



Clignotement-- mBlock

1. Objectifs :

- Se familiariser avec une carte de type Arduino UNO
- Réaliser son premier programme Arduino !

2. Matériel :

- Arduino UNO ou Bluno
- Résistance de 220 Ω
- DEL de la couleur désirée
- Ordinateur + fil de branchement pour la carte
- Quelques fils
- Platine d'essai (« breadboard »)

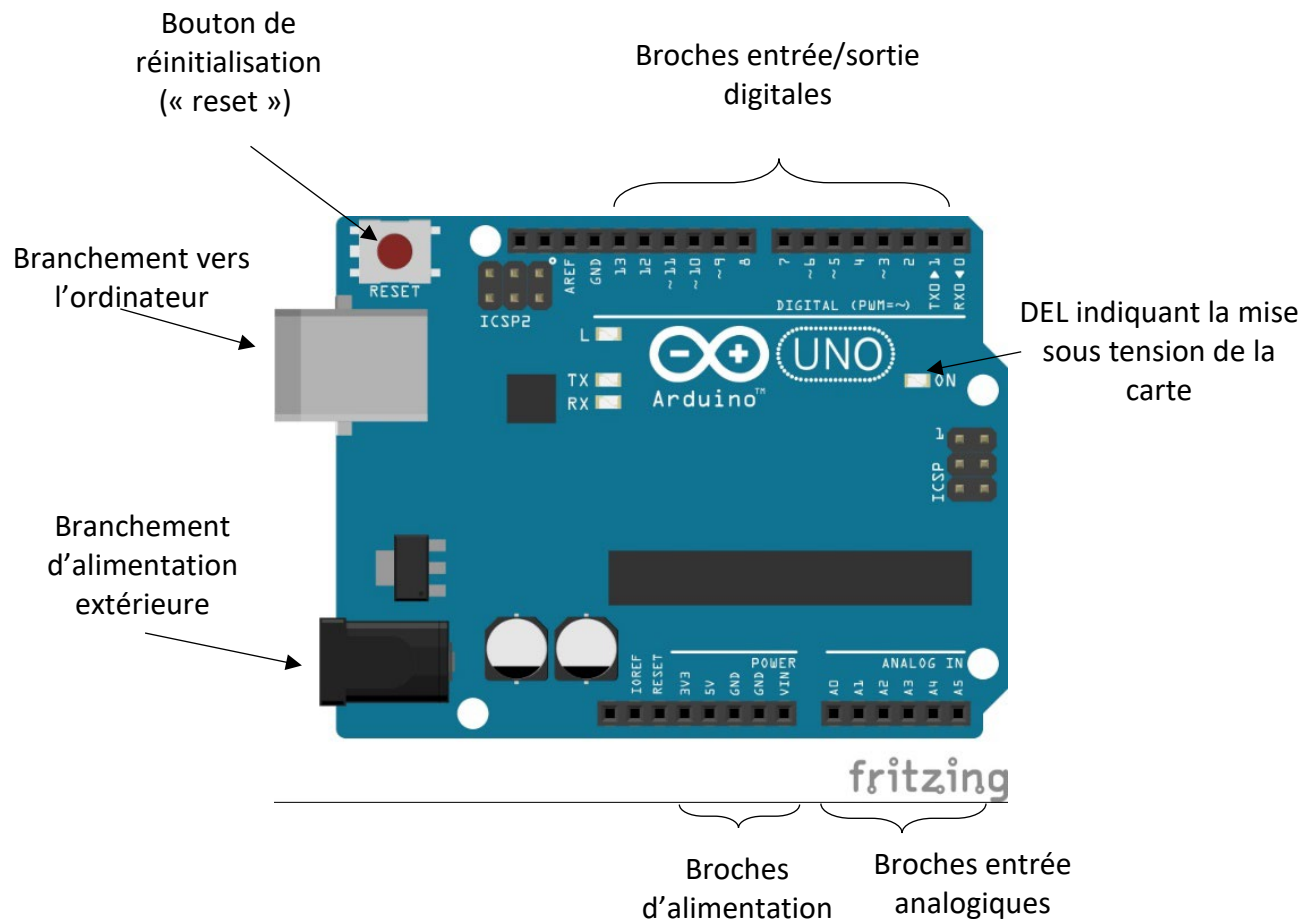
3. Les cartes de type Arduino UNO

Les cartes Arduino ou compatibles avec Arduino sont des cartes sur lesquelles on retrouve un microcontrôleur permettant de programmer les actions des différentes composantes branchées aux entrées et sorties de la carte. Il est ainsi possible de réaliser un programme permettant d'allumer une ampoule (signal de sortie) lorsque la température (signal d'entrée) dépasse une certaine valeur.

Voici les différentes parties de base d'une carte de type Arduino UNO®. Il y a d'autres fonctionnalités qui seront vues plus tard au besoin.

Les broches identifiées « GND » servent de mise à la terre pour le retour du courant. Elles jouent toutes le même rôle et peuvent être interchangées sans problème.

Pour l'instant, l'alimentation électrique sera fournie par le branchement vers l'ordinateur.

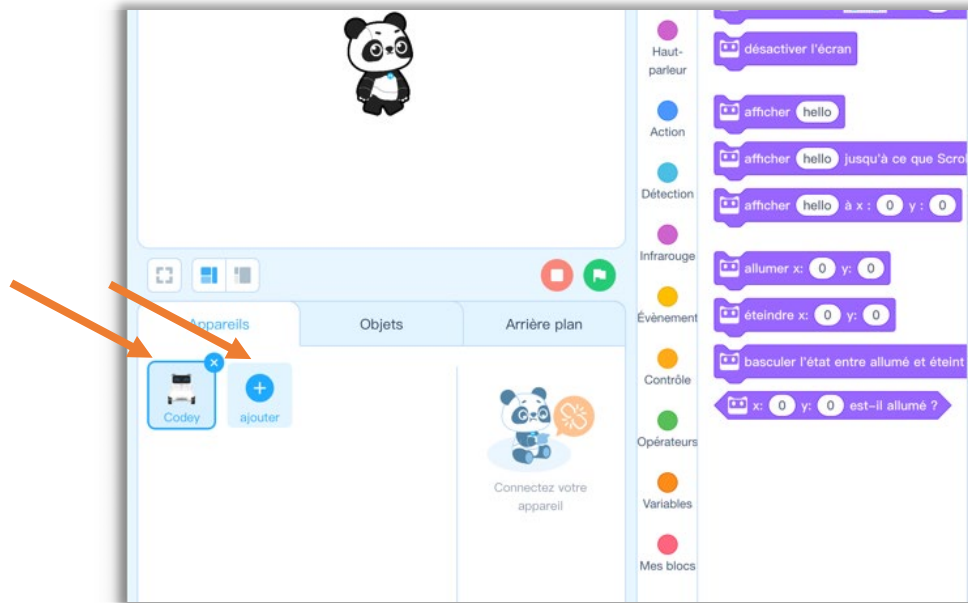


4. Ce que fait le programme

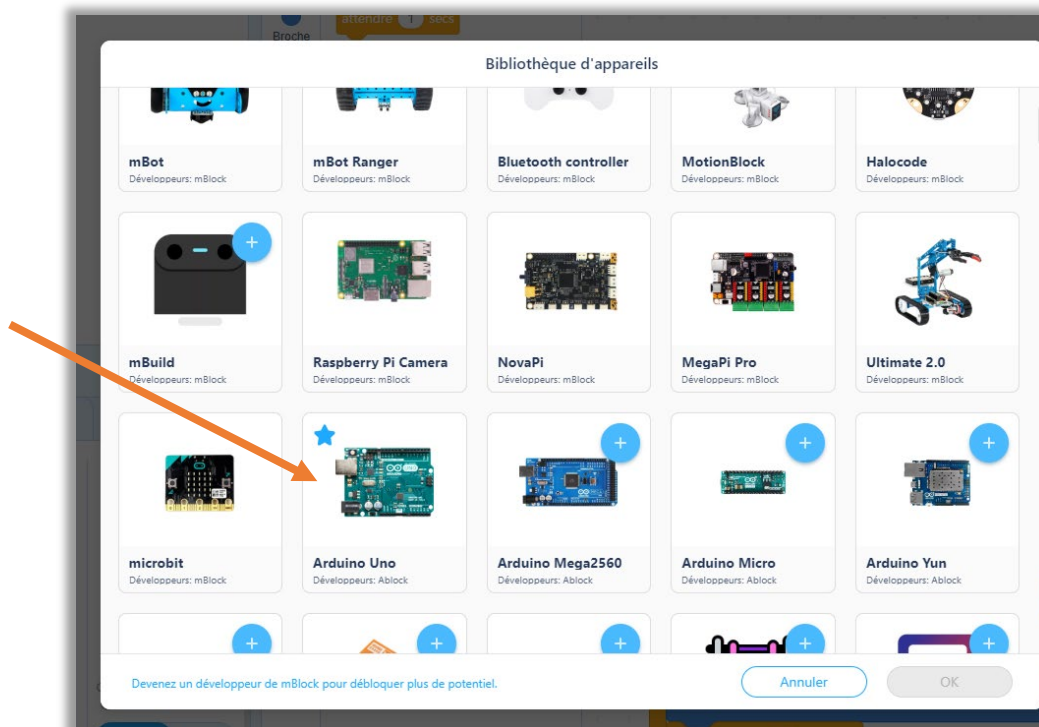
Ce programme permet de faire clignoter une diode électroluminescente (DEL) : Allumée pendant 1 s et éteinte pendant 1s, et ainsi de suite.

5. Étapes de réalisation

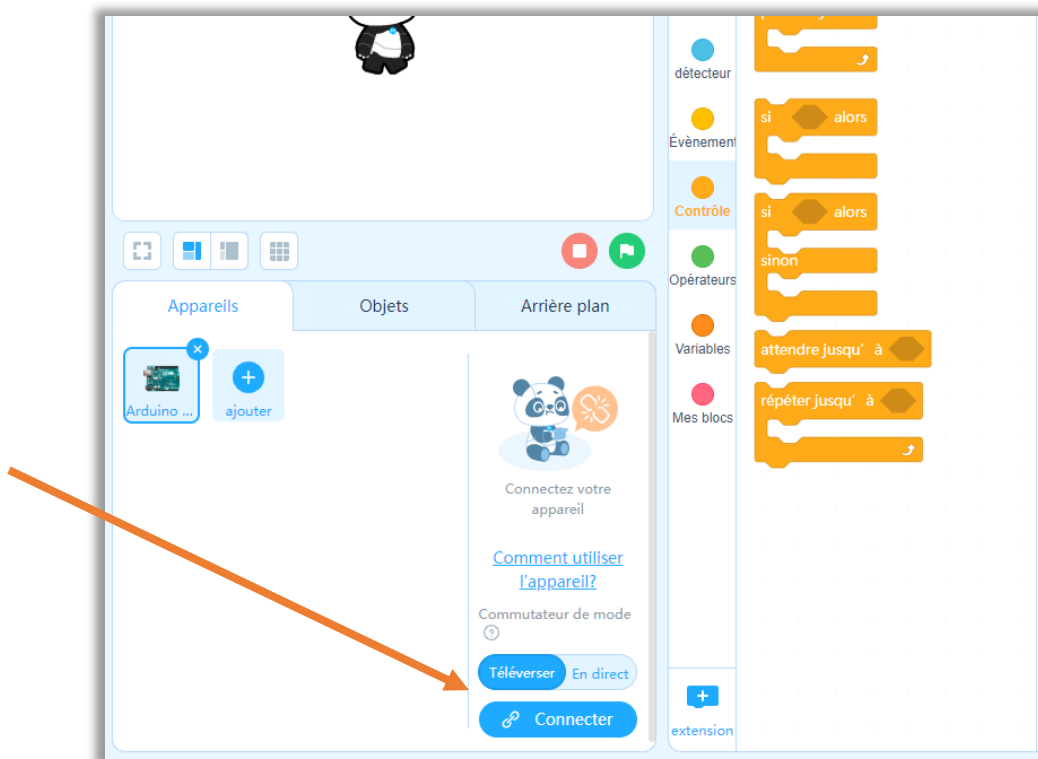
1. Brancher la carte Arduino UNO dans l'ordinateur avec le fil approprié. La DEL (verte) d'alimentation de la carte devrait être allumée.
2. Ouvrir le logiciel mBlock 5.
3. Dans la section *Appareils*, en bas à gauche, supprimer *Codey* ou autre et cliquer sur *ajouter*.



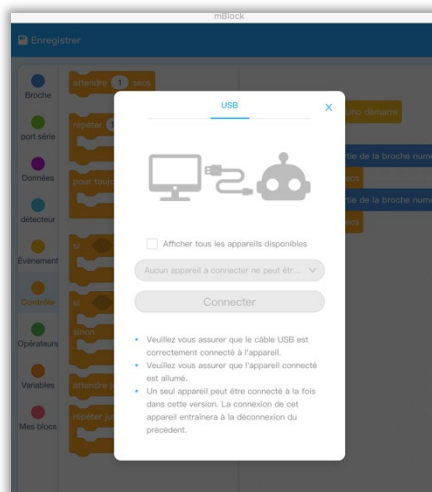
4. Sélectionner la carte Arduino UNO dans la fenêtre qui s'est ouverte. Si un cercle vert avec une flèche blanche est situé sur la vignette de la carte Arduino, ou encore un cercle bleu avec une croix blanche, il faut cliquer sur ce cercle (bleu ou vert) avant de pouvoir « Ajouter » la carte Arduino. Cela signifie simplement que la carte n'est pas installée et le clic l'installera.



5. Dans la section *Appareils*, cliquer sur le bouton *Connecter*.

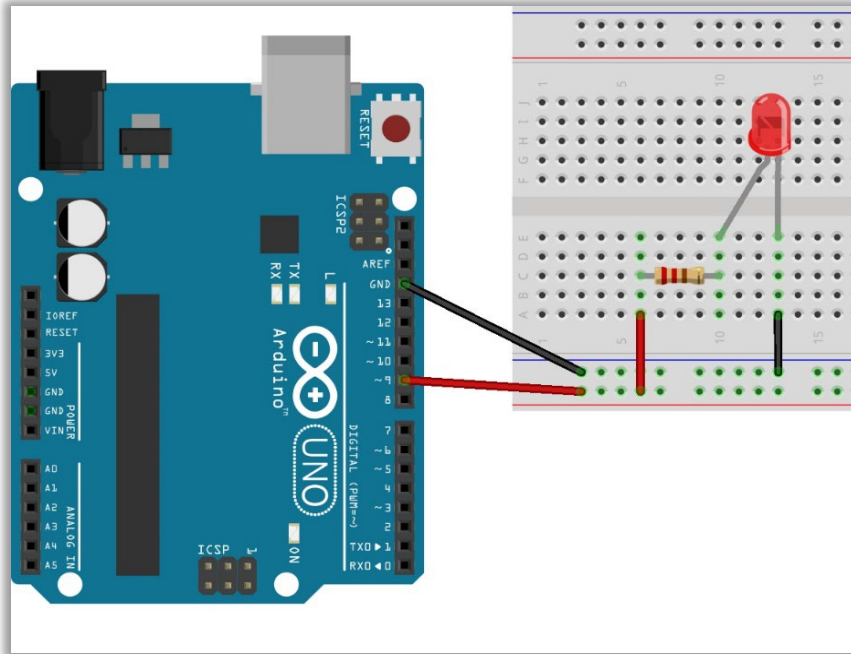


6. Dans la fenêtre qui s'ouvre, sélectionner le port USB approprié dans la liste déroulante et cliquer sur *Connecter*. Si nécessaire, cocher *Afficher tous les appareils disponibles*.

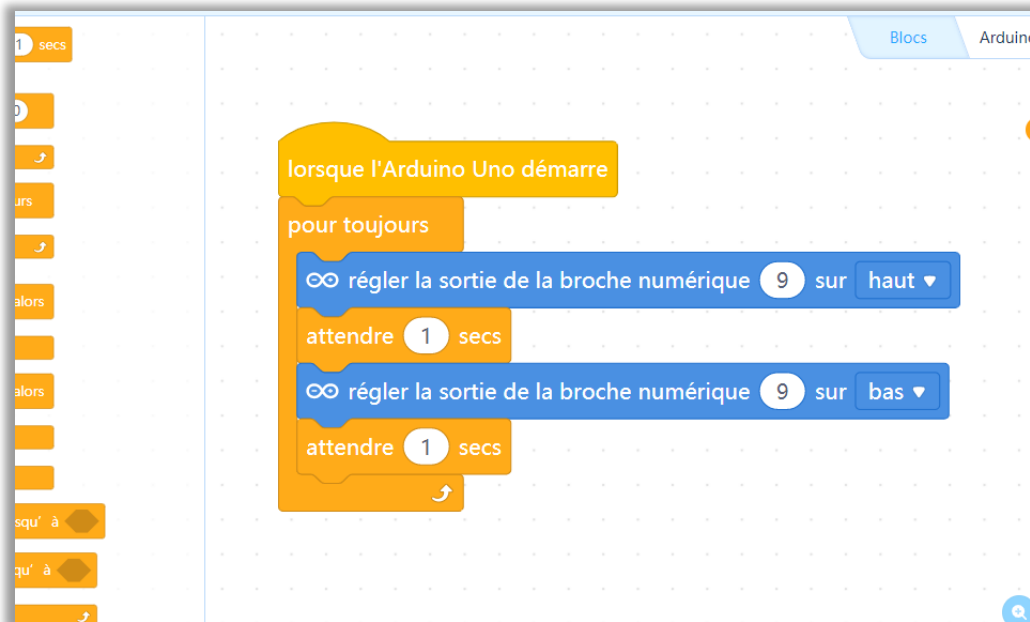


7. Réaliser le circuit.
8. Écrire le code en cherchant et en trouvant et déplaçant les blocs appropriés dans les différentes sections. Indice : *Évènement*, *Broche* et *Contrôle*.
9. Appuyer sur *Téléverser* et observer le résultat !

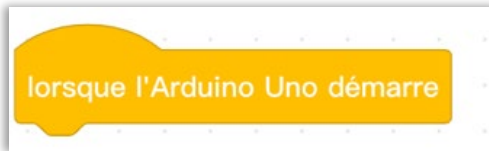
6. Schéma du circuit



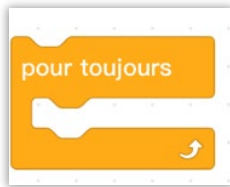
7. Code



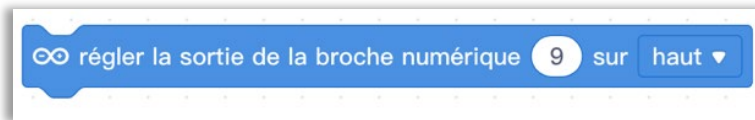
8. Blocs utilisés



Les blocs de type *Évènement* sont utilisés pour amorcer une série d'actions qui débutent lorsque l'évènement décrit dans le bloc se produit. Ici, les blocs suivants seront exécutés lorsque la carte Arduino démarre, ou lorsque l'on appuie sur le bouton « Reset » de la carte.

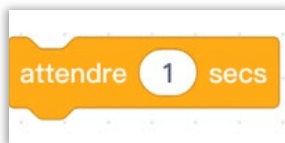


Ce bloc de contrôle est ce que l'on appelle une *boucle*. Des blocs peuvent être inclus à l'intérieur de ce bloc. Ces blocs seront exécutés un à la suite de l'autre et cette séquence se répétera, pour cette boucle, *pour toujours*. Du moins, tant que la carte Arduino est alimentée en courant.



Ce bloc permet d'envoyer un signal électrique à la broche 9. La tension électrique fournie par la broche sera d'environ 5V lorsque la valeur *haut* est utilisée et de 0V pour la valeur *bas*.

La tension électrique ne change pas tant qu'un bloc similaire n'est pas utilisé pour la modifier.



Ce bloc introduit un délai d'attente du nombre de secondes indiquées. Si l'on souhaite une attente qui est une fraction de seconde, on doit utiliser un nombre décimal avec une « . ». Par exemple : 2.3 ou 0.4.

Défis :

- Changer la broche (PIN) de branchement de la DEL et ajuster le programme en conséquence.
- Modifier le programme pour faire d'autres rythmes de clignotement!
- Faire clignoter la DEL une seule fois!
- Réussir à faire clignoter la DEL pour obtenir le code morse de S.O.S.!
Rechercher dans Google si vous ne connaissez pas le code morse!