



# Contrôle d'intensité- mBlock

## 1. Objectifs :

- Contrôler l'intensité de la lumière émise par une ampoule DEL.
- Ajouter un commentaire dans un programme.
- Utiliser une boucle for.
- Utiliser un opérateur de contrôle.

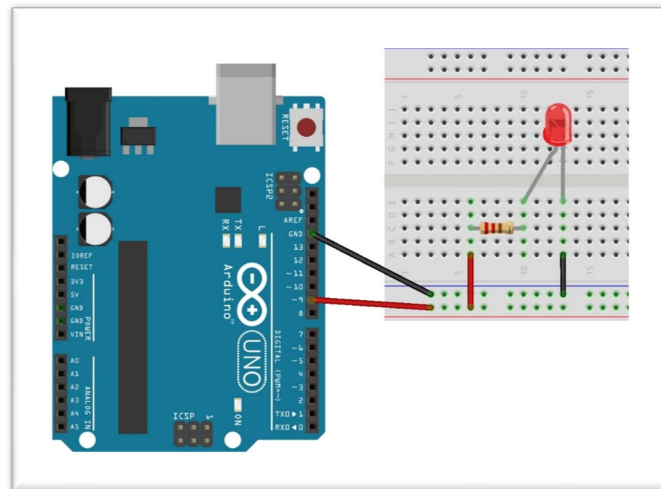
## 2. Ce que fait le programme

Ce programme permet de moduler l'intensité de la lumière émise par une DEL. L'intensité passera de nulle à maximum à plusieurs reprises.

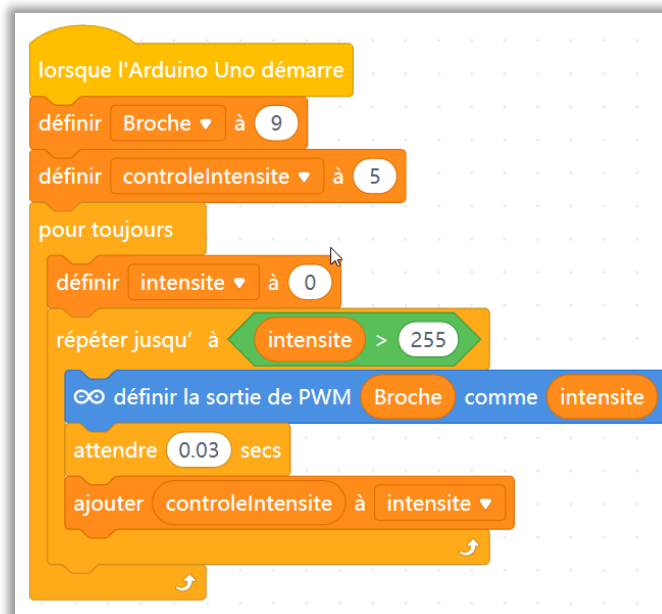
## 3. Étapes de réalisation

1. Brancher la carte Arduino UNO dans l'ordinateur avec le fil approprié. La DEL (verte) d'alimentation de la carte devrait être allumée.
2. Ouvrir le logiciel mBlock et se connecter à la carte Arduino.
3. Écrire le code trouvé plus bas. Il faudra créer des variables dans la section de blocs «Variables » en cliquant sur *Créer une variable*. Les blocs portant les noms des différentes variables apparaîtront alors dans cette section. On peut utiliser ses blocs en les entrant dans les sections circulaires/elliptiques des autres blocs.
4. Réaliser le circuit.
5. Téléverser le code et observer le résultat !

#### 4. Schéma du circuit



#### 5. Code



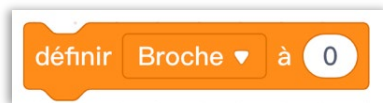
## 6. Blocs utilisés

Il est intéressant de noter que la forme des blocs et des emplacements dans ces derniers varie selon le type. La couleur aussi signifie quelque chose. Chaque catégorie de blocs a une couleur différente. Cela permet de les retrouver facilement.

Par exemple, les emplacements de forme cercle/ellipse représentent des valeurs numériques. Les emplacements en forme rectangulaire avec des bouts en triangle représentent des valeurs logiques ou booléennes (VRAI ou FAUX).

Les blocs de départ ont un demi-cercle sur le dessus et une « coche » permettant d'emboîter un autre bloc en-dessous. Le début et la fin des boucles est indiquée par la forme et on peut glisser des blocs à l'intérieur. Ces blocs seront exécutés en séquence un certain nombre de fois selon le type de boucle. Les blocs de fin qui terminent les exécutions ne permettent pas l'ajout d'un autre bloc en-dessous.

Les différentes couleurs représentent les différentes catégories de blocs selon leur rôle (événement, contrôle, opérateurs, ...).



Ce bloc permet de définir la variable sélectionnée dans la liste déroulante et déjà créée dans la section *Variables* des blocs. Ici, la variable *Broche* a été sélectionnée. Toutes les variables doivent être définies. On utilise ce bloc pour donner une autre valeur à une variable. On sélectionne la variable et on modifie la valeur.

À noter : au lieu d'entrer une valeur numérique directement au clavier, on peut utiliser une variable créée pour remplir l'emplacement, ou encore un autre bloc. La forme du bloc ou de la variable doit correspondre à la forme de l'emplacement (cercle/ellipse ici).



Ce bloc se trouve dans la section *Opérateurs*. Il s'agit d'un opérateur logique. Si la valeur dans le premier cercle/ellipse est plus petite que ou égale à 50, alors la valeur retournée par l'opérateur est FAUX. Si la valeur dans le premier cercle/ellipse est plus grande que 50, la valeur retournée est VRAIE. Ce bloc est utilisé ici comme condition pour la boucle.



Le bloc *répéter jusqu'à* est une boucle qui sera répétée jusqu'à ce que la condition que l'on doit entrer comme argument soit vraie. Si la condition n'est jamais vraie, la boucle sera répétée pour toujours, où tant que la carte Arduino est alimentée.



Ce bloc modifie la valeur de la variable *intensite* en lui ajoutant 1. Il correspond donc à :

$$\text{intensite} = \text{intensite} + 1$$

On peut utiliser ce type de bloc pour additionner des valeurs autres que 1 : des valeurs décimales et négatives (soustraction) sont aussi possibles.



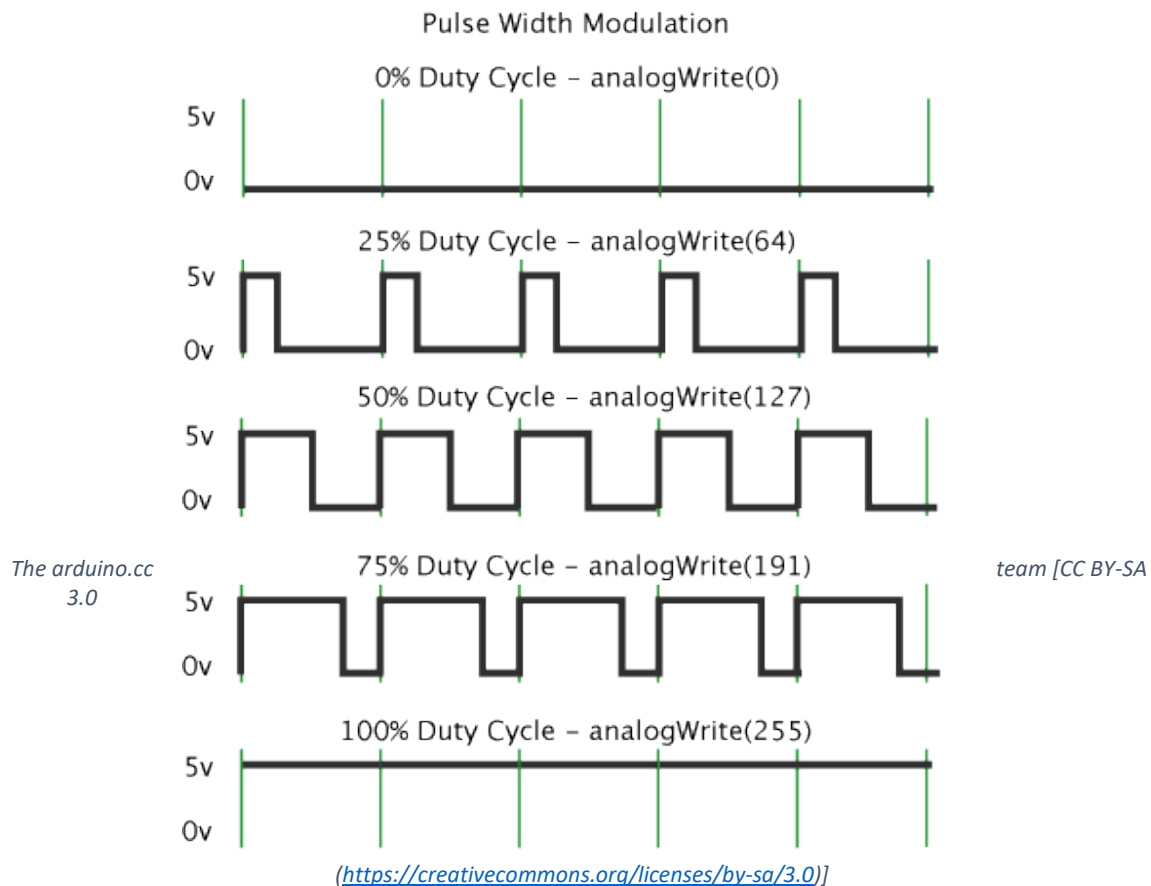
Ce bloc est utilisé pour contrôler l'intensité d'une lumière ou encore la vitesse d'un moteur. Elle ne fonctionne que sur les broches digitales qui ont un symbole « ~ ». Sur une carte de type Arduino UNO, les broches sont 3, 5, 6, 9, 10 et 11. Les broches analogiques A0 à A5 ne le permettent pas.

La valeur est un nombre entier compris entre 0 et 255.

Comme vu précédemment, les broches digitales peuvent envoyer un signal de 5V (HIGH) ou de 0V (LOW). Il n'est pas possible d'envoyer un signal intermédiaire (2.5 V par exemple). Pour réussir à avoir un résultat qui simule l'effet d'une tension intermédiaire, on procède en modulant la largeur d'une impulsion carrée (« Pulse Width Modulation » ou PWM).

La carte Arduino génère une onde carrée à une fréquence d'environ 500 Hz. Cela correspond à 500 cycles par seconde ou encore un cycle de 0,002 s.

La *valeur* détermine la fraction du cycle pendant laquelle la valeur de la tension sera à 5V. La figure de la page suivante montre le principe de base.



Il est à noter que l'œil humain ne peut pas percevoir le clignotement d'une DEL à une fréquence de 500 Hz. On perçoit moins de lumière provenant de la DEL lorsque le temps où la tension est à 5V est plus court.

### Défis :

- Faire passer l'intensité de la lumière de maximum à minimum.
- Modifier le programme pour faire d'autres rythmes de modulation de l'intensité.
- Faire passer l'intensité de la lumière de maximum à minimum et ensuite à maximum en répétition.
- Faire clignoter la lumière en plus d'en moduler l'intensité
- Tout autre chose que vous voudriez essayer!