

Arduino

Emetteur Morse *Base*

Nous aurons besoin de



L'Arduino Uno



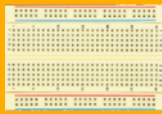
*Le programme
EmetteurMorse*



*L'éditeur de
programme*



L'ordinateur



*La planche
d'essai*



*1 bouton
poussoir*



*1 résistance
220 Ω*

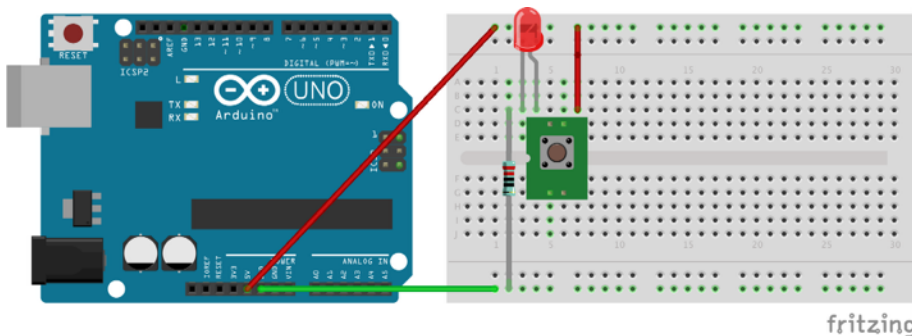


1 LED



*Le cable
USB*

Il faut construire le circuit comme ceci



Le programme est quasiment vide.

La ligne qui commence par `//` est un commentaire. Ce sont des informations pour le programmeur.

- `setup` qui dit ce qu'il faut faire 1 seule fois au démarrage
- `loop` qui dit ce qu'il faut faire de manière répétée à chaque cycle

La LED s'allume lorsqu'on appuie sur le bouton.

Ce que fait la fonction est décrit entre les signes { }. Ces signes s'appellent des **accolades**.

`void loop() { ... }` est aussi une fonction.

Ces fonctions contiennent les instructions que l'on donne à l'Arduino. Ce sont aussi des fonctions mais celles-ci sont décrites dans du code qui sera envoyé avec notre programme. On n'a pas à les écrire. On appelle ce code fourni avec l'Arduino une **bibliothèque de fonctions**.

La LED s'éclaire quand il y a du courant et s'éteint quand il n'y en a plus. Pour la faire clignoter on va envoyer un courant, attendre un peu, puis ne plus envoyer le courant, attendre un peu. Et ensuite on recommence, c'est `loop` qui gèrera ça.

- `pinMode(13, OUTPUT)` indique que l'alimentation est faite par le pin13
- `digitalWrite(13, HIGH)` envoi la quantité maximale de courant sur le pin 13. La LED s'éclaire.
- `delay(1000)` attend 1s (1000 ms)
- `digitalWrite(13, LOW)` envoi la quantité minimale de courant sur le pin 13. La lampe s'éteint.

Les valeurs qui sont entre parenthèses après les fonctions s'appellent des **paramètres**. Par exemple, un paramètre permet d'utiliser `delay` pour attendre plus ou moins longtemps selon la valeur.

- `delay(1000)` attend 1s (1 000 ms)
- `delay(10000)` attend 10s (10 000 ms)