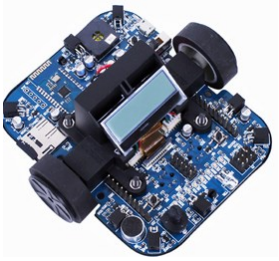
# ANALYSE FONCTIONNELLE



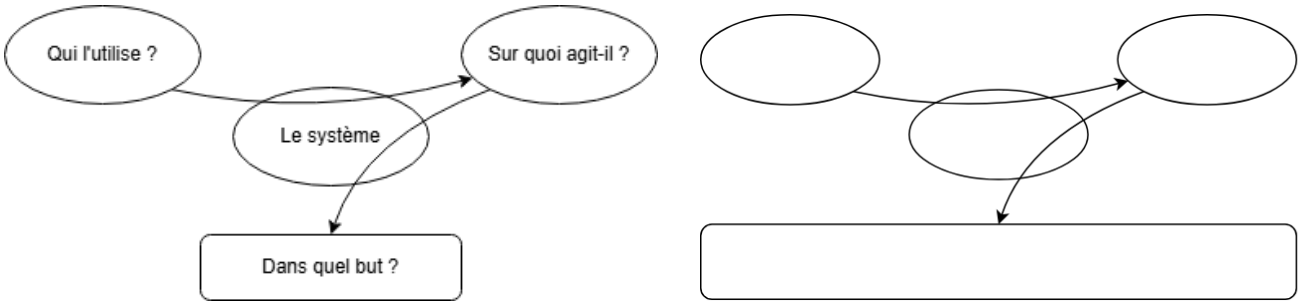
Pour permettre aux robots de fonctionner il faut assembler des pièces mécaniques commandées par des composants électroniques. L'organisation fonctionnelle peut se décomposer en une chaine d'informations et une chaîne d'énergie.

- **La chaîne d'information** agit sur les flux de données.
- **La chaîne d'énergie** agit sur les flux de matières et d'énergies.

Différents diagrammes permettent de décrire l'organisation fonctionnelle, structurelle, énergétique, etc..., d'un système

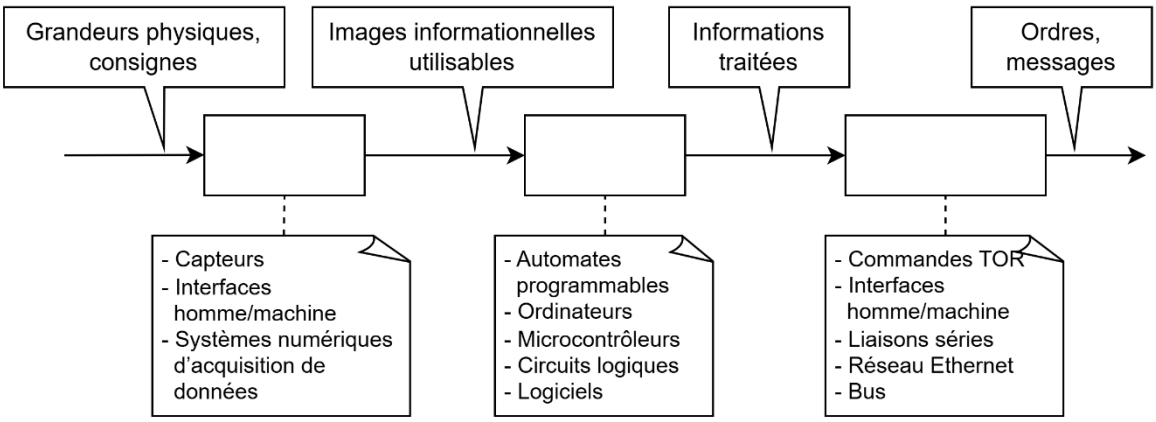
| Que veut-on décrire ? | Diagramme à utiliser |
|---|---|
| **Le besoin** | Diagramme bête à corne |
| **Les interactions entre le système et ses utilisateurs** | Diagramme de cas d'utilisation (SYSML) |
| **Les exigences et les contraintes** | Diagramme des exigences (SYSML) |
| **L'enchaînement séquentiel des interactions** | Diagramme de séquence (SYSML) |
| **Les composants et les énergies** | Diagramme interne de bloc (SYSML) |
| **Le comportement du système** | Diagramme d'état (SYSML) |
| **Le programme** | Algorithme |

## 1. DIAGRAMME BETE A CORNE




**Complétez** le diagramme bête à cornes du robot pédagogique formula all code

## 2. LA CHAINE D'INFORMATION

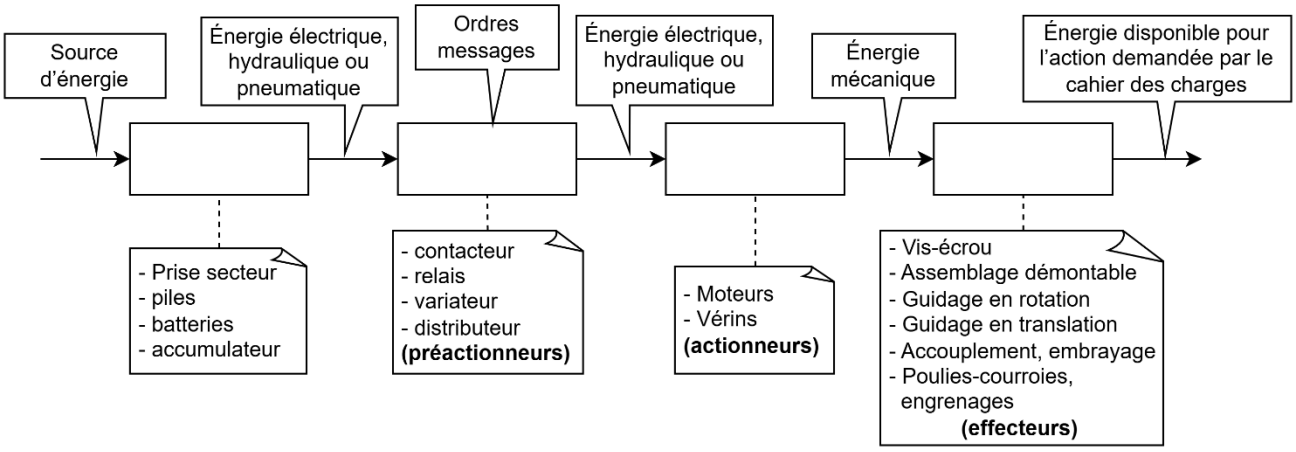



**ACQUERIR :** adapte les caractéristiques d'une information, pour être compatible avec le système de traitement,

**TRAITER :** traite les informations afin d'en établir de nouvelles pour modifier le fonctionnement du système,

**COMMUNIQUER :** communique les informations générées par le système de traitement vers la machine et peut en informer l'homme.

## 3. LA CHAINE D'ENERGIE :



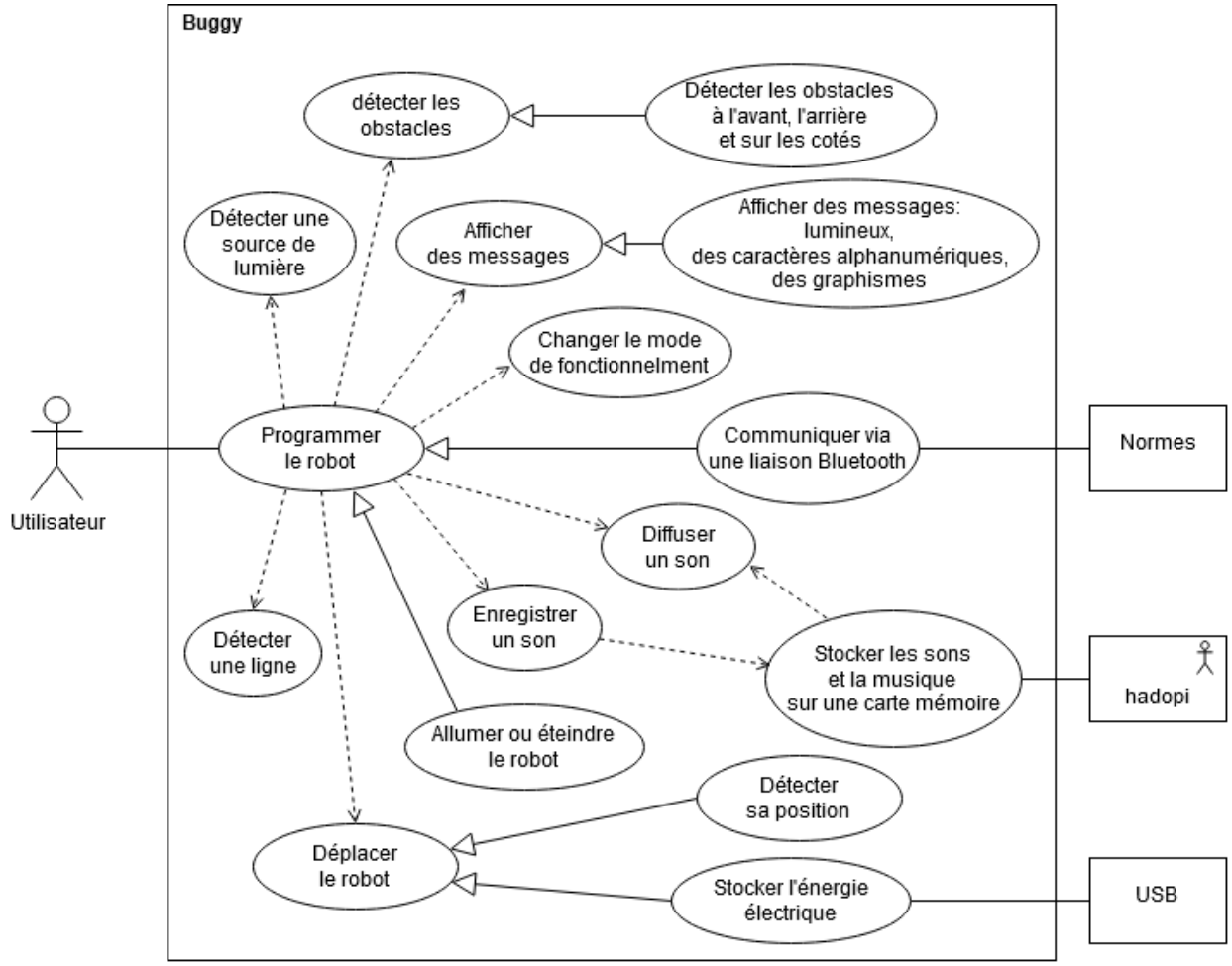

**ALIMENTER :** apporte l'énergie appropriée au système pour fonctionner,

**DISTRIBUER :** permet de bloquer ou laisser passer l'énergie,

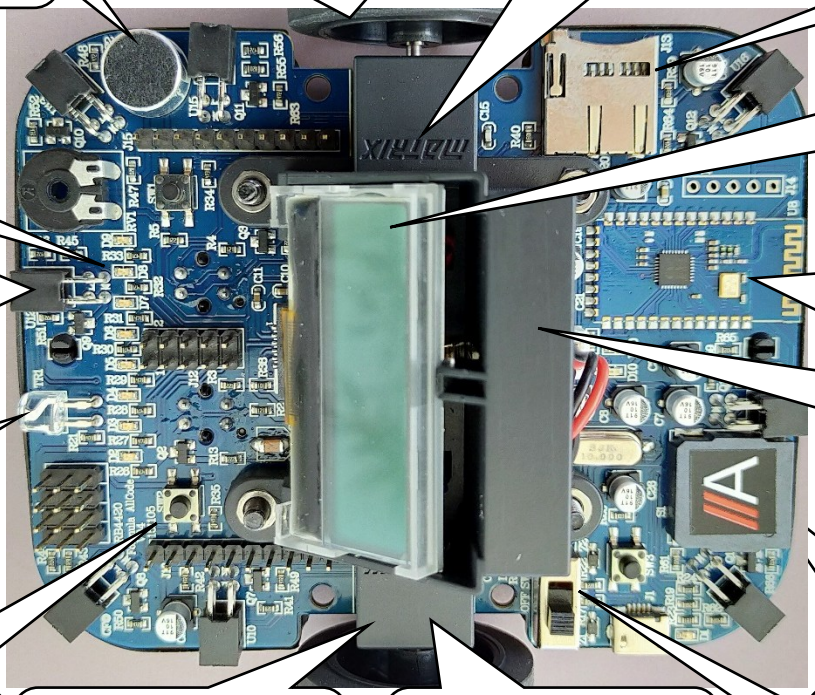
**CONVERTIR :** transforme l'énergie,

**TRANSMETTRE :** permet d'animer la chaîne cinématique.

## 4. DIAGRAMME SYSML DE CAS D'UTILISATION



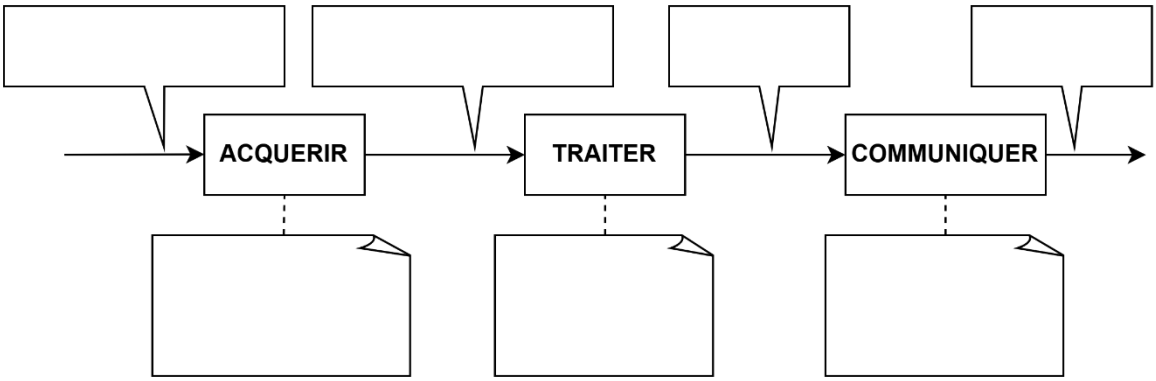
Compléter les éléments ci-dessous en précisant leur fonction et à quelle chaîne ils appartiennent




## 5. LA CHAINE D'INFORMATION ET D'ÉNERGIE DU ROBOT

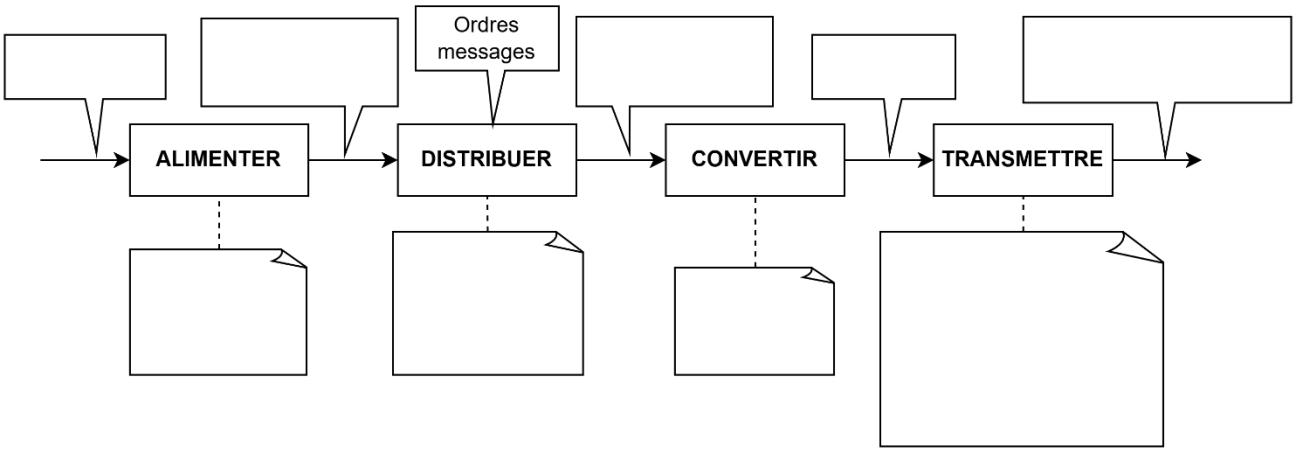
Compléter la chaine fonctionnelle en plaçant les éléments suivants :

- CI : Microphone, Roues, Moteur, Microcontrôleur, Récepteur BT, Émetteur BT, Capteur luminosité, Capteur IR.

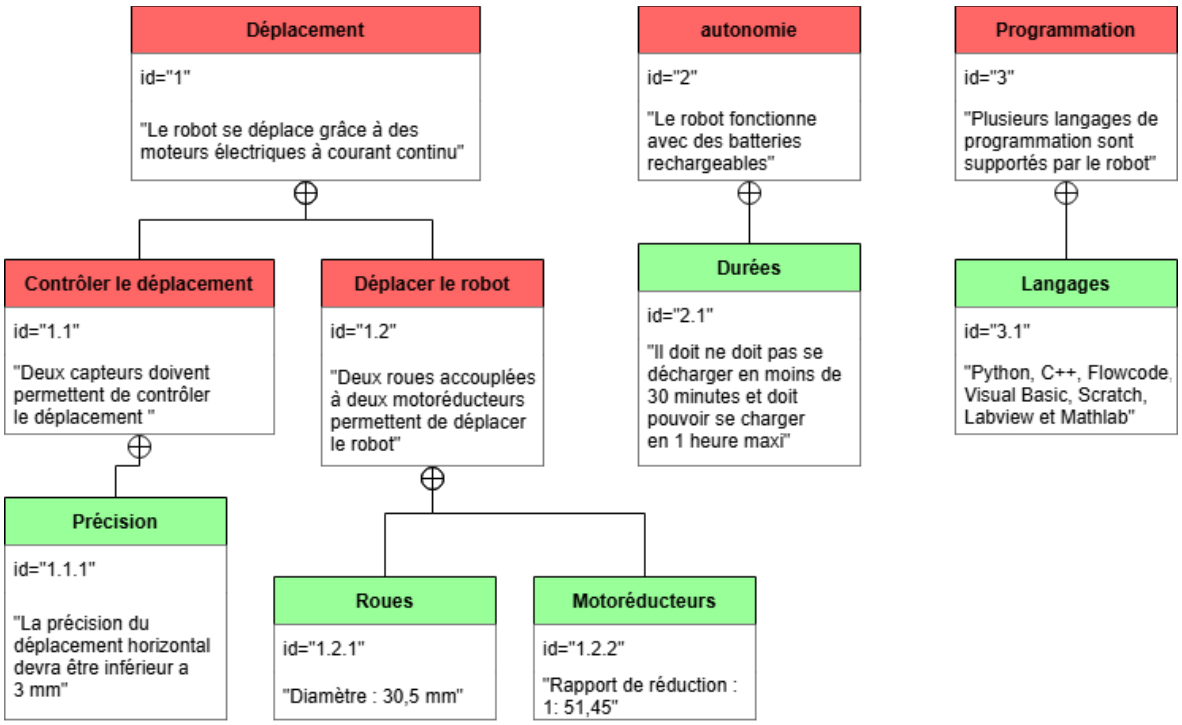


Compléter la chaine d'énergie en plaçant les éléments suivants :

- CE : Batterie, Engrenages, Interrupteur M/A, LED, Haut-parleur, Boutons poussoirs, Variateur de vitesse, Afficheur LCD, Transistors



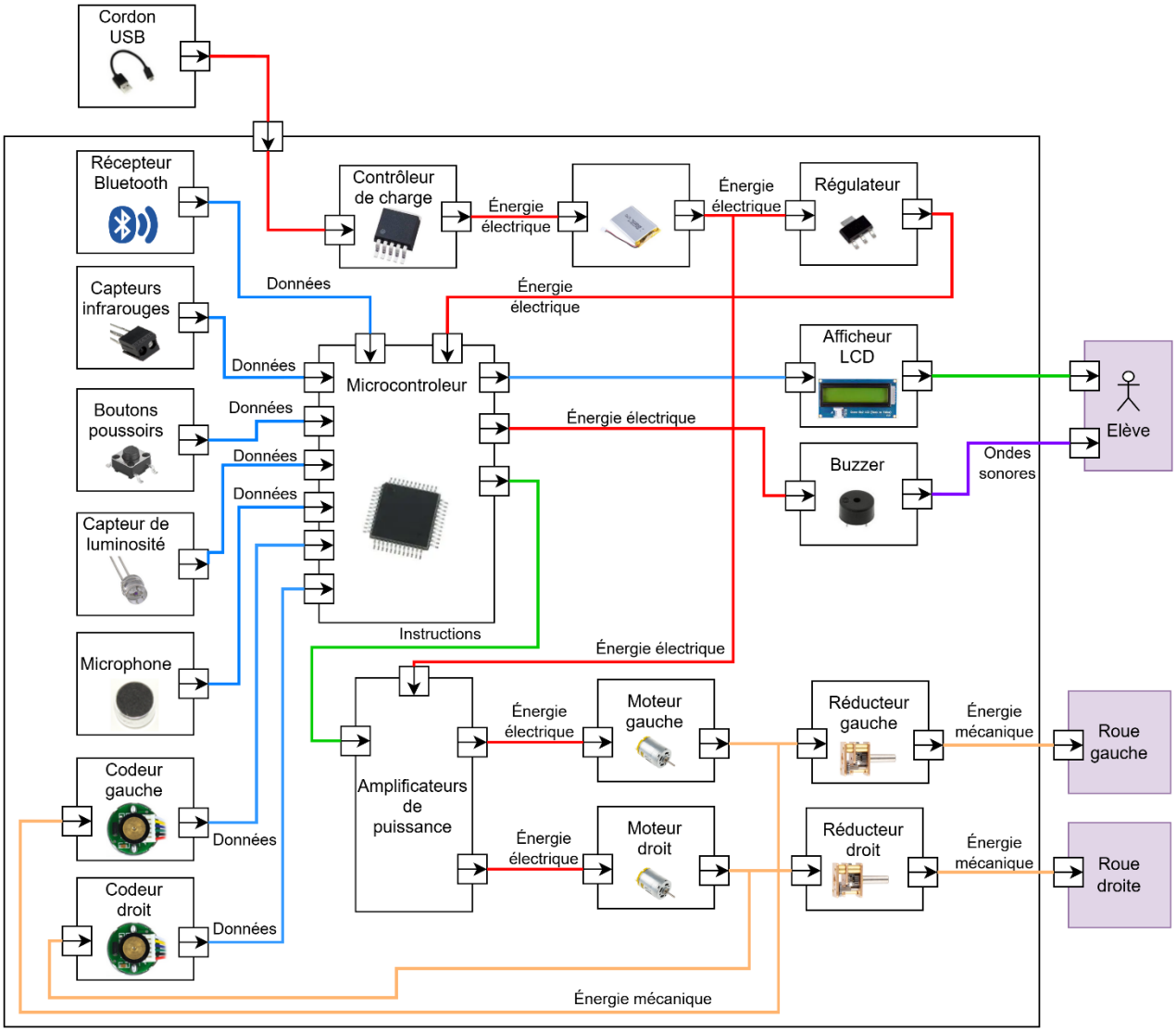

## 6. DIAGRAMME SYSML PARTIEL DES EXIGENCES PARTIEL DU ROBOT




Répondre aux questions suivantes :

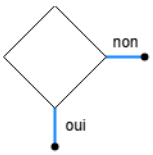
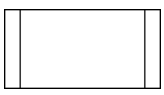
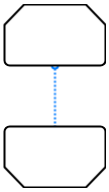
- Combien de temps peut-on faire fonctionner le robot avant qu'il ne soit déchargé ? _____
- Quel est le diamètre des roues ? _____
- Rechercher le rapport de réduction du motoréducteur : _____
- Quelle est la précision de déplacement du robot ? _____
- Peut-on programmer le robot en langage javascript ? _____

## 7. DIAGRAMME INTERNE DE BLOC (PARTIEL)




## 8. ALGORIGRAMMES ET ALGORITHMES

Les algorithmes permettent de décrire un programme sous forme de langage naturel, les algorigrammes permettent de décrire un programme sous forme de blocs

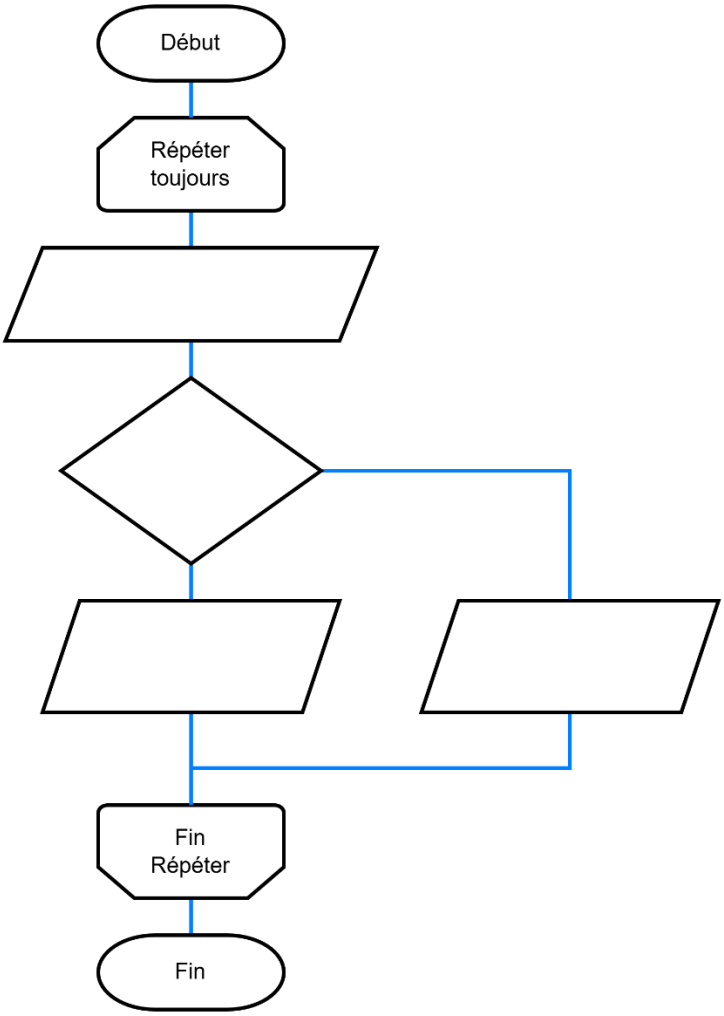
| Symbole | Description | Symbole | Description |
|---|---|---|---|
| | Début ou fin du programme | | Action ou calcul |
| | Entrée/sortie |  | Test |
|  | Appel d'un sous-programme |  | Lien entre blocs |
| | Pause |  | Boucle |

Le programme est exécuté par le microcontrôleur :

- Les LED sont considérées comme ses _____
- Les boutons poussoirs sont considérées comme ses _____

Exemple d'algorithme

```
Début
    Répéter toujours
        bp ← lire l'état de
BP1
        si bp = 1 alors
            allumer la LED1
        sinon
            éteindre la LED1
        fin si
    fin répéter
fin
```




Compléter l'algorigramme ci-contre correspondant à l'algorithme

## 9. PROGRAMMATION EN LANGAGE PYTHON

Complétez le programme suivant :

```
while True :                    # répéter indéfiniment
    bp = etat_BP("gauche")      # Lire l'état du bouton poussoir gauche
    if bp == ______ :           # s'il est activé alors
        allumer_led(__)         # allumer la LED
    _______ :                   # sinon
        eteindre_led(__)        # éteindre la LED
```