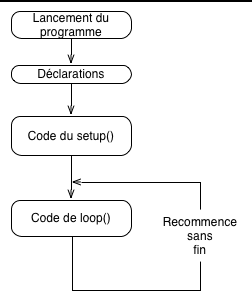
***Petit résumé de l’arduino***

***Important : les explications peuvent être inexactes et mêmes fausses, je ne suis responsable de rien, si vous trouver des fautes, ou si vous avez d’autres choses à ajouter veuillez me contacter s’il vous plait.***

**Definitions:**

* le « programme » arduino est composé de deux fonctions principales: setup et loop , la première contient les instructions à exécuter une seule fois au début , et la deuxième contient les instructions qui seront exécuter infiniment tant que l’arduino est sous tension.

****

* l’arduino est composé de 13 pins, chaque pin peut être configuré pour être utilisé en tant que sortie ou entrée , les pins 0 et 1 sont un peu différent , ils ne doivent pas être utilisé si on veut utiliser le moniteur (ça veut dire si on ne veut écrire dans la console de l’IDE de l’arduino sur votre PC)

**Les modes des pins**

* Il y a deux modes possibles pour chaque pin : INPUT(entrée) ou OUTPUT(sortie), un pin ne peut prendre qu’un seul de ces modes à la fois

**Les états des pins en mode entrée/sortie**

* Un pin en mode sortie(écriture) peut soit émettre un signal électrique (être en état HIGH) ou ne rien émettre (être en état LOW)
* Un pin en mode entrée (lecture) peut soit recevoir un signal électrique (HIGH) soit ne rien recevoir (LOW)

**Le branchement des dipôles :**

* On branche l’anode (+) avec le générateur (pin ou sortie 5v) et le (-) avec la masse (gnd)
* Il faut choisir la résistance convenable pour ne pas griller les composants (pour un LED il faut une résistance entre 100 Ω et 1000 Ω), cette résistance peut être brancher sur le pole positif ou négatif.

**Les instructions :**

**Initialisation de pin (choix du mode : entrée ou sortie)**

* **pinMode**(PIN\_NO, mode)
  + PIN\_NO : numéro de pin (de 1 à 13).
  + Utilisation : soit entrée (**INPUT**) soit sortie (**OUTPUT**).

Il y a aussi un mode **INPUT\_PULLUP** qui connecte une résistance 20K integré, ce mode permet de lire le signal à l’entrée d’un pin sans devoir ajouter une résistance **PULL UP** ou **PULL DOWN** (expliqué ci-dessous), attention ce mode ne peut pas être utilisé sur le pin 13 …

**Lecture/écriture de « signal électrique »**

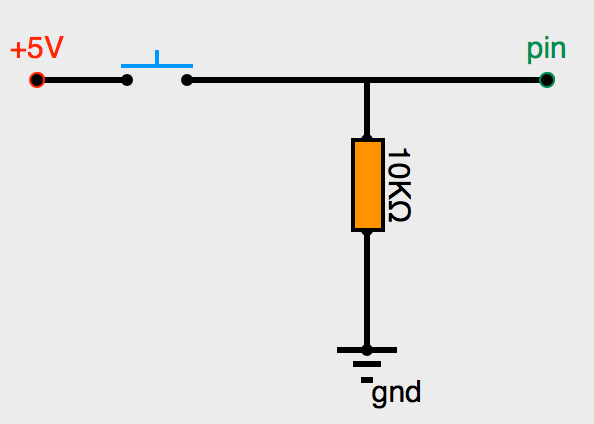
* **digitalWrite**(PIN\_NO,état)
  + digitalWrite peut être utilisé seulement sur les pins en mode sortie (OUTPUT).
  + PIN\_NO : le numéro de pin où on veut émettre (ou non) un signal électrique.
  + état : HIGH pour émettre un signal électrique (~5V) ou LOW (~0V).
* **digitalRead**(pin\_no)
  + digitalRead peut être utilisé seulement sur les pins en mode entrée (INPUT).
  + PIN\_NO : le numéro de pin d’où on veut lire le signal électrique.
  + digitalRead renvoie HIGH s’il y a un courant électrique qui passe sur le pin demandé et LOW sinon.
* **delay**( durée\_en\_ms )
  + Permet de « mettre en pause » le programme pour une durée précise en milliseconde.

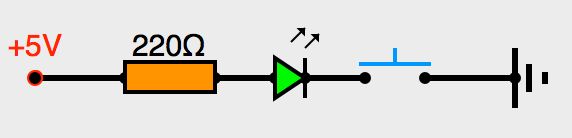
**Ecriture dans le moniteur série (Sérial):**

* Serial est la console dans votre IDE d’Arduino, elle permet à votre Arduino de communiquer avec votre ordinateur.
* Si on veut utiliser le moniteur série, on doit laisser les pins 0 et 1 vide.
* **Serial.begin**(vitesse): permet de définir la vitesse de communication entre l’ordinateur et l’arduino , la vitesse est en baud i.e nombre de caractère (et non pas de bit mais de 8 bits, dans le cas de l’arduino) par seconde, la vitesse habituellement utilisée est 9600 baud.
* **Serial.println**(argument) : permet d’afficher « argument » (un text ou une variable …) sur le moniteur série
  + println ajoute automatiquement un retour à la ligne à la fin , print ne l’ajoute pas
* **random(min,max) (resp random(max))** permettent de générer un nombre pseudo aléatoire compris entre min et max (resp 0 et max), avec max exclut et min (resp 0) inclut.
  + Afin d’éviter d’avoir les mêmes séries de nombre on ajoute dans setup **randomSeed(analogRead(0))**

pour résoudre le problème des lectures aléatoires sur les pin en mode INPUT il faut ajouter soit :

Résistance **PULL DOWN** (résistance 10K entre le pin et le ground ) :



Résistance **PULL UP** (résistance 10K entre le pin et la source de tension) 

Source :

* <https://openclassrooms.com/courses/programmez-vos-premiers-montages-avec-arduino/>

**continuer :**

<https://openclassrooms.com/courses/programmez-vos-premiers-montages-avec-arduino/utilisez-les-potentiometres-les-entrees-analogiques-et-la-fonction-de-mappage>

**à voir : (électricité)**

http://www.rcva.fr/