



Exploration avec les robots

Livret ressources

Premier temps du parcours – Activités préparatoires en classe

Introduction

Mise en situation

En 1996, la NASA a envoyé un robot explorateur sur Mars. Dénommé Sojourner, il est le premier véhicule à roues à se déplacer sur une autre planète que la Terre.

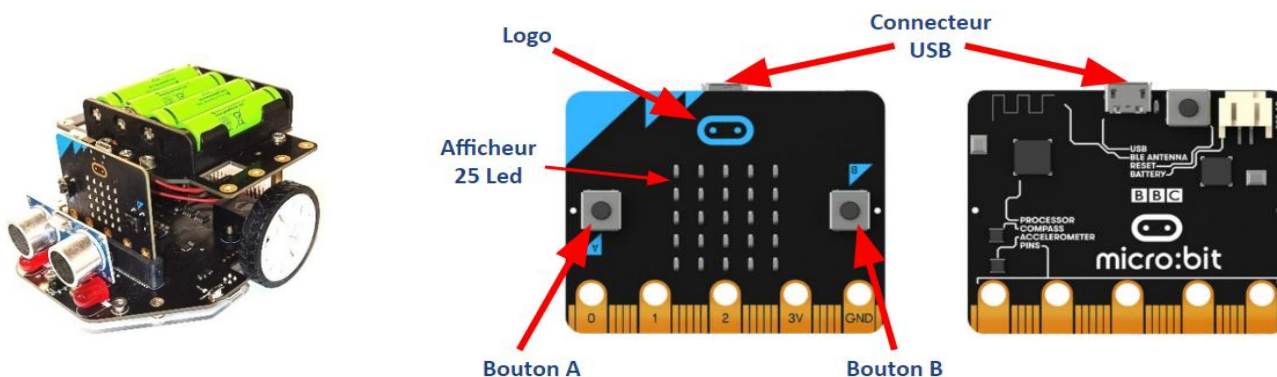
Nous allons chercher et comprendre quels sont les défis à la réalisation d'une telle avancée et l'importance de la robotique dans ce contexte.

Le saviez-vous ?

Pour nommer ses robots, la NASA lance un concours auprès de la jeunesse américaine. Alors que Valérie Ambroise n'a que 12 ans, elle remporte le concours et donne le nom de Sojourner.

Sojourner signifie voyageur ou étranger, mais Valérie a choisi ce nom en hommage à Sojourner Truth, une afro-américaine militante pour le droit des femmes, quelques que soient leurs origines, et pour l'abolitionnisme de l'esclavage au XIX siècle.

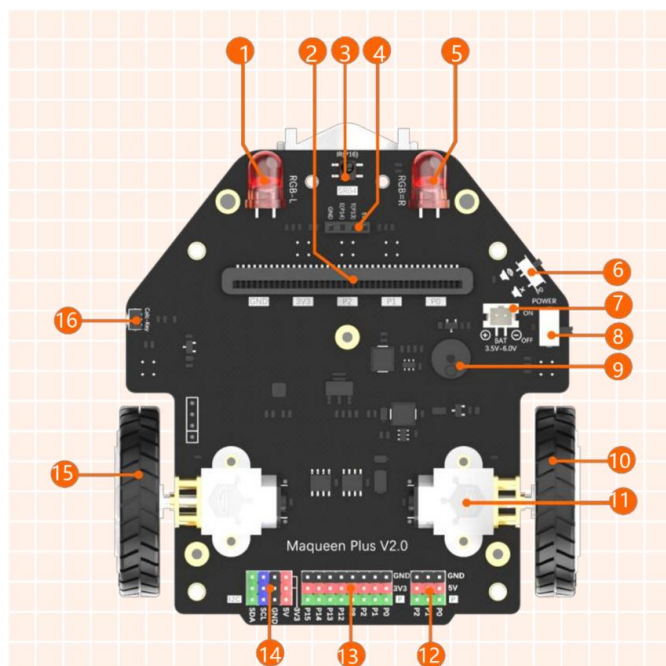
Présentation du matériel



Notre robot est pilotable grâce à une carte électronique munie d'un microprocesseur et de son programme qui permet le pilotage du moteur de chaque roue. Nous allons utiliser une carte micro:bit de [Microbit.org](https://microbit.org).

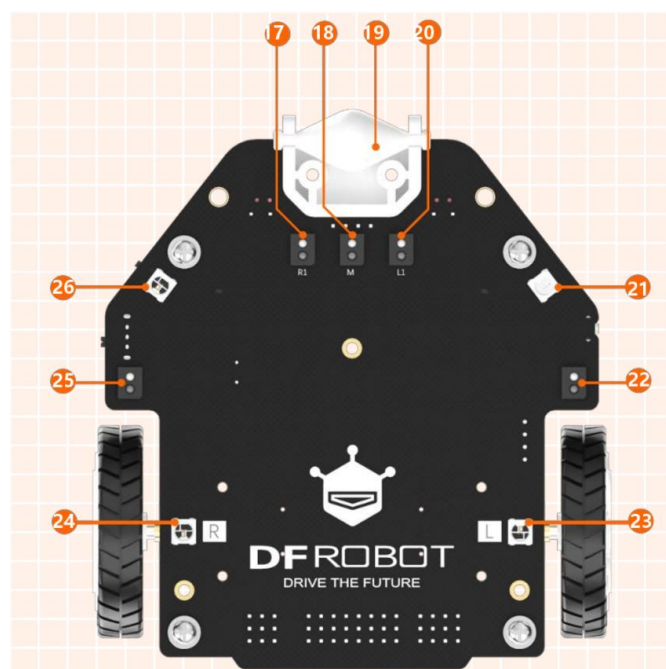
Découverte du robot

Vue du dessus



- 1 – LED RGB Gauche
- 2 – Port micro:bit
- 3 – Récepteur infrarouge
- 4 – Port capteur à ultrason
- 5 – LED RGB Droite
- 6 – Interrupteur du buzzer
- 7 – Port d'alimentation
- 8 – Interrupteur d'alimentation
- 9 – Buzzer
- 10 – Roue droite
- 11 – Moteur
- 12 – Ports pour servomoteur
- 13 – Ports d'entrée/sortie
- 14 – Port I2C (permet la connexion de périphériques)
- 15 – Roue gauche
- 16 – Bouton de calibration des capteurs de ligne

Vue du dessous



- 17 – Capteur de ligne « droit »
- 18 – Capteur de ligne « milieu »
- 19 – Roue d'appui
- 20 – Capteur de ligne « gauche »
- 21 – LED RGB 0
- 22 – Capteur de ligne « arrière gauche »
- 23 – LED RGB 1
- 24 – LED RGB 2
- 25 – Capteur de ligne « arrière droit »
- 26 – LED RGB 3

Sur le robot, on va trouver des **capteurs** qui permettent de recevoir des informations, et des **actionneurs** qui réalisent l'action.

Consigne : entourer les noms des capteurs du robot Maqueen Plus V2, et souligner les noms des actionneurs dans le livret élève.

Activité 1 – Introduction à l’algorithmie

Un **algorithme** est une suite d’instructions ou d’étapes à suivre pour résoudre un problème ou faire fonctionner quelque chose.

Exemple :

Algorithme de déplacement automatique d’un robot qui peut rencontrer un obstacle

Si le robot ne détecte pas d’obstacle à moins de 2 cm, celui-ci doit continuer d’avancer sinon le robot et son programme s’arrêtent.

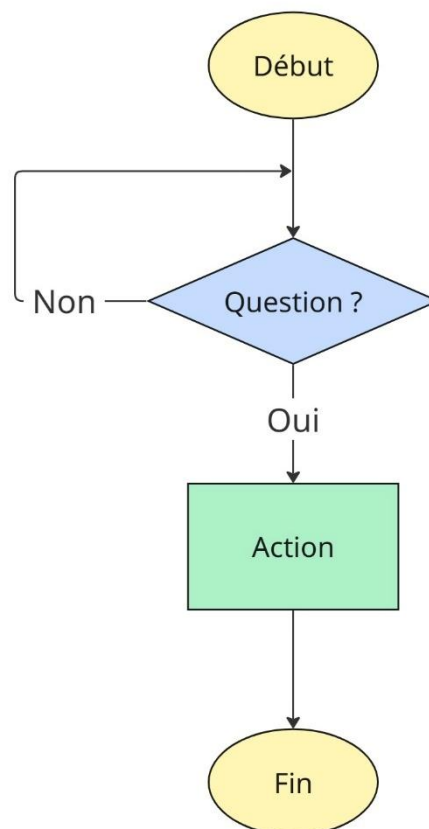
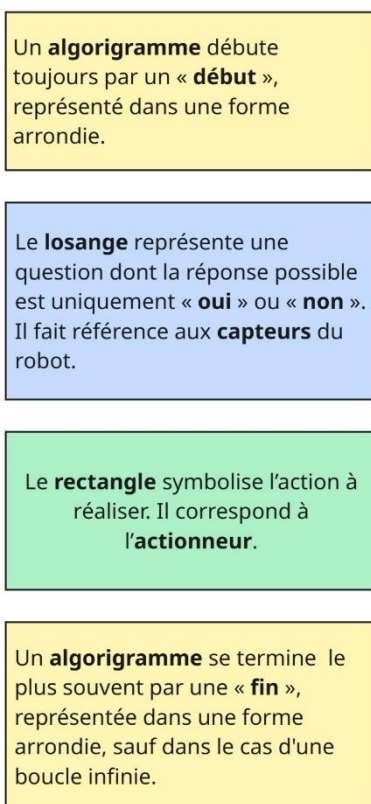
Si on traduit la phrase précédente en langage algorithmique, cela donne :

Obstacle à moins de 2 cm ?

→ Oui : le robot s’arrête + fin du programme

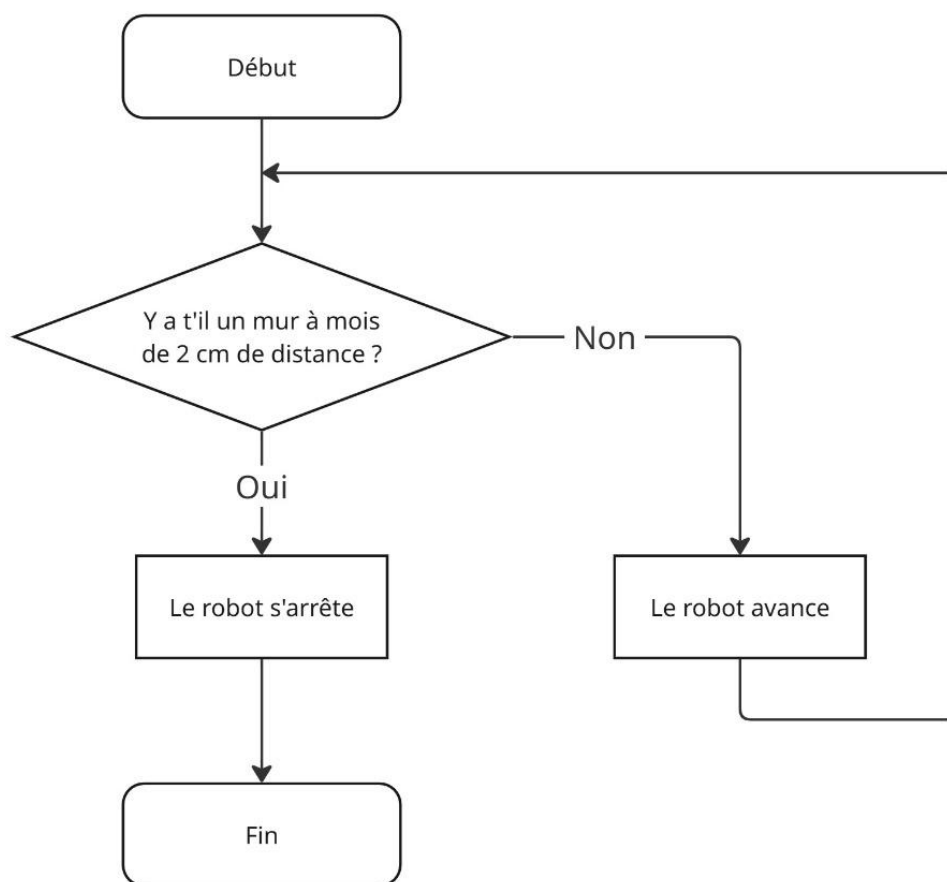
→ Non : le robot avance

Un **algorithme** (ou organigramme algorithmique) est une représentation schématique et codifiée d’un algorithme.



Exemple :

Algorithme de déplacement automatique d'un robot qui peut rencontrer un obstacle à moins de 2 cm.



A toi de jouer !

Voici la description du comportement d'un robot :

Si le robot ne détecte pas d'obstacle à moins de 2 cm, celui-ci doit continuer d'avancer, sinon le robot pivote à gauche puis avance de nouveau.

Consignes :

- Traduire cette phrase en langage algorithmique dans livret élève.
- Construire l'algorithme associé dans le livret élève.

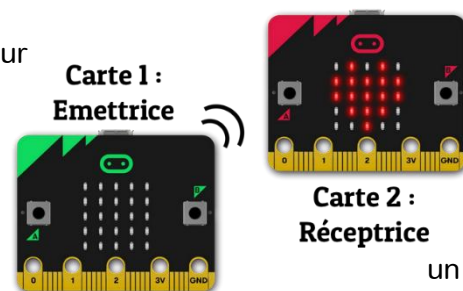
Activité 2 – Communication à distance entre cartes

Pour pouvoir échanger des données entre la Terre et le robot sur Mars, il faut réussir à transmettre des informations à distance. Celles-ci peuvent être des commandes de déplacements, des images, du son...

Le but de l'activité est de programmer une carte « émettrice » et une carte « réceptrice » tout en respectant cahier des charges.

Cette programmation se fera en Python.

Attention, si en général la boucle infinie est interdite en programmation celle-ci peut être judicieuse pour un système embarqué. On commencera donc par importer les bibliothèques et une boucle while TRUE :

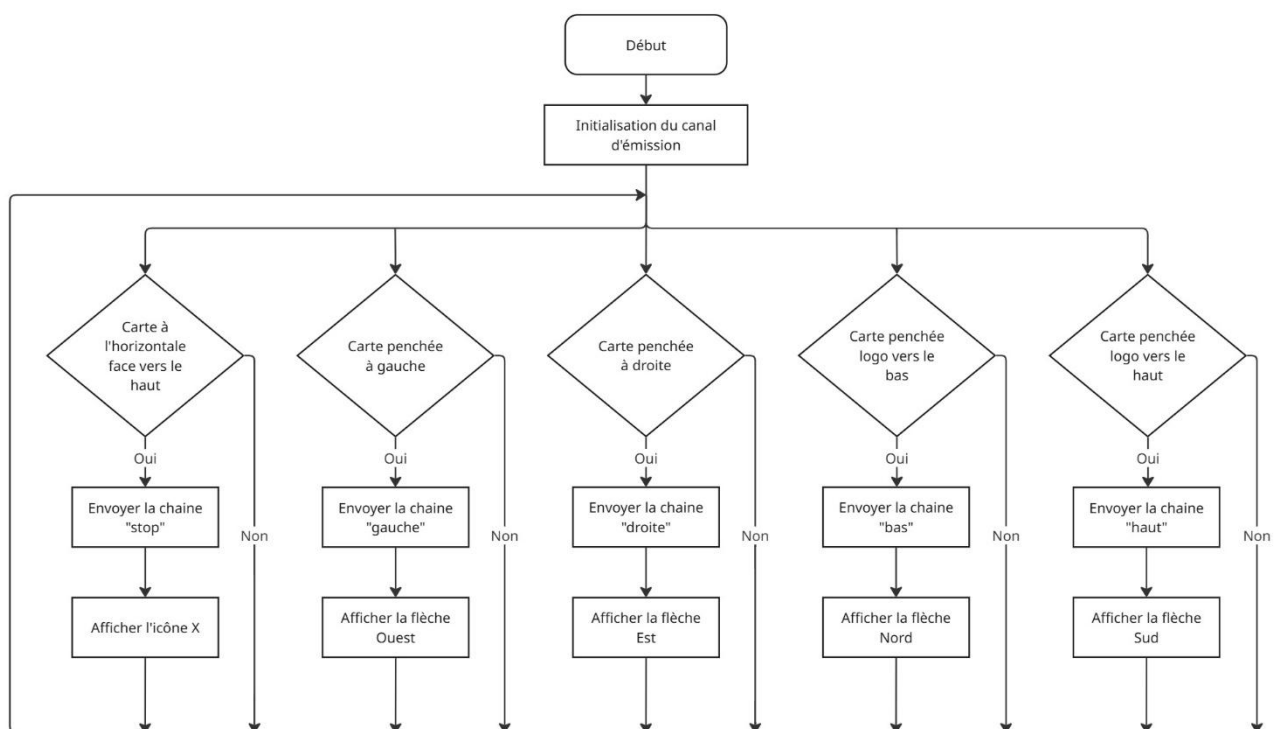


Document 1 – Code utile

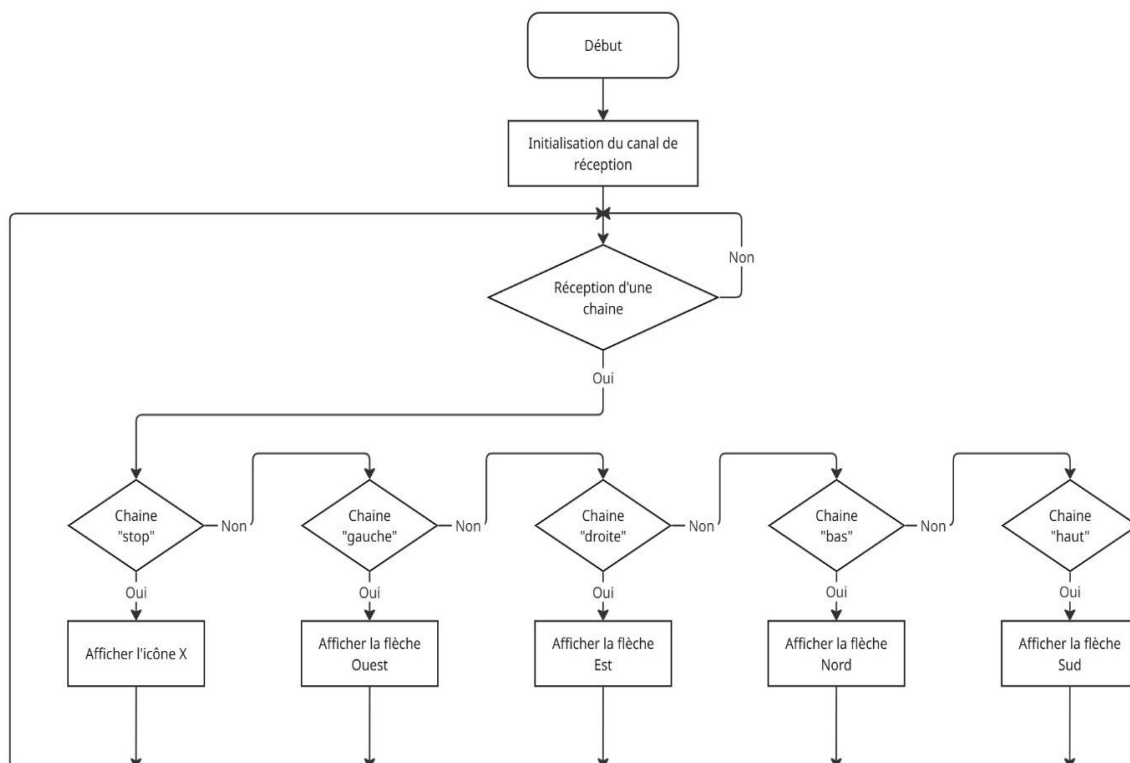
Code	Description
<code>from microbit import *</code>	Importe la bibliothèque microbit. Doit obligatoirement être en tête de programme.
<code>import radio</code>	Importe la bibliothèque radio. Doit obligatoirement être en tête de programme.
<code>radio.config(group=1)</code>	Définit le canal de communication et ses paramètres. A ajuster dans le cadre de plusieurs groupes d'élèves (group=xx par émetteur/récepteur).
<code>radio.on()</code>	Active le module radio.
<code>radio.send()</code>	Envoie un message sur le canal.
<code>radio.receive()</code>	Tente de recevoir un message et le stocke
<code>display.show(Image.NO)</code>	Permet d'afficher une croix sur la matrice de LED de la microbit.
<code>display.show(Image.ARROW_N)</code>	Permet l'affichage d'une flèche spécifique : • N : Nord • S : Sud • W : Ouest • E : Est
<code>accelerometer.current_gesture()</code>	Renvoie une chaîne de caractères indiquant le geste effectué : • shake : secouer • up : logo vers le haut • down : logo vers le bas • left : bascule de la carte vers la gauche • right : bascule de la carte vers la droite • face up : carte à l'horizontale face vers le haut

Plus d'informations sur la librairie micro:bit [ici](#).

Document 2 - Algorithme de la carte « Emettrice »



Document 3 - Algorithme de la carte « Réceptrice »



Consignes :

Chaque groupe aura un canal dédié défini par l'enseignant pour éviter les conflits de communication.

A partir des documents précédents, vous allez programmer les deux cartes.

Pour cela, ouvrir le mode multi en cliquant sur le lien : [Mode Multi](#)

⚠ Nous vous conseillons d'utiliser un navigateur qui accepte la communication avec les cartes micro:bit.

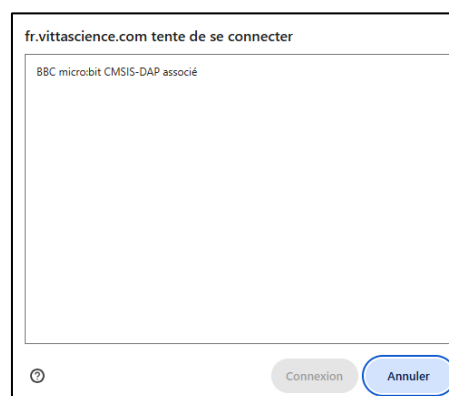
- Dans la partie gauche, il faut programmer la carte « Emettrice ».
- Dans la partie droite, il faut programmer la carte « Réceptrice ».
- Connecter la carte « émettrice » à l'ordinateur à l'aide du câble USB

- Cliquer sur  dans la partie gauche.
- Sélectionner “BBC micro:bit CMSIS-DAP” puis cliquer sur “se connecter”.

- Déconnecter la carte.

- Connecter la carte « réceptrice » à l'ordinateur à l'aide du câble USB

- Cliquer sur  dans la partie droite.
- Sélectionner “BBC micro:bit CMSIS-DAP” puis cliquer sur “se connecter”.
- Alimenter les deux cartes.



Observer le comportement.

Activité 3 – Contrôle du robot à grande distance

L'objectif de l'activité est de simuler le comportement d'un robot contrôlé à grande distance.

Il faudra garder un regard critique sur la situation.

Consignes :

Ouvrir le mode multi en cliquant sur le lien : [Mode Multi](#)

Dans la partie gauche, il faut ouvrir le fichier "télécommande Mars" :

- Cliquer sur  ou  puis "ouvrir projet"
- Puis taper "télécommande Mars" dans la barre de recherche et dérouler les "projets partagés"

Dans la partie droite, il faut ouvrir le fichier "Robot Commandé" :

- Cliquer sur  ou  puis "ouvrir projet"
- Puis taper "Robot Commandé" dans la barre de recherche et dérouler les "projets partagés"

Ensuite, simuler le comportement des deux cartes en **cliquant** sur les deux boutons .

⚠ Un rafraichissement de la page peut être nécessaire.

Questions : (répondre dans le livret élève)

1. Quelle difficulté rencontrez-vous lors de la simulation ?
2. Pourquoi a-t-on ajouté un délai de 3s avant l'envoi de chaque commande ?
3. Cette durée vous semble-t-elle cohérente avec la situation de contrôle d'un robot sur Mars à partir de la Terre ?

Vérifiez votre hypothèse par le calcul.

Application : (répondre dans le livret élève)

La distance entre Mars et la Terre selon son orbite est de 55,7 à 401 millions de kilomètres.

Les ondes radio se déplacent à une vitesse proche de celle de la lumière, qui est de 300 000 kilomètres par seconde dans le vide.

1. Quelle est la durée minimale pour qu'un signal radio émis à la surface de la Terre arrive jusqu'à la surface de Mars ? Exprimer le résultat dans une unité adéquate.
2. Quelle est la durée maximale pour qu'un signal radio émis à la surface de la Terre arrive jusqu'à la surface de Mars ? Exprimer le résultat dans une unité adéquate.

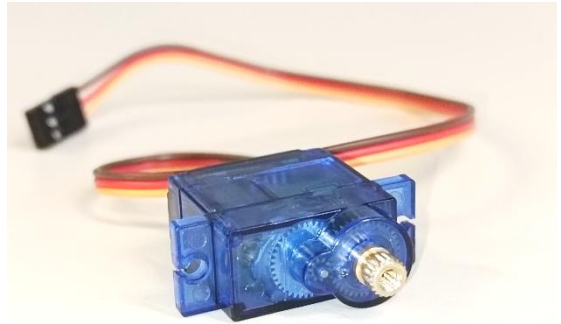
Activité 4 – Contrôler un servomoteur

Un robot d'exploration peut être équipé de caméras, de pinces ou d'autres détecteurs qui sont placés sur des bras commandés.

Ces bras sont commandés par des servomoteurs.

Document 1 – Qu'est-ce qu'un servomoteur ?

Un servomoteur est un système motorisé capable d'atteindre des positions prédéterminées extrêmement précises, puis de les maintenir. Celui que nous utiliserons est un servomoteur à moteur rotatif, la position de celui-ci est déterminée une valeur d'angle en degré comprise entre 0° et 180°. Il permettra d'activer une pince à l'avant de notre robot.



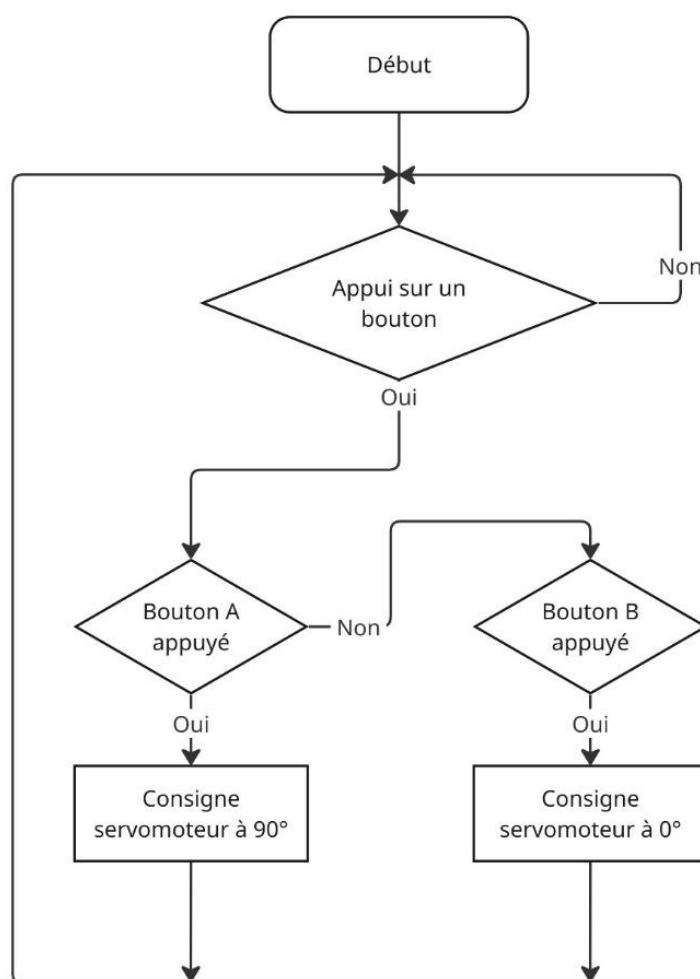
Quelques usages des servomoteurs :

- Barrière automatique
- Automatisation des processus dans les lignes de production
- Modélisme miniature, comme les petits robots ou avions télécommandés
- Instruments médicaux
- Contrôle de vannes
- Positionnement de miroirs ou d'antennes
- Danse avec les robots

Document 2 – Lignes de code utiles

```
if button_a.is_pressed(): #Effectue une commande si le bouton A est appuyé
maqueenplusv2.set_servo_angle(pin1, 90) # Positionne le servo sur la broche P1 à 90 degrés,
pincen fermées (attention, pas 90 ? ici le bloc refait quelque chose déjà présent dans la
bibliothèque)
```

Document 3 – Algorithme de contrôle du servomoteur



Consignes :

A partir des documents précédents, vous allez programmer la carte qui sera associée à un servomoteur.

Pour cela, ouvrir la page de programmation de Vittascience en cliquant sur le lien : [Programmer](#)

Puis cliquer sur **Maqueen Plus**.



Programmer votre carte et vérifier son comportement grâce à la simulation .

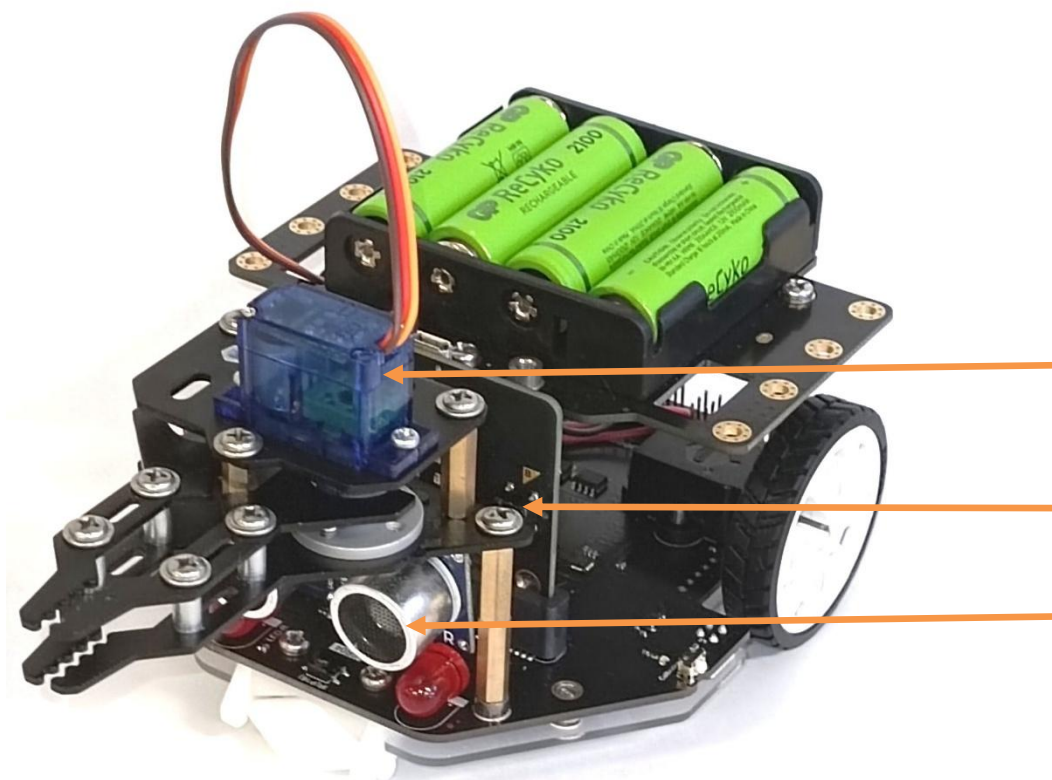
Conclusion

Consigne : (répondre dans le livret élève)

A partir des notions abordées dans les activités 1 à 4, quelles difficultés pourrait-on rencontrer à la réalisation d'une telle mission ?

Annoter la présentation suivante et reconnaître les capteurs et actionneurs ajoutés sur le robot Maqueen Plus V2 ([aide](#)).

Présentation du robot utilisé au Campus Numeria (Futuroscope)





Deuxième temps du parcours – Activités au Campus Numeria

Activité 1 – Pilotage à grande distance

Lors du premier temps du parcours vous avez simulé une télétransmission de commandes de déplacement à grande distance.

Vous allez tester en réalité cette situation pour mieux appréhender les difficultés.

Consignes :

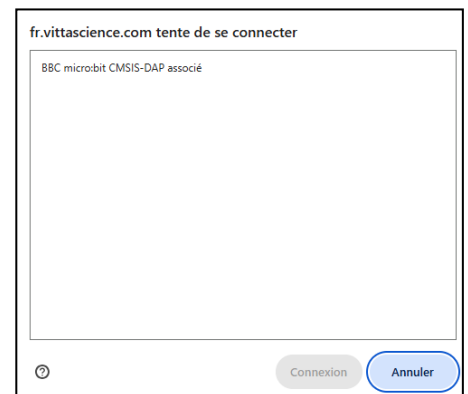
Ouvrir le mode multi en cliquant sur le lien : [Mode Multi](#)

Dans la partie gauche, il faut ouvrir le fichier “télécommande Mars” :

- Cliquer sur  ou  puis “ouvrir projet”
- Puis taper “télécommande Mars” dans la barre de recherche et dérouler les “projets partagés”

Dans la partie droite, il faut ouvrir le fichier “Robot Commandé” :

- Cliquer sur  ou  puis “ouvrir projet”
- Puis taper “Robot Commandé” dans la barre de recherche et dérouler les “projets partagés”
- Modifier le canal de communication (celui-ci correspond au numéro indiqué sous le robot)
- Connecter la carte « télécommande » à l’ordinateur à l’aide du câble USB
- Cliquer sur  dans la partie gauche.
- Sélectionner “BBC micro:bit CMSIS-DAP” puis cliquer sur “se connecter”.
- Déconnecter la carte et l’alimenter
- Connecter la carte « robot » à l’ordinateur à l’aide du câble USB
- Cliquer sur  dans la partie droite.
- Sélectionner “BBC micro:bit CMSIS-DAP” puis cliquer sur “se connecter”.
- Déconnecter la carte et l’insérer dans le robot.
- Alimenter le robot
- Déplacer les robots dans l’arène pour former un carré.

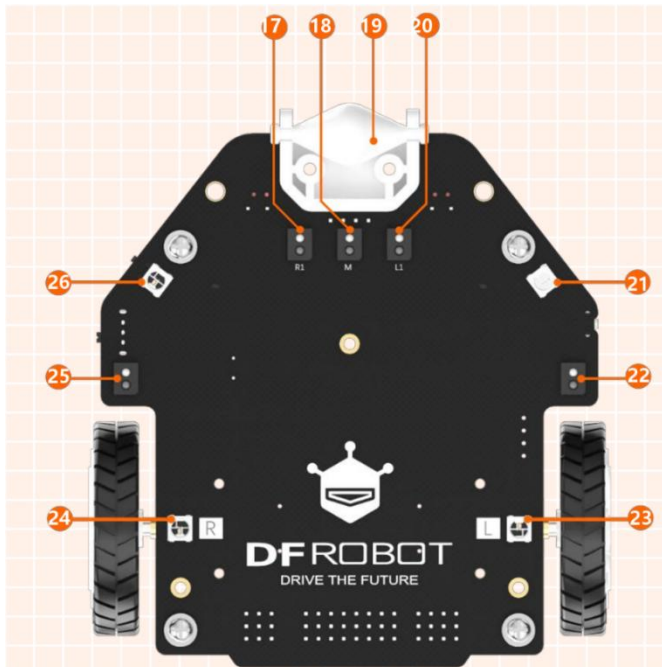


Activité 2 - Pilotage automatique par suiveur de ligne

Pour répondre aux difficultés rencontrées, nous allons utiliser un système de guidage autonome pour suivre un parcours défini à l'avance.

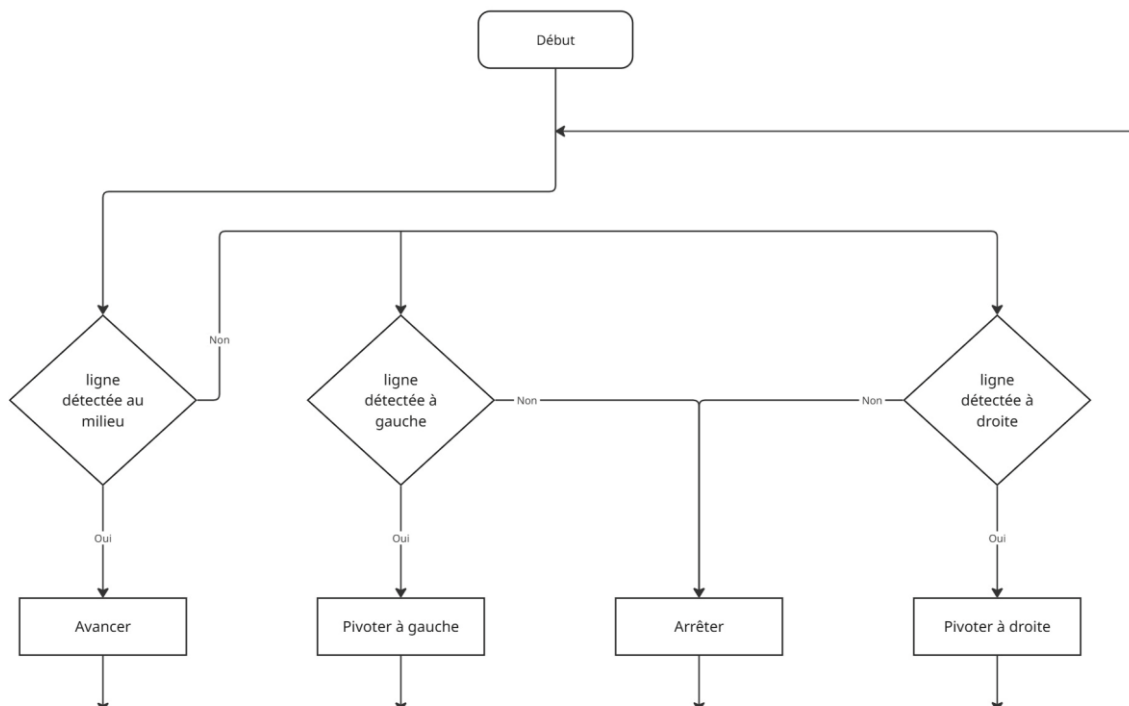
Sur Mars cela pourrait se faire par signaux radio.

Document 1 – Présentation du dispositif de suivi de ligne



- 17 – Capteur de ligne « droit » R1
- 18 – Capteur de ligne « milieu » M
- 19 – Roue d'appui
- 20 – Capteur de ligne « gauche » L1
- 21 – LED RGB 0
- 22 – Capteur de ligne « arrière gauche »
- 23 – LED RGB 1
- 24 – LED RGB 2
- 25 – Capteur de ligne « arrière droit »
- 26 – LED RGB 3

Document 2 - Algorithme



Consignes :

A partir des documents précédents, vous allez programmer la carte qui sera associée au robot pour que celui-ci se déplace automatiquement sur la ligne noire de la piste.

Pour cela, ouvrir la page de programmation de Vittascience en cliquant sur le lien : [Programmer](#)
Puis cliquer sur **Maqueen Plus**.

Ouverture du fichier “Suivi de ligne” :



- Cliquer sur  ou  puis “ouvrir projet”
- Puis taper “Suivi de ligne” dans la barre de recherche et dérouler les “projets partagés”
- Compléter le code en remplaçant les « ???? ».
- Téléverser pour tester.

Vous pouvez faire autant d’essais que nécessaire pour vérifier votre programme et régler la vitesse.

Remarque :

- Un bouton nommé « Calibration Key » sur le dessus du robot peut être utilisé pour définir la couleur de la ligne.
- Seuls les capteurs de ligne à l’avant seront utilisés.
- Une vitesse trop importante rend le comportement moins réactif.

Activité 3 – Fonctions

Mission 1 : Tri manuel

Lors de l’exploration, un intérêt particulier est porté sur la constitution de la matière (présence d’eau, roche calcaire...).

Nous allons coder une carte pour comprendre le fonctionnement de base.

Deux types d’objets existent :

- Ceux qui ont un intérêt (qui contiennent un aimant) ;
- Ceux qui ne suscite aucun intérêt (qui ne contiennent pas d’aimant).

Document 1 – Code spécifique

On pourra tester la pince avant de commencer à programmer

```
from microbit import *    # Importe les fonctionnalités de base
import maqueenplusv2      # Importe la bibliothèque du robot
# --- Configuration Matérielle ---
# SERVO de la pince sur le port P1 ou SP2 (PAS P0)
SERVO_PIN = pin1 #ici branché sur pin1 sur le robot
while True:
    for i in range(0,40,10): # de l'angle 0 à 40 par pas de 10
        maqueenplusv2.set_servo_angle(SERVO_PIN, i) # <-- ICI ON OUVRE LA PINCE AU DÉPART
        sleep(1000)
        display.scroll(i) # on affiche l'angle
```

Puis ensuite on peut mesurer le champ magnétique ambiant avec

```
get_magnetic_strength()
```

Pour mesurer la distance, on utilise le capteur ultra-son dont on propose ici une définition de fonction

```
def read_distance_cm():
    """ Mesure et retourne la distance en cm du capteur à ultrasons. """
    ULTRASONIC_TRIG.write_digital(0); sleep_us(2)
    ULTRASONIC_TRIG.write_digital(1); sleep_us(10)
    ULTRASONIC_TRIG.write_digital(0)
    duration = time_pulse_us(ULTRASONIC_ECHO, 1, 30000)
    return duration*0.0343/2
```

Donc on utilisera les opérateurs booléens, ici AND, pour vérifier que deux conditions sont vraies

Remarque :

Le temps mesuré est la durée aller-retour, donc la distance totale parcourue par le son est :

Distance totale = vitesse du son × temps

La vitesse du son dans l'air est d'environ 343 m/s, soit 0.0343 cm/μs.

Donc :

Distance totale (aller-retour) = 0.0343 (cm/μs) × durée (μs)

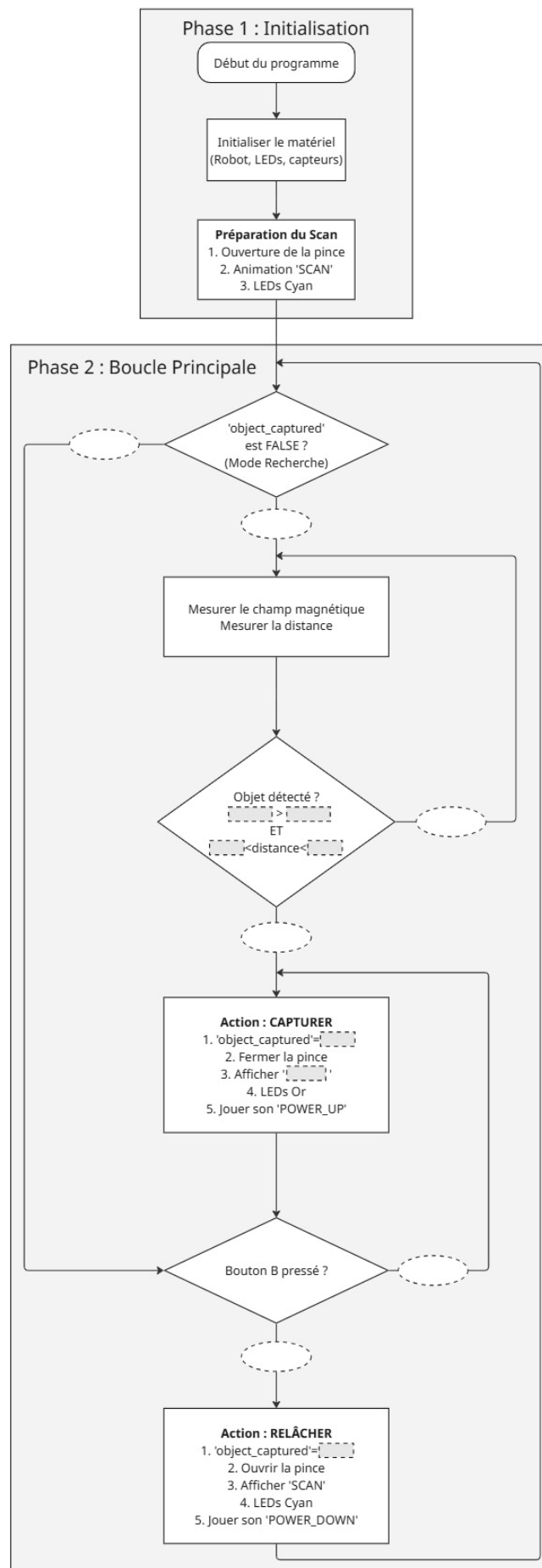
Mais on veut la distance allée simple (du capteur à l'objet), donc on divise par 2.

Consignes :

Ouverture du fichier “Tri par le robot” :

- Cliquer sur  ou  puis “ouvrir projet”
- Puis taper “Tri par le robot” dans la barre de recherche et dérouler les “projets partagés”
- Téléverser pour tester.

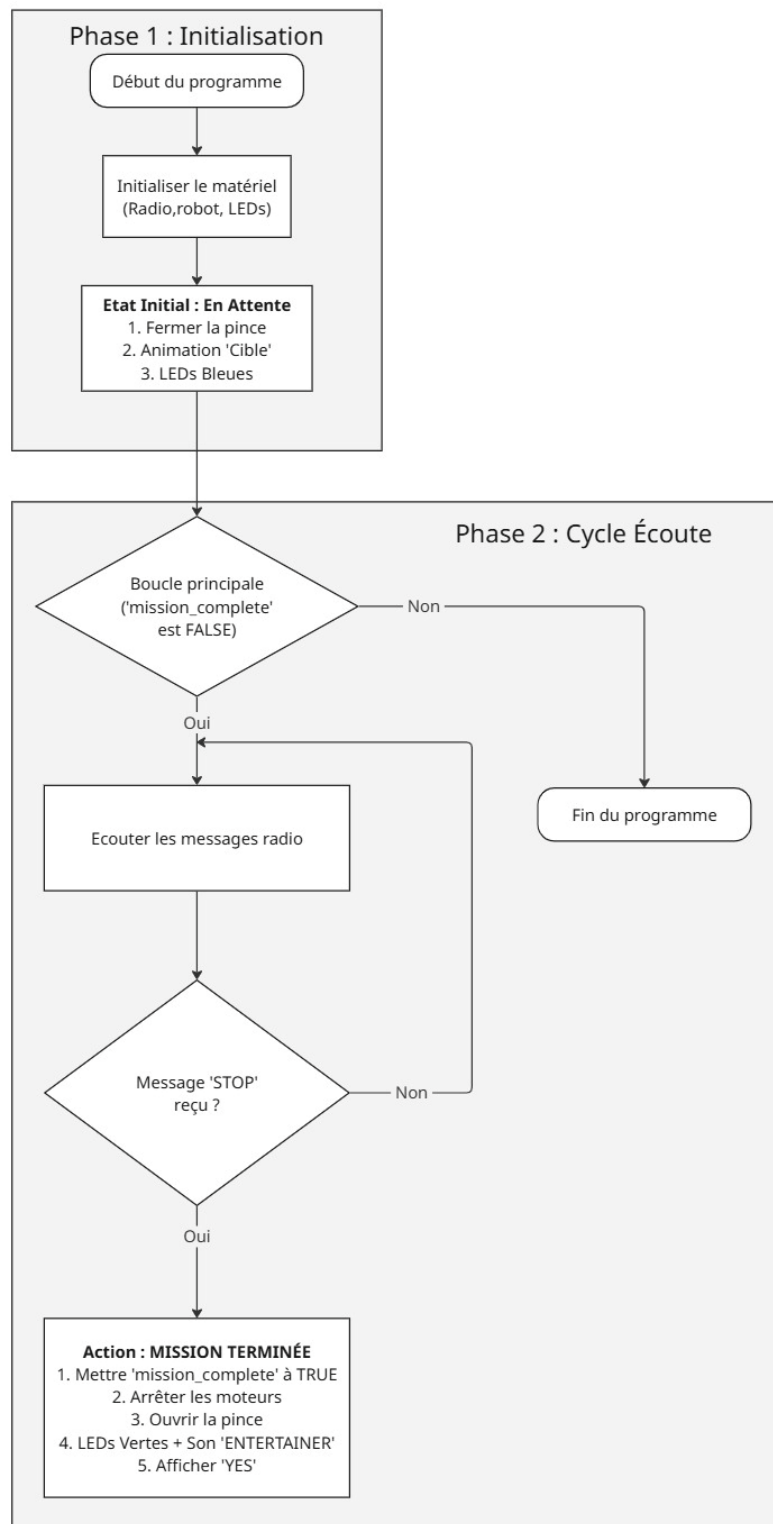
Dans le livret élève, compléter les zones en pointillés de l'algorithme suivant à partir du code et de vos observations.



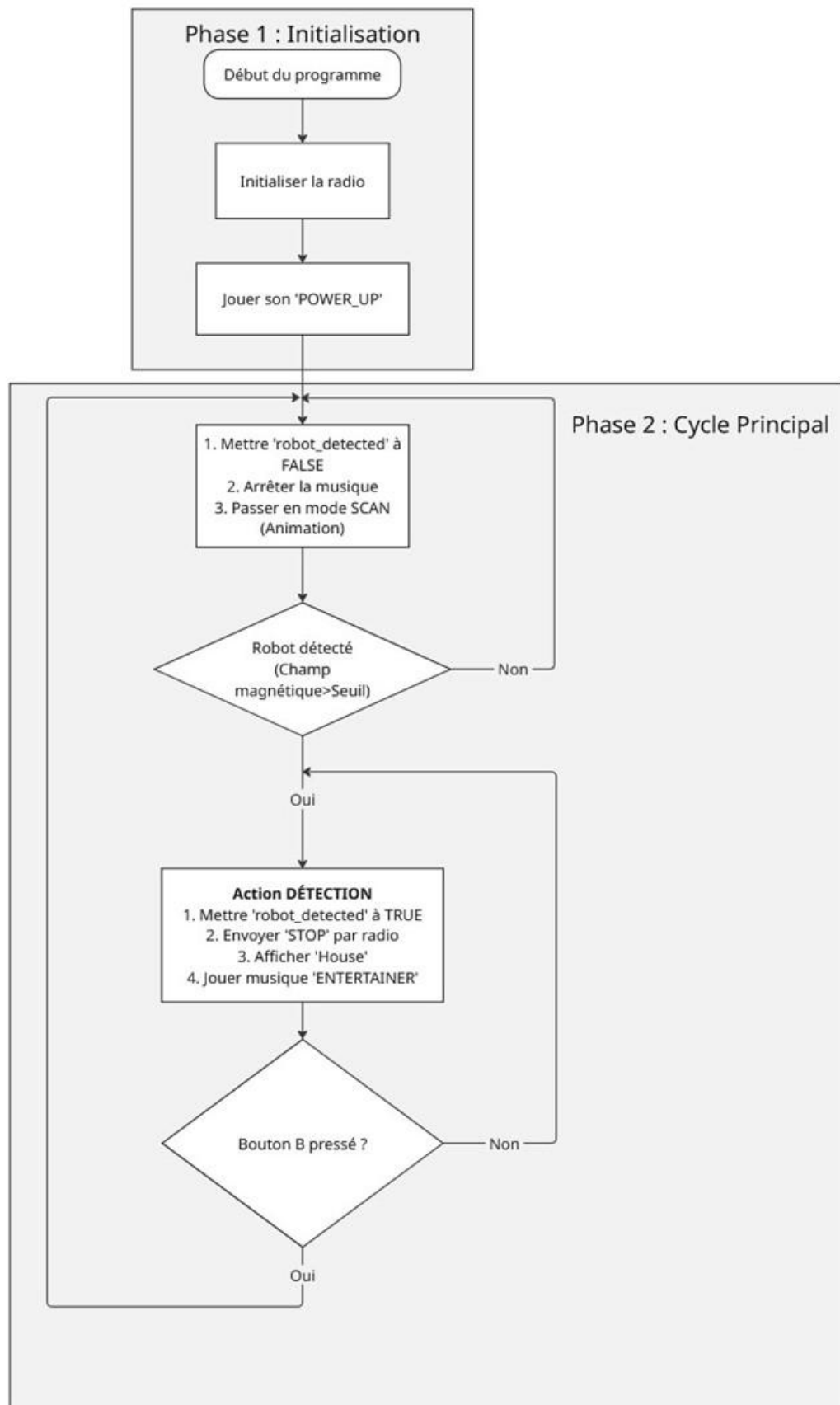
Mission 2 : Préparation du bagage pour le retour

Lorsque le robot retourne à la base en transportant un objet contenant un aimant, celui-ci continue son chemin jusqu'à ce que la base le prévienne de sa proximité.

Document 1 – Algorithme « robot »



Document 2 – Algorithme « base »



Consignes :

Ouvrir le mode multi en cliquant sur le lien : [Mode Multi](#)

Dans la partie gauche, il faut ouvrir le fichier “base Mars Python” :

- Cliquer sur  ou  puis “ouvrir projet”
- Puis taper “base Mars Python” dans la barre de recherche et dérouler les “projets partagés”

Dans la partie droite, il faut ouvrir le fichier “robot Mars Python” :

- Cliquer sur  ou  puis “ouvrir projet”
- Puis taper “robot Mars Python” dans la barre de recherche et dérouler les “projets partagés”

Modifier le canal de communication (celui-ci correspond au numéro indiqué sous le robot)

- Compléter les codes en remplaçant les « ???? ».
- Téléverser pour tester.

Activité 4 – Tout en un

Il est temps d’explorer Mars !

Les différentes parties abordées ont été reprises pour réaliser un programme complexe.

Consignes :

Ouvrir le mode multi en cliquant sur le lien : [Mode Multi](#)

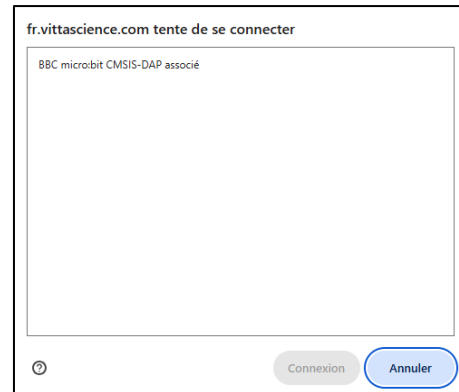
Dans la partie gauche, il faut ouvrir le fichier « base mission Mars Python»

- Cliquez sur  ou  puis “ouvrir projet”
- Puis taper “base mission Mars Python” dans la barre de recherche et dérouler les “projets partagés”

Dans la partie droite, il faut ouvrir le fichier « Mission Mars Python»

- Cliquez sur  ou  puis “ouvrir projet”
- Puis taper “Mission Mars Python” dans la barre de recherche et dérouler les “projets partagés”
- Modifier le canal de communication dans les deux programmes (celui-ci correspond au numéro indiqué sous le robot)
- Connecter la carte « robot » à l’ordinateur à l’aide du câble USB

- Cliquer sur  dans la partie gauche.
- Sélectionner “BBC micro:bit CMSIS-DAP” puis cliquer sur “se connecter”.
- Déconnecter la carte et l’insérer dans le robot.
- Connecter la carte « base » à l’ordinateur à l’aide du câble USB



- Cliquer sur  dans la partie droite.
- Sélectionner “BBC micro:bit CMSIS-DAP” puis cliquer sur “se connecter”.
- Vous pouvez commencer les tests sur le plateau en disposant les différents objets sur le chemin.
Attention les objets doivent être assez espacés les uns des autres.

Vous pouvez modifier les valeurs des variables dans la section « Au démarrage » pour adapter le comportement du robot.

Observations et regard critique :

- Quelles améliorations pourraient être apportées à votre robot ?
- Qu’est-ce qui fonctionne bien ? ne fonctionne pas ?
- Est-ce qu’un système 100% automatisé est envisageable ? ...



Troisième temps du parcours – Le retour sur Terre

Retour sur votre expérience

Chaque groupe présente à la classe les résultats obtenus pendant le deuxième temps du parcours.

A vous de jouer !

Regarder cette [vidéo](#).

A partir de celle-ci vous allez devoir :

- Traduire son comportement en phrases courantes dans le livret élève.
- Faire de même en langage algorithmique dans le livret élève.
- Construire l'algorithme associé dans le livret élève.

Etude de Sojourner et de l'exploration du sol de Mars

Le robot Sojourner est arrivée le **4 juillet 1997**.



Crédits : NASA/JPL

Durant 95 jours terrestres, ce rover a pu réaliser 16 analyses (roches et sol) et des expérimentations de mécanique des sols (abrasion des roues...).

Mais d'autres robots mobiles ont aussi exploré Mars :

- Spirit (NASA), 4 janvier 2004. Bloqué en 2009, sa mission s'achève en 2010.



Crédits : NASA/JPL/Cornell University

- Opportunity (NASA), 25 janvier 2004. Fin de mission en 2018 pour cause de tempête de poussière.

Ces deux derniers robots sont jumeaux et ont eu une durée de fonctionnement bien supérieure à celle prévue qui devait être de 90 jours. Cela a permis une exploration bien plus vaste du sol.

- Curiosity (NASA), 6 août 2012 et toujours actif.



Crédits : NASA/JPL-Caltech/MSSS

Curiosity n'est plus un simple robot explorateur martien (MER) mais une mini laboratoire alimenté par un générateur nucléaire contrairement aux précédents qui étaient alimentés par panneaux solaires.

- Perseverance (NASA), 18 février 2021 et toujours actif.

Perseverance est un robot similaire à Curiosity. Il effectue sa mission dans une autre région de la surface martienne.

- Zhurong (CNSA), 14 mai 2021, en sommeil depuis mai 2022.



Crédits : Laurent Garcia/Cité de l'espace

Zhurong est le premier rover martien chinois (il est 25 % plus petit que Curiosity). La Chine devient le deuxième pays à réussir un atterrissage martien et à établir une communication. À la suite d'une tempête de poussière pendant son hibernation, celui-ci n'a jamais pu être réveillé.



Crédits : NASA/JPL-Caltech

Des maquettes grandeur nature de trois générations de rovers martiens de la NASA illustrent l'augmentation de leur taille, du rover Sojourner (projet Mars Pathfinder) au centre, aux rovers jumeaux Spirit et Opportunity à gauche, jusqu'au rover Curiosity.

Embarquant des outils de mesure scientifique plus perfectionnés et une alimentation bien supérieure, ces robots peuvent explorer de plus vastes étendues et ainsi collecter des données plus riches.

Pour comparaison, Sojourner n'a parcouru qu'une centaine de mètres, alors que Perseverance avait parcouru plus de 25 km en mars 2024.

Contrôler un bras

Toute évolution des robots martiens entraîne de nouveaux défis et une précision de plus en plus grande.

Le bras de prélèvement de Perseverance nécessite d'utiliser des servomoteurs.

Si vous le souhaitez, nous proposons un parcours avec un bras analogue totalement contrôlable à distance. Les mouvements que vous programmez depuis votre classe sont envoyés au bras situé au Campus Numeria (Futuroscope). Les images sont retransmises en direct sur votre écran.

Concours robotique

Si vous avez apprécié ce parcours et que vous désirez aller plus loin, il existe des concours robotiques accessibles à tous. Pourquoi ne pas participer à l'un d'entre eux avec votre propre robot ?