

# Projets Arduino – 1 Clignotement

## MATÉRIEL

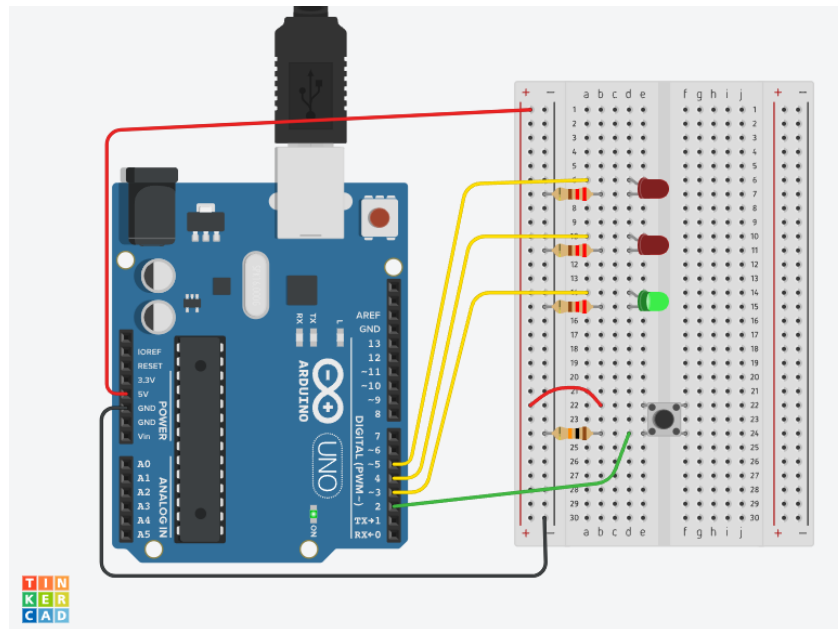
1 interrupteur à bouton-poussoir

3 DEL

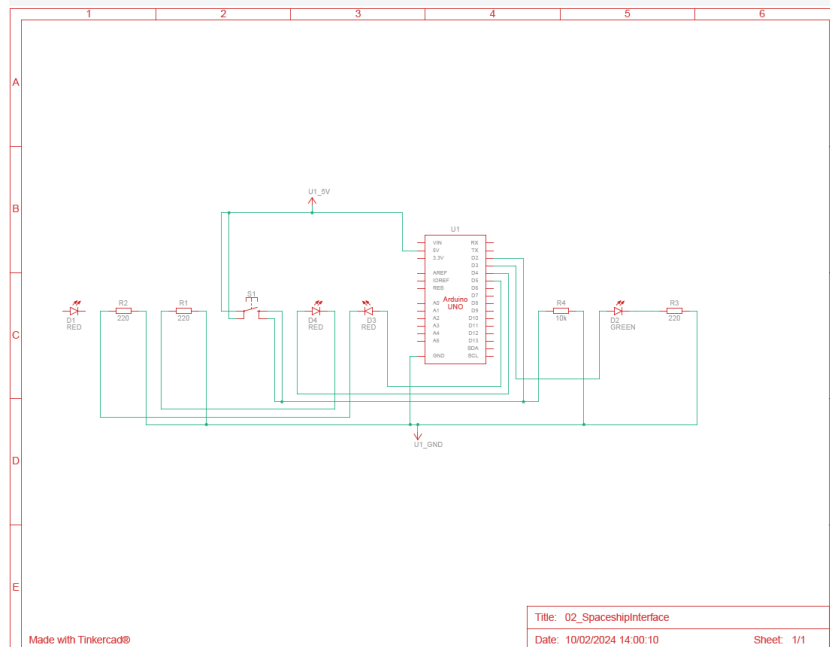
3 résistances de 220 Ω

1 résistance de 10 kΩ

## MONTAGE



## SCHÉMA



## CODE

```
// Projets Arduino – 1 Clignotement
```

```
int switchState = 0;
```

```
/*Déclare une variable au nom de switchState, on indique que sa valeur sera exprimée en nombres entiers (...-3, -2, -1, 0, 1, 2, 3...) et que sa valeur au commencement sera de zéro. Cette variable s'appelle switchState, qui veut dire l'état de l'interrupteur. En programmation il est important de respecter les espaces, la ponctuation, l'orthographe et la casse (majuscule/minuscule). */
```

```
void setup() {
```

```
/*Le code à l'intérieur de cette section donne les indications de départ nécessaires pour l'identification des principales broches utilisées, avant de commencer l'action. Ce code sera exécuté une seule fois au démarrage. Attention à bien écrire tous les symboles de ponctuation ( ){} ;, */
```

```
    pinMode(3, OUTPUT);  
    pinMode(4, OUTPUT);  
    pinMode(5, OUTPUT);  
    pinMode(2, INPUT);  
}
```

```
//Ces lignes indiquent quel est le mode des broches, soit entrée ou sortie d'information.
```

```
void loop () {
```

```
/*Le code à l'intérieur de cette section se fera en boucle et sera répétée indéfiniment. C'est le cœur du programme où les instructions sont données pour les actions concrètes. */
```

```
    switchState = digitalRead(2);
```

```
/*On spécifie la source d'information de la variable switchState en faisant une lecture digitale de la valeur de la broche 2. */
```

```
if(switchState == LOW){
```

```
/*On a ici une ligne de code de logique conditionnelle, une action sera déclenchée si une condition est remplie et on donne aussi des instructions dans le cas où elle ne serait pas remplie. Cette première ligne nous donne seulement la condition à remplir. Si la variable switchState, qui est branchée sur une broche digitale, nous renvoie une valeur basse (de 0), comme celle qui lui a été attribuée au départ. Ce qu'il faut comprendre c'est que si on n'appuie pas sur l'interrupteur, la valeur lue par la broche digitale est celle qu'on a donnée au départ (0) et qu'en langage de programmation, on le traduit par LOW. Dans ce cas, à la ligne suivante on va donner les indications. */
```

```
digitalWrite(3, HIGH);
```

/\*Si la condition de la ligne précédente est remplie, on fait une **écriture digitale** (sortie d'information) sur la broche 3 de la valeur HIGH, ce qui va faire allumer la DEL. Puisque LOW correspond à 0, HIGH correspond à 1. Dans le contexte digital c'est 0 pour fermé/éteint et 1 pour ouvert/allumé. \*/

```
digitalWrite(4, LOW);  
digitalWrite(5, LOW);  
}
```

/\*Deuxième et troisième actions si la condition est remplie : on écrit sur les broches 4 et 5 la valeur LOW, donc DEL éteinte. Il est primordial de ne pas oublier de fermer l'accolade. \*/

```
else {
```

/\*Dans le cas où la condition ne serait pas remplie, on utilise le terme *else* ou sinon en français. Autrement dit, si la valeur lue par la broche 2 n'est pas de 0, ça veut dire que sa valeur est de 1 et donc ça signifie qu'on appuie sur le bouton. Alors, **si** la condition est remplie, on va faire les 3 actions citées plus haut, **sinon**, on va faire les actions suivantes. \*/

```
digitalWrite(3, LOW);  
digitalWrite(4, LOW);  
digitalWrite(5, HIGH);
```

/\*Les broches 3 et 4 écriront LOW, les lumières qui y sont connectées seront fermées et la broche 5 enverra un signal HIGH, la lumière qui y est branchée sera allumée. \*/

```
delay(250);
```

/\*On donne un délai de 250 millièmes de seconde avant que la prochaine action soit faite. Si on ne donne pas de délai, la rapidité avec laquelle le processeur traite l'information fera en sorte que notre œil ne percevra pas les changements. D'ailleurs, après avoir vérifié le code, on va modifier les valeurs des délais et observer ce qui se passe. \*/

```
digitalWrite(4, HIGH);  
digitalWrite(5, LOW);
```

/\* On est toujours dans le sinon, les actions posées si la condition n'est pas remplie. Après le délai d'un quart de seconde, la DEL de la broche 4 s'allume et celle de la 5 s'éteint. Le signal écrit sur la broche 3 n'est pas changé. \*/

```
delay(250);  
}  
}
```

/\* On impose un autre délai avant que la boucle du sinon recommence afin de s'assurer que la différence soit perceptible. On remarque la présence de 2 accolades. C'est parce

qu'il y a la petite boucle du sinon qui se trouve dans la boucle du programme. Voir le programme dans son entièreté plus bas pour bien comprendre. \*/

**Programme dans son entièreté :**

```
int switchState = 0;

void setup() {
  pinMode(3, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(2, INPUT);
}

void loop () {
  switchState = digitalRead(2);
  if(switchState == LOW){
    digitalWrite(3, HIGH);
    digitalWrite(4, LOW);
    digitalWrite(5, LOW);
  }

  else {
    digitalWrite(3, LOW);
    digitalWrite(4, LOW);
    digitalWrite(5, HIGH);
    delay(250);
    digitalWrite(4, HIGH);
    digitalWrite(5, LOW);
    delay(250);
  }
}
```

**Tiré de : Fitzgerald, Scott et Michael Shiloh (directors) / The Arduino projects book. 2015**