

Le matériel

Arduino Uno est une carte programmable intégrant un micro-contrôleur.

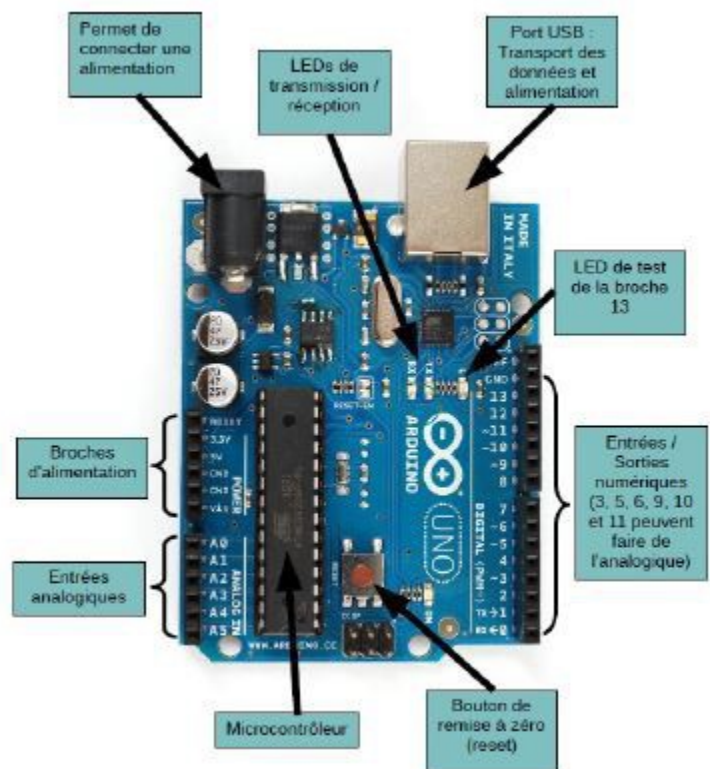
C'est un composant programmable qui pour l'instant ne sait rien faire ou presque.

Il se connecte via un câble USB.

Pour lui apprendre à réaliser des tâches utiles, il va falloir le programmer.

La carte peut recevoir sur ses **entrées** des signaux **numériques** ou **analogiques**.

Ensuite le micro-contrôleur traite les **informations** et les transmet sur ses **sorties numériques**.



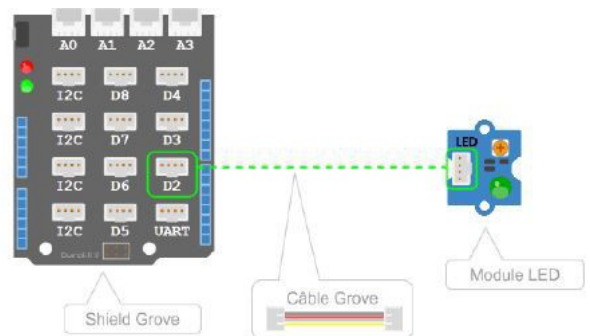
Comme vous pouvez le voir, le câblage de composants n'est pas facile pour des débutants, nous utiliserons donc des **modules** pré-câblés Grove et une interface Shield Grove.

A0 A3 : entrées analogiques

(Capteurs lumière, t°C, humidité)

D2 D7 : entrées/sorties numériques (digital)

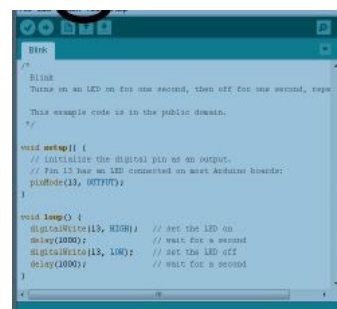
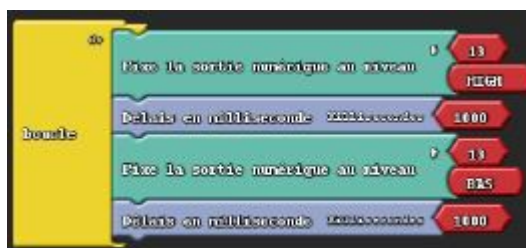
(Interrupteur, del, ultrason, moteur)



Les logiciels

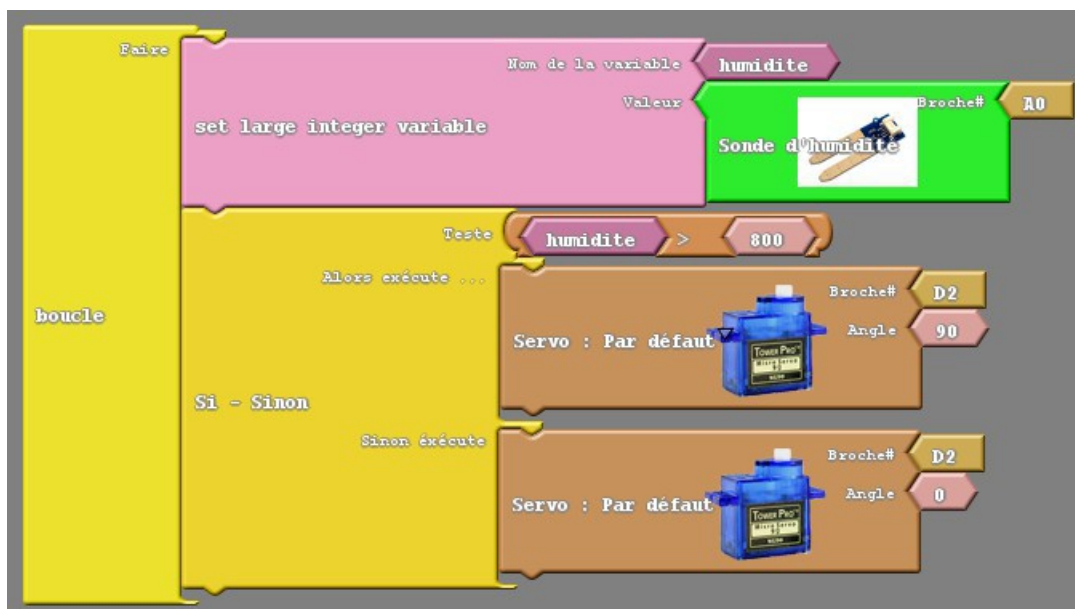
Algorithme :

- Programmation avec interface graphique Ardublock (type Scratch)
- Programmation avec du code (type C++)



Maîtriser Ardublock

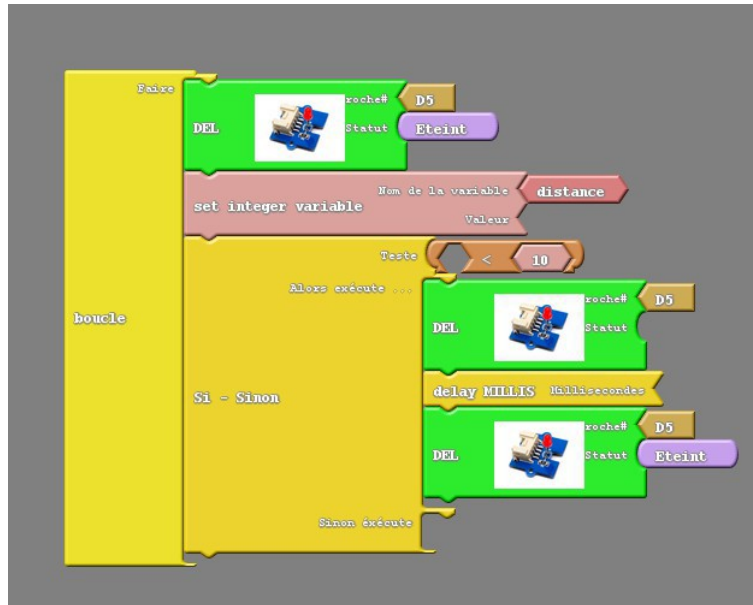
1. Analyser et retrouver les algorithmes correspondant aux programmes suivants



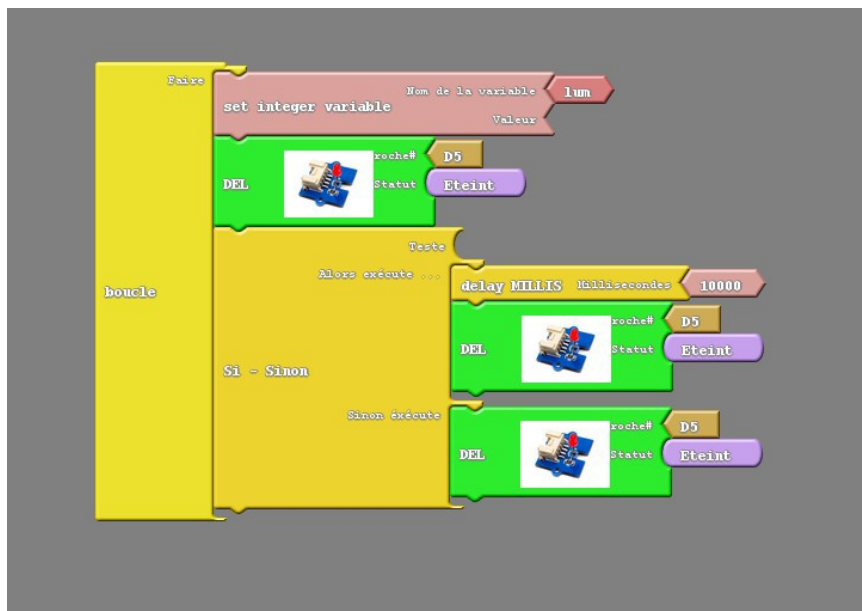
Comment programmer le tracker solaire automatiquement ?

2. A partir des algorithmes, compléter et tester les programmes sur les maquettes didactiques

- Algorithme 1 : la lumière s'allume 15s SI une présence est détectée à moins de 10cm

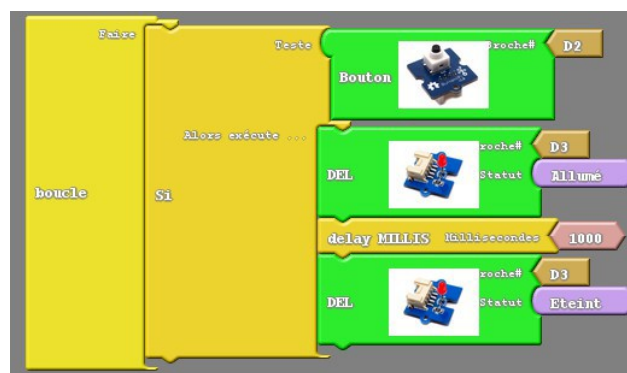


- algorithme 2 : la lumière s'allume 10s SI luminosité < 160 ET interrupteur appuyé



Maîtriser « un peu » le code

- Sur Ardublock, réaliser le programme suivant puis le téléverser sur l'Arduino. Recopier le code sur votre feuille et analysons ensemble les différentes structures.



Synthèse

- Un programme en code Arduino se présente de la façon suivante :

```
① définition des constantes (int ou byte ou boolean) et des bibliothèques de composants (#include <nom>)  
  
② void setup()                                Les instructions se terminent par ;  
{  
instruction à n'exécuter qu'une seule fois ;    { et } fonctionnent toujours par paire et  
                                                    encadrent un bloc d'instructions  
}  
  
③ void loop()                                // permet d'écrire un commentaire  
{  
instructions à répéter en boucle ;  
}
```

Algorithme : une del s'allume 10s si la luminosité est < à 512

```
int del = 3 ;                                // la del est connecté sur le Shield D3  
int ldr = A0 ;                               // la ldr est connectée sur le Shield A0  
int luminosite ;                             // on crée une variable luminosité  
  
void setup()  
{  
pinMode (del, OUTPUT) ;                     // la del est un signal d'entrée  
}  
  
void loop()  
{  
luminosite=analogRead(ldr) ;                 // lecture valeur analogique de la ldr assignée à luminosité if (luminosite<512)  
                                              // si la luminosité est inférieure à 512  
  
    {  
        digitalWrite (del, HIGH) ;           // mettre la del à la valeur allumée delay  
        (10000) ;                             // pendant 10s  
    }  
}
```

Programmer le tracker

3. Écrire l'algorithme relative au tracker (2 ldr couplés à un servomoteur pour l'azimut / 2 ldr couplés à un servomoteur pour la hauteur).
4. Dans notre projet, Ardublock atteint vite ses limites, **il va falloir programmer en lignes de code !**