. Je rédige une synthèse que je présenterai aux autres élèves. Je la donne au professeur pour qu'elle soit notée ;
  2. Je rédige deux questions qui seront insérées dans un quiz.
3. Je me regroupe avec 4 autres élèves qui portent un autre thème :
  1. Je prends connaissance des autres synthèses et je les note ;
  2. Je schématise les différents éléments d'un micro-contrôleur ;

4. Je dessine une carte Arduino ;
5. Je légende mon dessin pour identifier les différents éléments qui composent la carte ;
6. Au cours de la semaine, je réponds au quiz de la classe.

## Investigation : l'environnement Arduino

### Sources

Ordre des couleurs d'une résistance électrique, <https://jeretiens.net/ordre-des-couleurs-dune-resistance-electrique/>, Sam, 5/08/2013 consulté le 11/11/2016

Le code des couleurs des résistances, <http://www.positron-libre.com/>, consulté le 11/11/2016

Programmez vos premiers montages avec Arduino, Jean-Noël Rousseau, <https://openclassrooms.com/>, consulté le 11/11/2016

Arduino, Collectif, 22/12/2011, <https://www.flossmanualsfr.net/>

Arduino pour bien commencer en électronique et en programmation, Astalaseven, Eskimon et olyte, [www.lesiteduzero.com](http://www.lesiteduzero.com), 4/08/2012

Atelier Arduino, Jean Noël Montagné, Centre de ressources art sensitif, 11/2006

# Investigation : l'environnement Arduino

## Groupe Microcontrôleur 1/2

La carte Arduino repose sur un circuit intégré ou micro-contrôleur, c'est à dire, un petit processeur informatique relié à des entrées et des sorties numériques (0 ou 1) ou analogiques (tension variable). Il est capable de mémoriser et d'exécuter un programme visant à interpréter les entrées pour agir sur les sorties. Il se programme en général à l'aide d'un ordinateur mais peut fonctionner de manière autonome. En milieu industriel, les automates programmables qui servent à gérer et piloter des machines en sont une illustration.

Le microcontrôleur comporte notamment :

### L'unité centrale (CPU : Central Processing Unit)

Sa fonction principale consiste à décoder et à exécuter les commandes. Elle peut adresser des mémoires, gérer des entrées ou des sorties et réagir à des interruptions.

### Le bus de données

Il est constitué d'un ensemble de fils conducteurs qui véhiculent chacun des données logiques. Il sert à transporter les données numériques d'un bloc à un autre.

### Le contrôleur d'interruption

Puisque le processeur ne peut pas traiter plusieurs informations simultanément, un programme en cours d'exécution peut, grâce à une interruption être momentanément suspendu, le temps que s'exécute une routine d'interruption. Le programme interrompu peut ensuite reprendre son exécution.

Les périphériques ont un numéro d'interruption, que l'on appelle IRQ (Interrupt request, ce qui signifie «requête d'interruption»). A titre d'image, chaque périphérique tire une «ficelle» reliée à une cloche pour signaler à l'UC qu'il veut qu'il prête attention.

### L'horloge interne

Elle émet un signal périodique sur lequel est synchronisée chaque opération du microcontrôleur. Dans l'ATmega328, la fréquence du signal d'horloge est de 16 MHz.

## Investigation : l'environnement Arduino

1. Je note la synthèse que je partagerai avec mon groupe.

[illegible]

2. Je note 2 questions (QCM) pour le quiz de la semaine.

[illegible]

## Groupe Microcontrôleur 2/2

La carte Arduino repose sur un circuit intégré ou micro-contrôleur, c'est à dire, un petit processeur informatique relié à des entrées et des sorties numériques (0 ou 1) ou analogiques (tension variable). Il est capable de mémoriser et d'exécuter un programme visant à interpréter les entrées pour agir sur les sorties. Il se programme en général à l'aide d'un ordinateur mais peut fonctionner de manière autonome. En milieu industriel, les automates programmables qui servent à gérer et piloter des machines en sont une illustration.

Le micro-contrôleur comporte notamment :

### Les zones de mémoire

Il existe deux types de mémoires dans un microcontrôleur :

La mémoire de programme qui accueille le programme que le CPU doit exploiter. Elle ne doit pas s'effacer lors d'une coupure de courant. Elle peut toute fois être effacée ou écrasée mais à partir d'un signal électrique spécifique. Elle se compose d'une mémoire Electrically-Erasable Programmable Read-Only Memory (EEPROM) de 1 ko pour le système d'exploitation et d'une mémoire flash de 32 ko pour le stockage du programme.

La mémoire de données (SRAM) pour gérer les résultats de calcul du moment. Elle peut donc s'effacer lors d'une coupure de courant. On attend surtout d'elle que son temps d'accès soit rapide.

### Les ports d'entrées-sorties

Ils relient le microcontrôleur au monde extérieur. Ils constituent une interface à laquelle les périphériques peuvent être raccordés.

Les ports d'entrées recueillent les informations qui proviennent de l'extérieur (consignes de l'utilisateur, mesures de capteurs, ...).

Les ports de sortie écrivent les informations vers l'extérieur (affichage de message, commande de moteur, ...).

## Investigation : l'environnement Arduino

1. Je représente un tableau de synthèse pour les mémoires présentent dans un micro-contrôleur.

2. Je schématise le cycle du micro-contrôleur.

3. Je note la synthèse que je partagerai avec mon groupe.

---

---

---

---

---

---

---

4. Je note 2 questions (QCM) pour le quiz de la semaine.

---

---

---

---

---

---

---

---

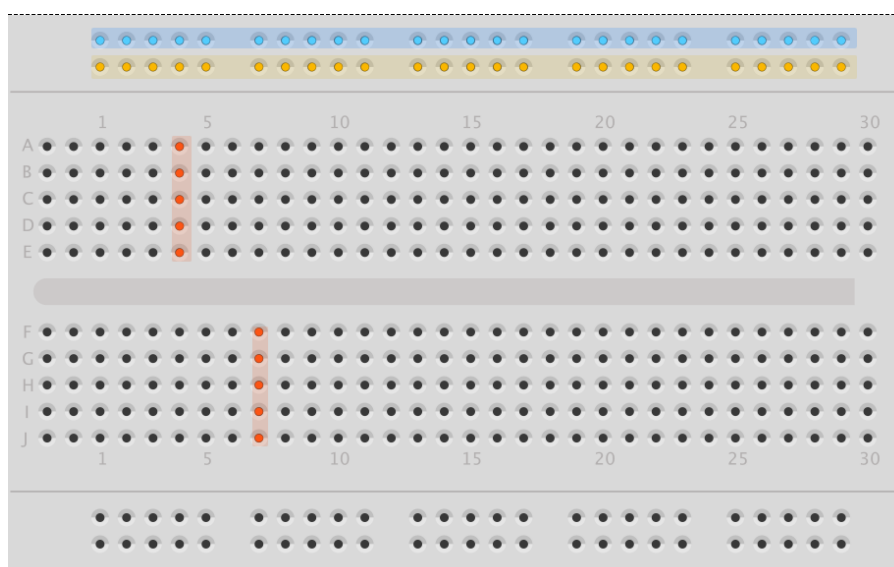
---

---

# Investigation : l'environnement Arduino

## La plaque d'essai ou breadboard

Quand vous devez réaliser un projet avec Arduino, la première étape consiste à réaliser son montage sur une platine d'essai aussi appelée breadboard. Une platine d'essai est un support qui comporte des petits trous dans lesquels vous allez pouvoir positionner vos composants ainsi que des fils qui vous permettront de réaliser votre circuit électrique. À l'intérieur de votre platine, il faut imaginer qu'il y a des chemins déjà existants que nous pouvons voir sur l'image ci-dessous.

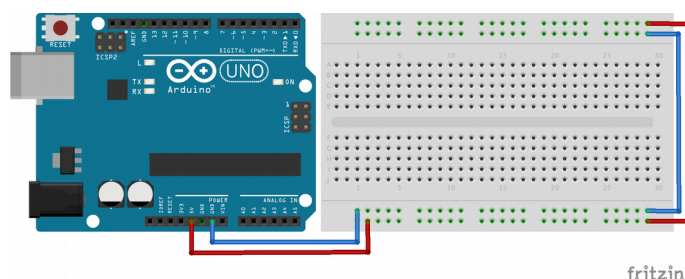


fritzing

Il y a donc les trous en haut et en bas, chaque ligne (ici bleue et jaune) montre des trous liés entre eux. Puis au centre, les 5 trous de chaque colonne (en rouge) sont liés. "Liés", ça signifie que si on connecte un composant dans un trou et un autre dans un trou lié, ils sont connectés entre eux.

Les deux grandes lignes servent souvent à relier les Vin (+5 V) et les grounds (Gnd) entre eux.

Du coup tout ce qui sera connecté aux trous liés à +5 V recevra du +5 V et tout ce qui sera connecté aux trous liés à Gnd ira vers le 0 V.



## Investigation : l'environnement Arduino

1. Je note la synthèse que je partagerai avec mon groupe.

---

---

---

---

---

---

---

---

---

---

2. Je note 2 questions (QCM) pour le quiz de la semaine.

---

---

---

---

---

---

---

---

---

---



# Investigation : l'environnement Arduino

## Groupe Environnement de programmation

L'installation de l'interface de programmation Arduino est relativement simple et possible sur les plates-formes Windows, Mac OS X et Linux. L'environnement de programmation Arduino est écrit en Java et l'interface est inspirée de Processing, un compilateur avr-gcc (pour le processeur du micro-contrôleur) ainsi que d'autres logiciels libres. Puisque Arduino s'appuie sur Java, il est nécessaire que la machine virtuelle Java soit installée sur votre système d'exploitation.

L'environnement de développement est disponible sur <https://www.arduino.cc/>

Le programmeur Arduino va utiliser un langage évolué pour rédiger son sketch (programme). Ce langage est basé sur les langages C/C++.

En revanche, le microcontrôleur ne connaît à son niveau d'interprétation que le langage machine (code natif) composé exclusivement de valeurs numériques. C'est pourquoi un environnement de développement traduisant dans un langage évolué a été créé.

Il faut alors une sorte de traducteur servant de lien entre le code natif et le langage évolué. C'est le rôle du compilateur.

Chaque programme Arduino comporte 3 parties distinctes :

1. La déclaration des variables en début de programme. Cette partie du programme est exécutée une (1) fois au lancement du programme ;
2. L'initialisation des entrées/sorties dans la fonction setup(). Cette partie du programme est exécutée une (1) fois au lancement du programme ;
3. Les instructions du programme dans la fonction loop(). Cette partie du programme est exécutée indéfiniment tant que la carte est alimentée.

Une fois le programme écrit, il faut le téléverser dans le micro-contrôleur de la carte Arduino. L'opération de téléversement s'effectue en deux étapes, la compilation puis le téléversement à proprement parlé.

En revanche l'opération inverse (récupérer un programme sur un micro-contrôleur) est impossible. Il est donc important de sauvegarder son travail avant le téléversement.

## Investigation : l'environnement Arduino

1. Je schématise le passage du langage évolué au langage machine.

2. Je note la synthèse que je partagerai avec mon groupe.

---

---

---

---

---

---

---

---

---

---

3. Je note 2 questions (QCM) pour le quiz de la semaine.

---

---

---

---

---

---

---

---

---

---

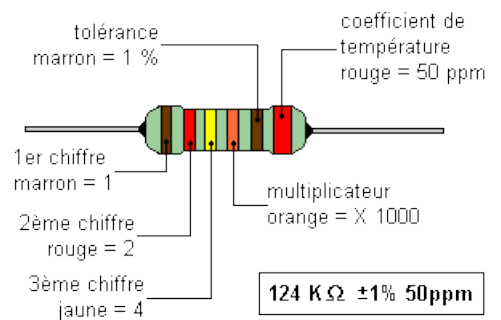
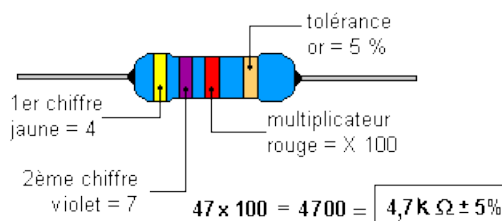
# Investigation : l'environnement Arduino

## Groupe Résistances et Masse

Les résistances sont utilisées dans de nombreux cas, pour réduire une tension, pour provoquer un courant, ou associées à d'autres composants pour des circuits plus complexes.

En électronique, la valeur d'une résistance est codée par des anneaux de couleurs. Pour les résistances à 4 anneaux, nous avons trois couleurs pour exprimer la valeur de la résistance et une couleur pour indiquer la tolérance.

Pour connaître la valeur de la résistance, positionner la résistance de façon à avoir l'anneau le plus à l'extérieur du corps de la résistance sur sa gauche (ou mettre la couleur dorée ou argentée sur sa droite) et suivre les exemples imagés suivant.



La phrase « Ne Mangez Rien Ou Jeûnez Voilà Bien Votre Grande Bêtise » est un moyen mnémotechnique pour se souvenir de l'ordre des couleurs et de leur valeur.

## La Masse

La masse, c'est un référentiel. En électronique on voit la masse d'un montage comme étant le zéro Volt (0V). C'est le point qui permet de mesurer une bonne partie des tensions présentes dans un montage. Elle est représentée par « GND » qui signifie « ground » c'est à dire sol/terre en anglais sur la carte Arduino.

## Investigation : l'environnement Arduino

1. Je calcule la résistance en ohms des exemples suivants :

R1		R2		R3	
Argent		Vert		Noir	
Rouge		blanc		Marron	
Rouge		jaune		Rouge	
Jaune		Or		Argent	

2. Je note la synthèse que je partagerai avec mon groupe.

---

---

---

---

---

---

---

---

---

---

3. Je note 2 questions (QCM) pour le quiz de la semaine.

---

---

---

---

---

---

---