

Tutoriel Arduino 1 : Sortie

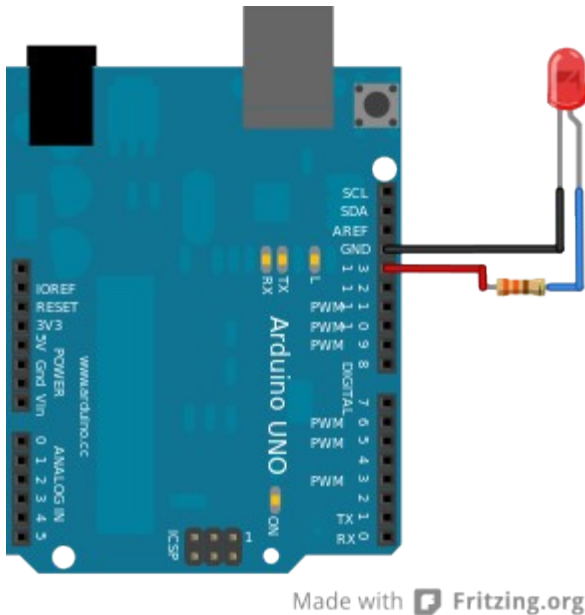
Objectif

L'objectif de ce premier tutoriel est de contrôler l'allumage d'une Diode Électro-Luminescente (DEL ou LED en anglais) afin de faire clignoter celle-ci : une seconde allumée, une seconde éteinte.

Schéma électrique

Le schéma électrique est très simple et consiste en :

- 1 carte Arduino
- 1 DEL
- 1 résistance de 330 Ω (Ohms)



Le principe d'une DEL est de s'allumer lorsque le courant passe à travers celle-ci. Pour cela il suffit de connecter l'une de ses broches à la borne \oplus d'une pile, c'est l'anode et l'autre broche à la borne \ominus , c'est la cathode. Attention de bien identifier chacune des broches pour ne pas monter la diode à l'envers car elle ne s'allumerait pas. L'identification est simple puisque l'anode est la broche la plus longue et la cathode la broche la plus courte. Un méplat peut aussi avoir été fait sur le corps de la DEL pour indiquer la cathode.

La résistance est utilisée pour limiter le courant électrique qui traversera la DEL. En effet la tension aux bornes d'une DEL est généralement d'environ 1,5 V alors que la tension en sortie de l'Arduino est de 5 V. Bien que brancher directement la DEL sur la carte soit possible il est recommandé d'ajouter une résistance pour éviter d'abîmer la DEL. Le schéma ci-contre montre une résistance montée en série avec la DEL : la résistance à la suite de la DEL.

La tension aux bornes de la résistance est calculée simplement, c'est la différence entre la tension de la carte et celle de la DEL, soit : $5\text{ V} - 1,5\text{ V} = 3,5\text{ V}$. La valeur de la résistance est ensuite calculée afin de limiter le courant passant à travers la DEL à environ 10 mA. La loi physique permettant de relier l'intensité du courant électrique à la tension passant à travers la résistance ou la DEL est ce qu'on appelle la loi d'ohm. Elle définit la relation mathématique suivante : $U = R \times I$ où U est la tension, R la valeur de la résistance et I , l'intensité du courant. Dans notre cas on connaît déjà la tension $U = 3,5\text{ V}$ et l'on sait que l'on veut une intensité d'environ 10 mA. On en déduit donc la résistance $R = U / I = 3,5\text{ V} / 10\text{ mA} = 350\ \Omega$. Toutes les valeurs des résistances n'étant pas disponibles on choisit celle qui s'en rapproche le plus, dans notre cas 330 Ω .

Pour finir on connecte la cathode de la DEL à la broche GND de la carte Arduino (GROUND, la masse en

français : l'équivalent de la borne \ominus) et la broche de la résistance à la broche 13 de la carte (qui jouera l'équivalent de la borne \oplus .)

Programme Arduino

Maintenant que le schéma électrique est réalisé, il ne reste plus qu'à écrire le programme qui contrôlera la DEL. L'objectif de ce programme est d'allumer la DEL pendant une seconde puis de l'éteindre pendant une seconde de l'allumer encore une seconde puis de l'éteindre et ainsi de suite, en boucle.

Commençons tout d'abord par voir comment contrôler l'allumage de la DEL. Le schéma électrique montre qu'elle est branchée sur la broche 13 de la carte Arduino. Pour allumer la DEL, il faudra donc que cette broche soit l'équivalent de la borne \oplus d'une pile, c'est-à-dire qu'elle soit à 5 V. C'est ce qu'on appellera la position haute ou **HIGH** (haut en anglais.) L'inverse est la position basse ou **LOW** (bas en anglais) pour laquelle la tension de la broche est égale à celle de la masse (GROUND ou GND en anglais.)

Pour changer la position d'une broche on utilise l'instruction ou fonction Arduino `digitalWrite()`. Cette fonction nécessite deux paramètres : le numéro de la broche (13) et la position (**HIGH** ou **LOW**.) Pour allumer la DEL on écrira donc : `digitalWrite(13, HIGH)` tandis que pour l'éteindre on écrira `digitalWrite(13, LOW)`.

Pour l'éteindre ou l'allumer pendant une seconde on utilise l'instruction ou fonction Arduino `delay()`. Cette fonction est une fonction de délai c'est-à-dire qu'elle ne fait rien pendant un temps donné passé en paramètre. Cette durée est exprimée en millisecondes. Dans notre cas nous souhaitons utiliser un délai d'une seconde soit : `delay(1000)`.

Voici le programme complet :

```
void setup()
{
  pinMode(13, OUTPUT);
}

void loop()
{
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

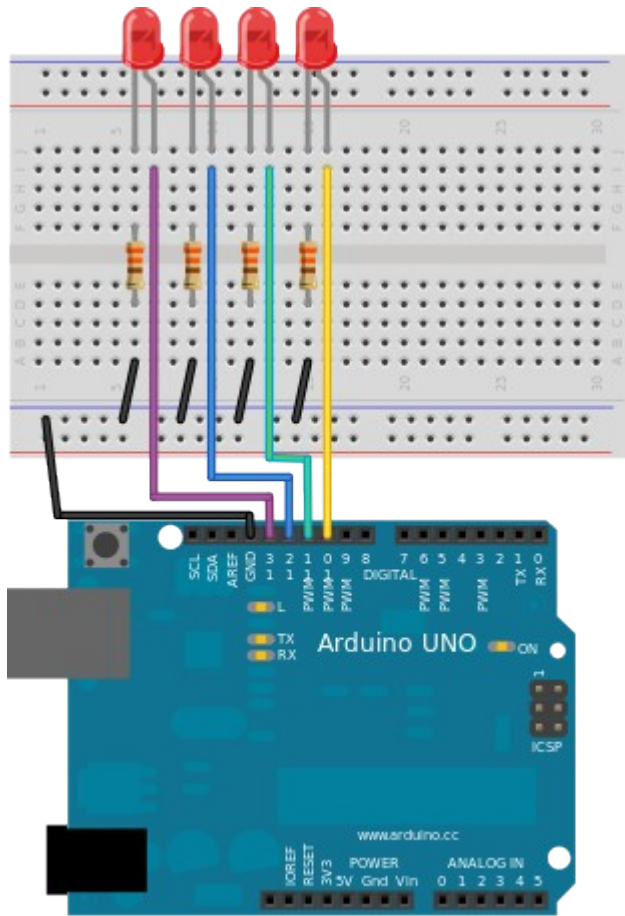
Il est divisé en deux parties. La première initialise (**setup** en anglais) la carte et configure la broche 13 comme étant une sortie ou **OUTPUT** (sortie en anglais) :

```
void setup()
{
  pinMode(13, OUTPUT);
}
```

La deuxième partie est la boucle (**loop** en anglais) principale de notre programme. On y retrouve les instructions pour allumer la DEL puis attendre une seconde puis éteindre la DEL puis attendre encore une seconde. Ces instructions sont répétées en boucle.

Avec un peu plus de diodes

On peut très facilement étendre les programme et schéma électrique précédents pour réaliser un chenillard. Ici un exemple avec 4 DELs :



```
void setup()
{
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);

  digitalWrite(10, LOW);
  digitalWrite(11, LOW);
  digitalWrite(12, LOW);
  digitalWrite(13, LOW);
}

void loop()
{
  digitalWrite(13, LOW);
  digitalWrite(10, HIGH);
  delay(500);
  digitalWrite(10, LOW);
  digitalWrite(11, HIGH);
  delay(500);
  digitalWrite(11, LOW);
  digitalWrite(12, HIGH);
  delay(500);
  digitalWrite(12, LOW);
  digitalWrite(13, HIGH);
  delay(500);
}
```

Made with  Fritzing.org