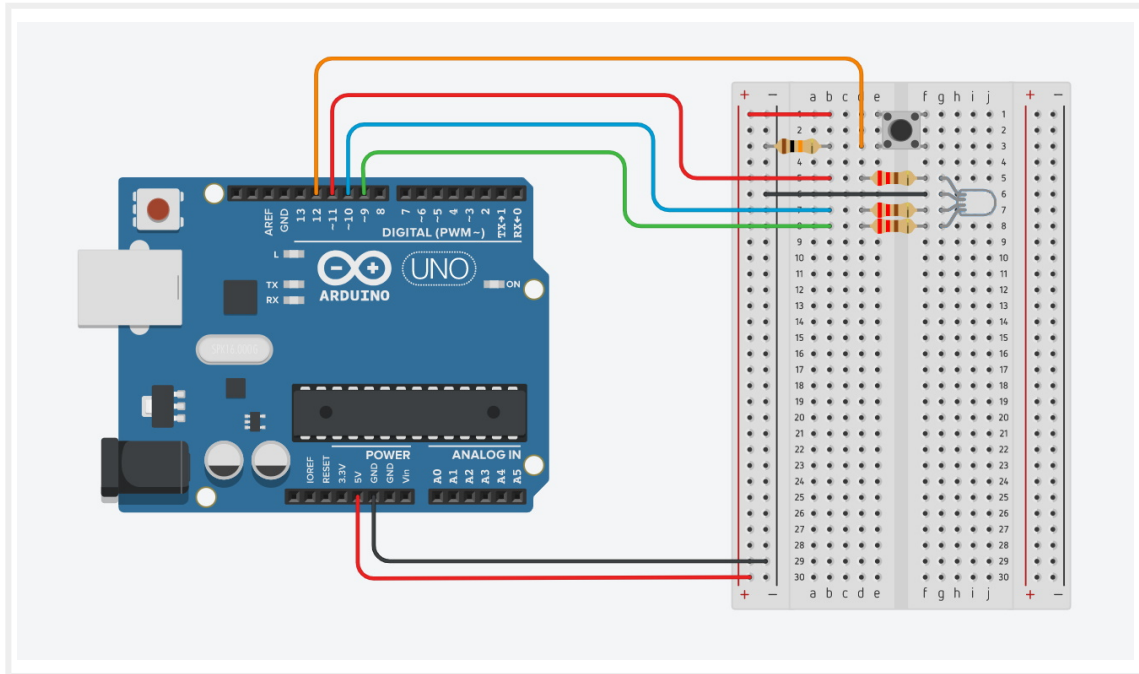


DEL RVB – Brightness

(Contrôler la luminosité d'une DEL avec un bouton-poussoir)



Liste des composants :

- . 1 DEL RVB
- . 3 résistances de 220 Ω (résistances des DELs)
- . 1 bouton poussoir
- . 1 résistance de 10 k Ω (résistance du bouton poussoir)
- . 1 plaque d'essai
- . Fils de connexion

Objectif

Cette activité a pour objectif l'apprentissage de l'utilisation des sorties analogiques de l'Arduino UNO.

Contrairement aux sorties numériques qui ne peuvent avoir que deux valeurs 0 ou 1 (0 ou 5V), une sortie analogique (ou plutôt **PWM**) permet d'obtenir une tension entre 0 et 5 V. les broches 3, 5, 6, 9, 10 et 11 peuvent être configurés en sortie analogique.

C'est pour cela que les anodes de la DEL RVB (à cathode commune) de notre circuit sont connectées sur les broches :

- 9 pour la DEL rouge
- 11 pour la DEL verte
- 10 pour la DEL Bleue

Nous allons utiliser ces sorties pour alimenter une DEL (DEL rouge, verte ou bleue de la DEL RVB) et faire varier sa luminosité suivant ce principe de fonctionnement :

- La DEL étant éteinte, si on appuie sur le bouton poussoir, la diode s'allume.
- On règle la luminosité de la DEL en maintenant le bouton poussoir appuyé, du moins au plus lumineux (broche de la DEL à +0V) jusqu'à un maximum (broche de la DEL à +5V).
- Quand le maximum de la luminosité est atteint et que le bouton poussoir est maintenu appuyé, la luminosité revient au minimum (broche de la DEL à 0V).
- La DEL étant allumée, si on appuie sur le bouton poussoir, elle s'éteint.

Le code pourra être modifié pour voir l'influence des variables (choix de la DEL).

. Rappel

Une DEL RVB (RVB pour Rouge-Vert-Bleu), n'est rien d'autre qu'un ensemble de 3 DELs, une rouge une verte et une bleue rassemblées dans un seul et même boîtier.

Une DEL RVB a 4 broches : une commune à l'ensemble des DELs et une pour chaque couleur de la DEL. La broche commune pourra, selon les modèles, être le + (anode commune) ou le - (cathode commune).

La DEL ci-dessous a pour broche commune la cathode :



. Le programme

```
// Déclaration des constantes et variables

const int PinLED = 11;
const int PinButton = 12;

int ValButton = 0;
int OldValButton = 0;
int State = 0;
int Brightness = 0;
unsigned long StartTime = 0;
unsigned long DeltaTime = 0;

// Initialisation des entrées et sorties

void setup() {
    pinMode (PinButton, INPUT);
}

// Fonction principale en boucle

void loop() {
    ValButton = digitalRead(PinButton);
    delay(10);
    if ((ValButton == HIGH) && (OldValButton == LOW)) {
        State=1-State;
        StartTime= millis();
    }
    if ((ValButton == HIGH) && (OldValButton == HIGH)) {
        DeltaTime = millis() - StartTime;
        if (State == 1 && DeltaTime > 500) {
            Brightness = Brightness + 1;
            delay(10);
            if (Brightness > 255) {
                Brightness = 0;
            }
        }
    }
    OldValButton = ValButton;
    if (State == 1) {
        analogWrite(PinLED, Brightness);
    }
    else {
        analogWrite(PinLED, 0);
        Brightness = 0;
    }
}
```

Déroulement du programme :

Déclaration des constantes et variables

- N° de la broche correspondant à la DEL: **const int PinLED = 11**
- N° de la broche correspondant au bouton poussoir: **const int PinButton = 12**
- Variable pour stocker la valeur de la broche du bouton poussoir: **int ValButton = 0**
- Variable pour stocker l'ancienne valeur de la broche du bouton poussoir: **int OldValButton = 0**
- Variable correspondant à la valeur de la tension appliquée à la broche de la DEL:
int Brightness = 0 (la luminosité de la Del est donc liée à cette variable)
- Variable correspondant à l'action à effectuer: **int State = 0**
- Variables pour mesurer le temps d'appui sur le bouton poussoir: **unsigned long StartTime = 0**
et **unsigned long DeltaTime = 0**

void setup()

initialisation des entrées et sorties

- La broche du bouton poussoir est initialisée comme une entrée digitale. Des données seront donc envoyées depuis cette broche vers le microcontrôleur :

pinMode (PinButton, INPUT)

void loop()

Fonction principale en boucle

Lecture de l'état logique de la broche du bouton poussoir:

ValButton = digitalRead(PinButton)

Attente (10 ms) pour éviter les perturbations dues au phénomène de rebond du bouton poussoir:

delay(10)

bouton poussoir appuyé (broche à +5V):
ValButton = HIGH
et avant relâché (broche à 0V):
OldValButton = LOW

?

Oui

- Mise à jour de la variable correspondant à l'action à effectuer: **State=1-State**
la variable ne peut prendre que 2 valeurs (0 ou 1):
- **State = 1** (Allumer la DEL)
- **State = 0** (Eteindre la DEL)
- Mise à jour de la variable pour stocker le début d'appui sur le bouton poussoir:
StartTime = millis()

bouton poussoir maintenu appuyé
(broche à +5V):
ValButton = HIGH
et **OldValButton = HIGH**

?

Oui

- Mesure du temps d'appui sur le bouton poussoir:
DeltaTime = millis() - StartTime
- Si la DEL est allumée (**State=1**) et **DeltaTime > 500 ms**, la variable de la luminosité est incrémentée tant que le bouton est maintenu appuyé jusqu'à 255:
Si **Brightness > 255** alors **Brightness = 0**

Stockage de l'état logique de la broche du bouton poussoir lu:

OldValButton = ValButton

State = 1 ?

Oui

Non

Allumer la DEL ou régler la luminosité

analogWrite(PinLED, Brightness)

Eteindre la DEL

analogWrite(PinLED, 0)