

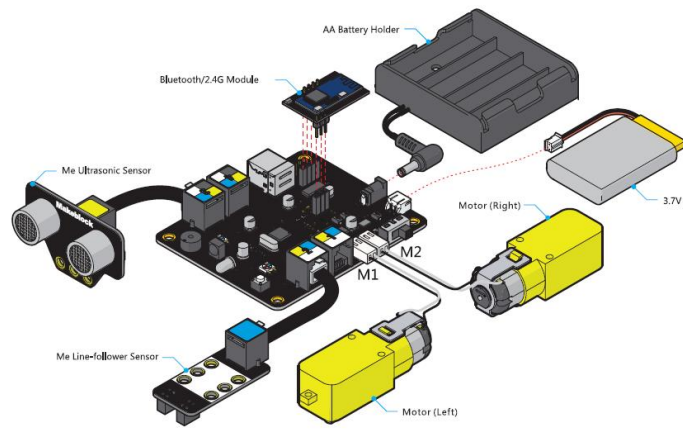
CYCLE 4

Synthèse programmation
Initiation à la programmation textuelle

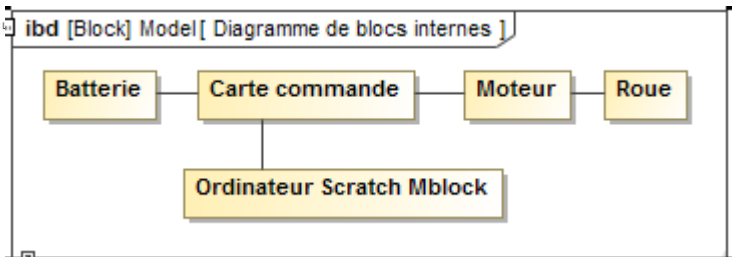
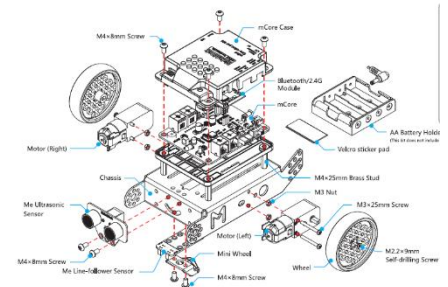
NIVEAU
TROISIEME

Connexions d'un système (blocs internes)

Un système utilise des composants pour fonctionner. Il faut connaître comment sont connectés ses composants.



Parts List



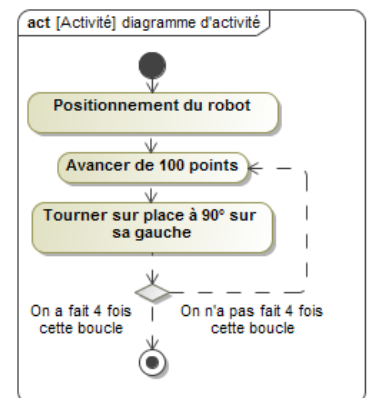
Un microcontrôleur héberge et exécute le programme informatique.

Notion d'algorithme

Avant de programmer un système, une phase d'analyse est indispensable pour définir le fonctionnement attendu et en repérer les différentes étapes.

Ces étapes sont ensuite organisées dans un ordre chronologique. Pour visualiser cet enchaînement, il est courant de réaliser un **diagramme d'activité** (ou un organigramme).

Un algorithme est une suite ordonnée de traitements élémentaires effectués sur des données en vue d'obtenir un résultat.



Les étapes d'un algorithme sont présentées sous forme de phrases simples.

On obtient le même résultat à chaque fois avec les mêmes données et le même algorithme.

DEBUT¶

- Allumer le clignotant¶
- Attendre 1s¶
- Eteindre le clignotant¶
- Attendre 1s¶
- Allumer le clignotant¶
- Attendre 1s¶
- Eteindre le clignotant¶

FIN¶

Coder (programmer) une application

Le développement d'une application s'appuie sur un environnement de développement (IDE) tel que Scratch, mBlock ou Vittascience. Programmer consiste à concevoir et transmettre des suites d'instructions ordonnées à un système.

Au-delà de la programmation par blocs, un même algorithme peut être transcrit dans d'autres langages, notamment des langages textuels comme le Python. Ce type de programmation repose sur l'écriture de commandes textuelles précises et normées.

Quel que soit le code choisi, un langage de programmation permet de communiquer avec une machine par l'intermédiaire d'un traducteur (un compilateur ou un interpréteur) qui convertit les instructions en langage machine (binaire).

Notion de variable

On utilise une variable pour sauvegarder des informations (nombre, texte, booléen = type).

Une variable est définie par son nom et son contenu et son type.

Pour affecter une valeur à une variable (mémorisation d'une donnée) :

```
lumiere=3
texte="bonjour"
valeur_binaire=True
```

Commander des actions

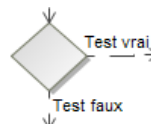
Pour commander une action, le microcontrôleur pilote ses sorties en envoyant deux types de signaux :

- **Un signal logique (Tout ou Rien)** : Le signal n'a que deux états (0 ou 1), utile pour allumer ou éteindre un composant.
- **Un signal modulé** : Ce signal permet d'appliquer plusieurs valeurs de commande intermédiaires (ex : faire varier la vitesse d'un moteur).

Structures algorithmiques

Test de condition : On peut effectuer des instructions en fonction d'un test vrai ou faux.

```
if condition_vraie:
    #instructions si condition vraie
else:
    #instructions si condition fausse
```



Les boucles non bornées sont à utiliser lorsque l'on ne sait pas le nombre de boucles à réaliser à l'avance. On utilise une information (condition) pour stopper la boucle.

```
while condition vrai:
    #instructions à répéter...
    #instructions à répéter...
```

