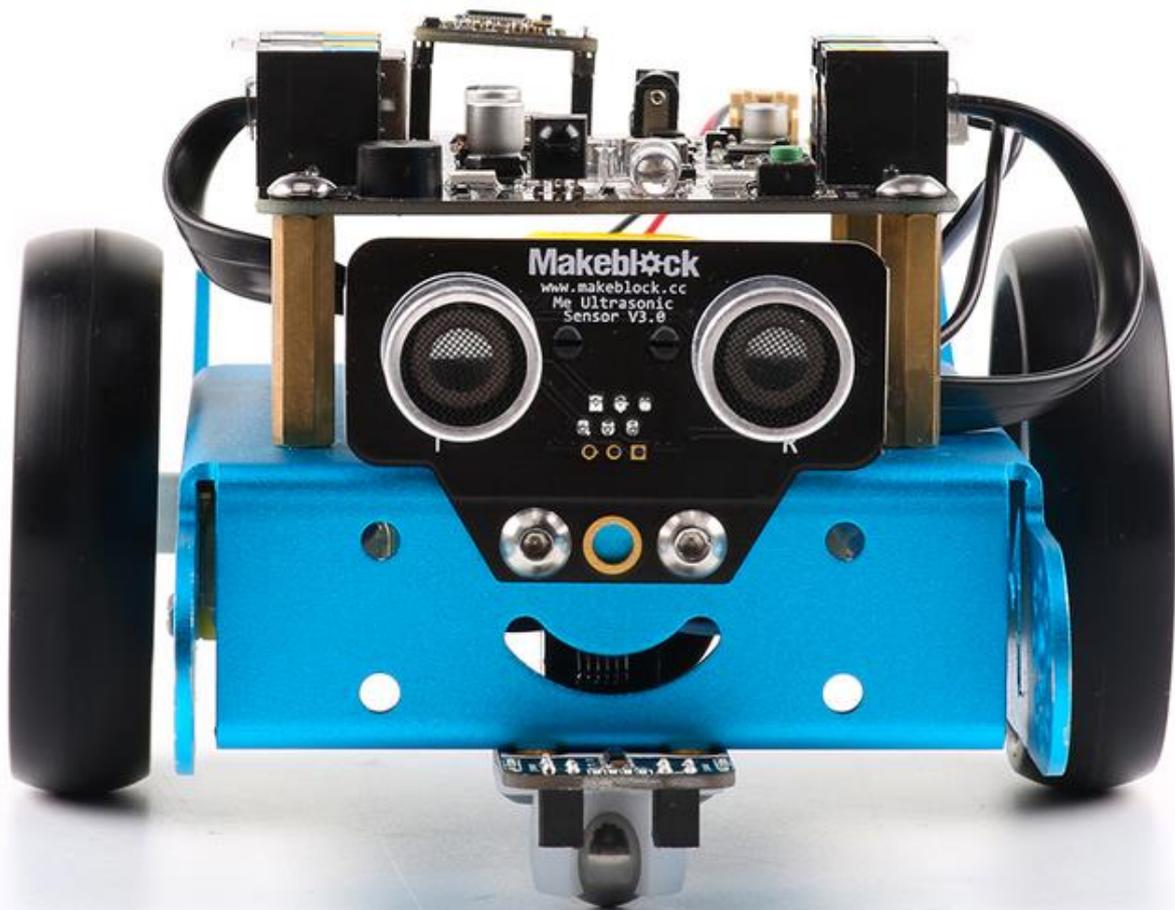


Le robot Mbot et sa programmation avec mBlock

F.Prieur

Collège de Roujan

Edition 2019-2020



Ce document vous permettra de découvrir le robot mBot et d'acquérir les bases de la programmation de ses capteurs et actionneurs avec l'outil logiciel mBlock dérivé de Scratch.

Au-delà de son utilisation de base, deux manipulations plus conséquentes sont proposées autour de mBot et peuvent être travaillées dans le cadre d'un travail collectif de projet en fin de cycle 4 : sortie automatique d'un labyrinthe et suivi de ligne « intelligent »

1/ Présentation de mBot :

Il s'agit d'un robot mBot du commerce. Il est doté d'une carte électronique programmable. Celle-ci peut être programmée ou télé-opérée grâce à un ordinateur soit par fil (port USB), soit par Wifi. On utilisera le logiciel « mBlock » pour créer nos programmes. Le langage graphique utilisé est Scratch.



Le robot **mBot** interagit avec son environnement en fonction du programme qu'on lui implante. Pour cela, il est capable de collecter des informations grâce **à ses capteurs** et de réaliser des actions grâce **à ses actionneurs**.

Actions et actionneurs :

- le robot vendu de base, est capable de **se déplacer** : il est équipé de **deux moteurs** indépendants reliés chacun à une roue (qui devient donc **une roue motrice**).
- il peut **émettre des sons** grâce à un **buzzer**.
- il peut **émettre de la lumière** grâce à **2 DEL 3 couleurs** (RGB) dont la couleur est paramétrable. -d'autres actionneurs peuvent être branchés **en option** (afficheur 128 leds, motoréducteur, blocs 4 leds, afficheur 7 segments...).

Boutons et capteurs :

Pour interagir avec son environnement et y recueillir des informations, on retrouve sur le robot :

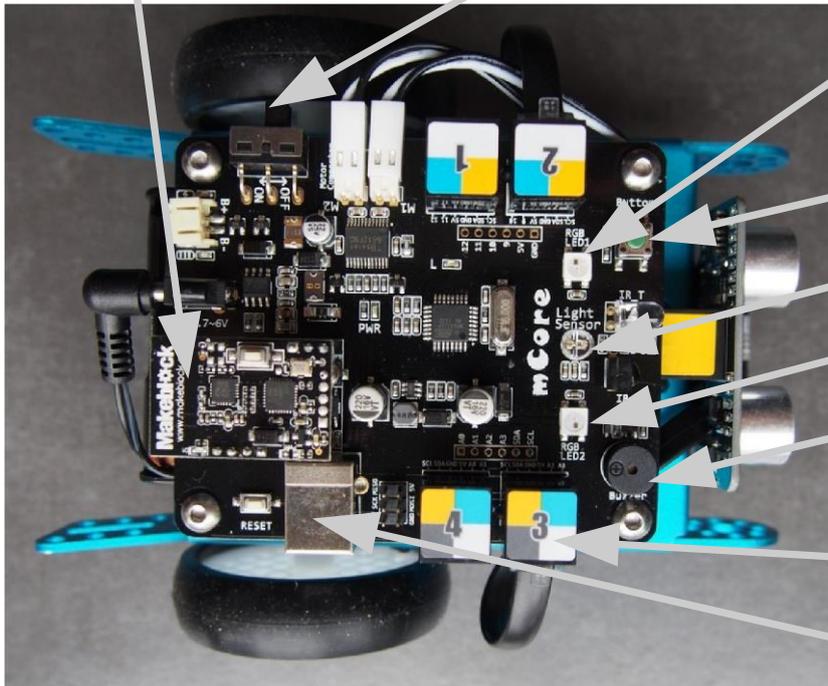
- un **capteur de luminosité** qui le renseigne sur la luminosité ambiante.
- un **module à ultrasons** qui lui permet de « voir » les obstacles à l'avant et d'en connaître la distance. un **module de suivi de ligne au sol** à infrarouge.
- un **bouton** paramétrable.
- un **bouton de mise sous tension**.
- d'autres capteurs peuvent être branchés **en option** (humidité, flamme, fumée, gyroscope...)



LOCALISER LES CAPTEURS ET ACTIONNEURS

Carte Wifi

Bouton de mise sous tension



LED(3 couleurs) RGB1

bouton

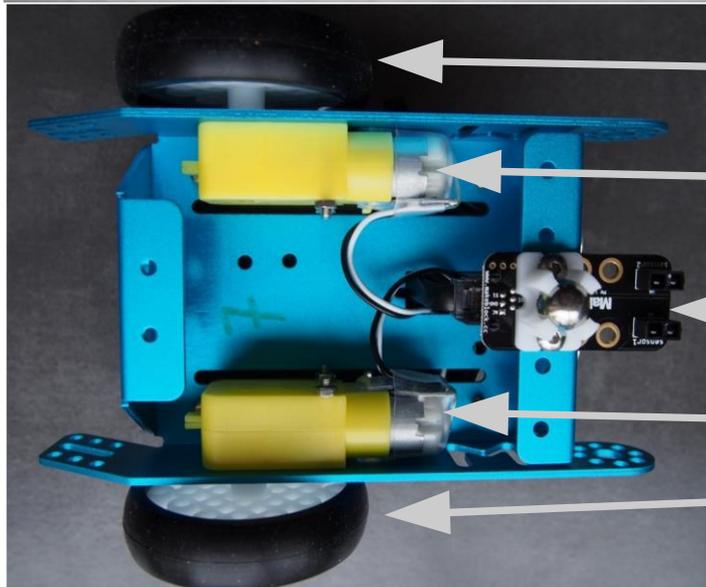
Capteur de luminosité

LED (3couleurs) RGB2

Buzzer

Prises pour capteurs
ou actionneurs
optionnels

Prise USB



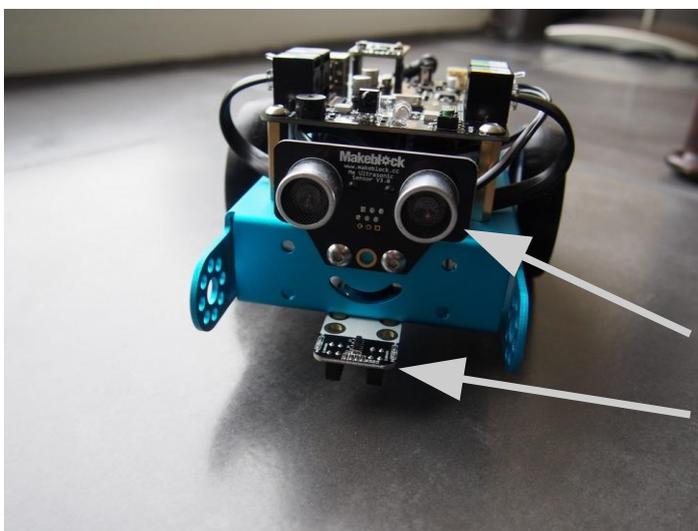
Roue motrice droite

Moteur de la roue droite

Module de suivi de ligne
à infrarouge

Moteur de la roue gauche

Roue motrice gauche



Module à ultrasons

Module de suivi de ligne
à infrarouges

2-Le logiciel mBlock

mBlock est une extension de Scratch laissant apparaître une palette supplémentaire dédiée aux extensions matérielles (mBlock, Arduino...). Vous noterez que vous retrouvez toutes les palettes habituelles de Scratch. On peut donc faire du Scratch « pur » sous mBlock.

1- Lancez le logiciel mBlock qui se trouve sur votre bureau



2- Dans le menu « choix de la carte », sélectionnez **mBot** puis dans « Choix des extensions », sélectionnez parmi les différents choix possibles l'option **Makeblock**.



Pour un meilleur affichage, fermer cette fenêtre inutile ici (clic dessus)

Toutes les instructions disponibles de la bibliothèque Pilotage

Vous trouverez ici les différentes bibliothèques d'instructions pour concevoir vos programmes. En ce qui nous concerne, nous utiliserons essentiellement :

- Blocs et variables
- Evénements
- Contrôle
- Opérateurs
- Pilotage



3- Pour créer le programme, il suffit de **glisser les instructions** des éléments de la bibliothèque **dans la zone de script** tout comme vous le faisiez avec Scratch. Attention, il faut parfois les déposer avec précision pour que cela s'imbrique à la manière d'une pièce de puzzle.

Palettes courantes de mBlock

The image displays the mBlock software interface with several palettes highlighted by black boxes:

- Blocs & variables:** Located at the top left, containing blocks for creating variables, setting values, and displaying/hiding variables.
- Evènements:** Located at the top center, containing event-based blocks like 'when green flag clicked', 'when space key pressed', and 'when volume is greater than 10'.
- Contrôle:** Located at the top right, containing control blocks like 'wait 1 seconds', 'repeat 10 times', and 'if-then-else'.
- Opérateurs:** Located at the bottom right, containing mathematical and logical operators like '+', '-', '=', '>', '<', 'and', 'or', and 'not'.
- Pilotage:** Located at the bottom left, containing blocks for controlling the mBot robot, such as 'mBot - generate code', 'move at speed', 'activate motor', and 'set servo motor'.

A central text box with a white background and black border contains the text: **Palette spécifique à mBlock (absente de Scratch)**. Red arrows point from this box to the 'Evènements' and 'Pilotage' palettes.

3-Mbot en « mode connecté » :

Le **mode connecté** correspond au mode où le câble USB reste branché à mBot. Ce mode permet de télécommander mBot et/ou charger des programmes dans celui-ci.

En mode connecté, votre programme commencera par :

Il démarrera donc lorsque vous cliquerez sur le drapeau vert situé sur l'écran d'accueil.



1 . Pilotage en mode connecté

a. Connectez le robot à l'ordinateur à l'aide du câble USB

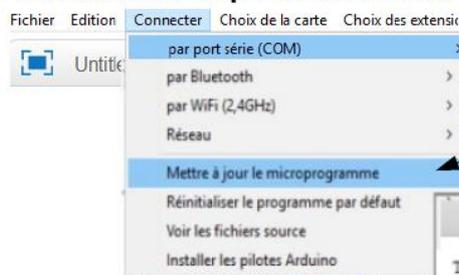


b. Allumer le robot mbot

c. Indiquez la carte à commander



d. Sélectionnez le port de communication



e. Attendre que la liaison soit établie

f. Mettre à jour le microprogramme de communication



g. Réaliser le programme



h. Lancer le programme



Le mode connecté permet de piloter le mBot en direct avec un ordinateur via une connexion USB ou sans fils...mais une fois déconnecté, le mBot restera inerte.

3 – Premiers programmes en « mode connecté »

1- Tout d'abord, **dans votre dossier personnel** (Mes documents), créez un **dossier « mBot »**

Tous vos programmes seront enregistrés dans ce dossier au fil de votre travail.

2- Créez le programme demandé dans mBlock en utilisant les palettes Scratch traditionnelles et la palette « Pilotage » dédiée à mBot.

3- Vérifiez sur le robot que le marche/arrêt de mBot est bien sur « **ON** ».

3- Assurez-vous que dans l'onglet « **Connecter** », « **par port série (COM)** » l'instruction « **COM...** » est bien activée. (En général, c'est la dernière « com » de la liste)

4- mBot émet un « bip » et la palette « Pilotage » affiche un point vert...tout cela est très bon signe.

Programme n°1 : faire clignoter (1s) une DEL RGB (1 ou 2) en rouge 10 fois de suite

Méthode	Éléments nécessaires
<p>Réaliser une boucle et mettre à l'intérieur :</p> <ul style="list-style-type: none"> Allumer les deux DEL Attendre 1s Eteindre les deux DEL Attendre 1s Programmer la boucle pour exécuter le programme 10 fois 	

Programme n°2 : Identique à P1 mais faire clignoter les 2 DEL RGB en alternance « façon pompiers » avec un « pin-pon » émis par le robot

Méthode	Éléments nécessaires
<p>Réaliser une boucle et mettre à l'intérieur :</p> <p>Allumer la DEL droite en rouge, éteindre la DEL gauche et jouer un do5 (C5) Attendre 1s Eteindre la DEL droite en rouge, allumer la DEL gauche et jouer un la5 (A5) Attendre 1s Programmer la boucle pour qu'elle répète le programme indéfiniment</p>	<p>Idem au programme 1 en utilisant cette fois :</p>

Programme n°3 : Utiliser une condition et le détecteur de lumière pour faire fonctionner le P2

Méthode	Éléments nécessaires
<p>Avant la boucle du P2, ajouter une condition Si (il se passe ceci) alors...</p> <p>Ici l'événement déclencheur est la baisse de la lumière mesurée sur la carte en dessous de 500 lux (à adapter à la luminosité de la salle...)</p>	

Programme P4 : remplacer la note par une petite mélodie !!!

Programme n°5 :

Il ne s'exécutera **qu'une seule fois** (à cause du « fil à la patte » qui limite le mouvement). Le robot avance durant 2 secondes à la vitesse 100. Puis il s'arrête.

Méthode	Éléments nécessaires
<p>Réaliser une boucle et mettre à l'intérieur :</p> <p>Avancer le robot à vitesse 100 Attendre 2s Avancer le robot à vitesse 0</p>	

4- mBot en « mode autonome » ou comment « Flasher » mBot étape par étape :

Cette fois-ci, nous allons « flasher » votre code en mémoire de mBot ce qui lui permettra de fonctionner de façon **autonome et sans plus aucune connexion au PC**. Le programme est mis en « mémoire flash » et est **conservé même si on éteint mBot**

Il suffira de remplacer dans votre script

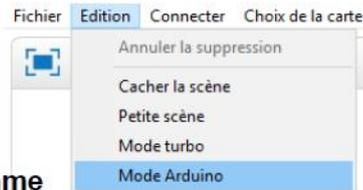


par



2 . Pilotage en mode automate

a. Passer en mode Arduino



b. Modifier la tête du programme



c. Téléverser le programme



d. Attendre la fin du téléversement

e. Débrancher le câble et vérifier le fonctionnement du robot

mBot - générer le code

Une fois le programme implanté dans la carte du robot, il est mémorisé et le robot exécutera toujours ce programme, jusqu'au prochain téléversement.



Une fois le transfert achevé, il s'exécutera selon vos instructions de programmation (ou bien directement soit par une manipulation sur le robot de votre part (appui sur le bouton poussoir par exemple...et si vous l'avez prévu dans votre programme)

Remarque : pour repasser plus tard en « mode connecté » il vous faudra « **télécharger le microprogramme de communication** » dans le menu « Connecter »

Dans les programmes suivants nous utiliserons le mode autonome pour que le robot soit libre de ses mouvements...mais nous aurions pu continuer en mode connecté.

Programme n°6 :

Dissocier la vitesse des moteurs ; Le robot tourne en rond 3s (sens horaire) puis 3s (sens trigonométrique) 2 fois de suite

Méthode	Éléments nécessaires
Réaliser une boucle avec : Faire tourner la roue gauche seule Attendre 3s Faire tourner la roue droite seule Attendre 3s	

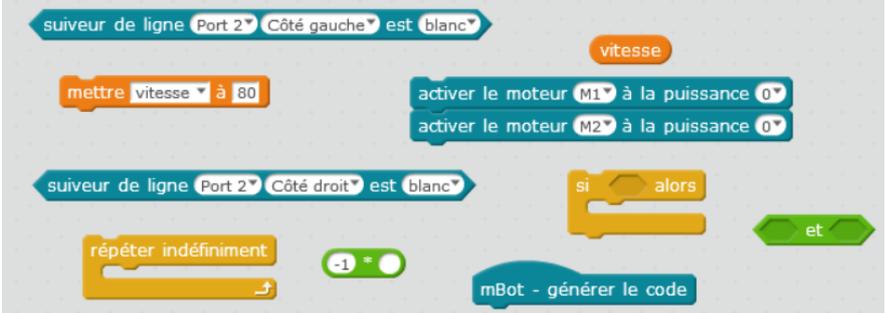
Programme n°7 :

Le robot avance s'il voit un obstacle devant lui, il recule, tourne à droite et reprend son avance ;

Méthode	Éléments nécessaires
Régler la vitesse à 50 (utiliser une variable) Réaliser une boucle infinie Si la distance > 50 alors avancer Si la distance ≤ 50 reculer pendant 0,5s puis tourner à droite pendant 0,5s Remarque : vérifiez si votre capteur US est bien sur le « Port 3 » sinon, modifier le « port »	

Programme n°8 :

Suivre une piste à l'aide du Module de suivi de ligne à infrarouge

Méthode	Éléments nécessaires
Régler la vitesse à 100 (utiliser une variable) Réaliser une boucle infinie Si blanc à droite à tourner vers la gauche Si blanc à gauche alors tourner vers la droite Si blanc à droite et gauche alors s'arrêter Sinon avancer droit	

Voilà, vous connaissez **tous les capteurs et actionneurs** de mBot, vous pouvez maintenant passer à des applications plus « évoluées » mêlant tous ces possibilités dans un fonctionnement autonome du robot.

Vous trouverez **d'autres propositions** d'applications sur mon site <http://www.fredtechnocollege.org>

Correction des programmes précédents

Programme n°1 :

Proposition de solution : Pas de soucis particulier pour ce programme. Ne pas oublier cependant le second « Attendre 1s » faute de quoi le vert sera extrêmement fugitif avant que ne recommence le programme avec le rouge.



Programme n°2 :

Proposition de solution : On pourrait par exemple écrire ce programme :

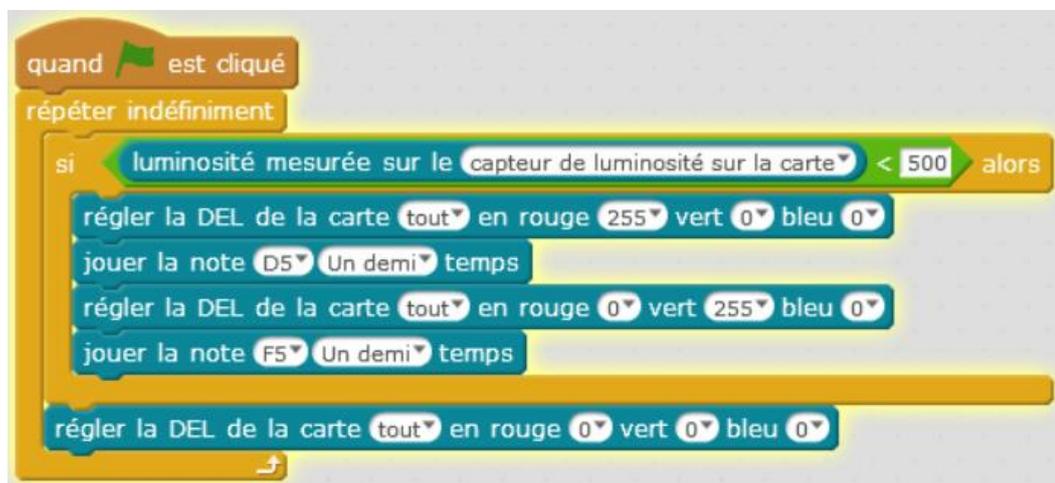


Noter que le « Attendre 1s » est superflu car le fait de jouer une note (ici sur un demi-temps soit une croche) suffit à temporiser le programme sans l'utilisation de délai supplémentaire.

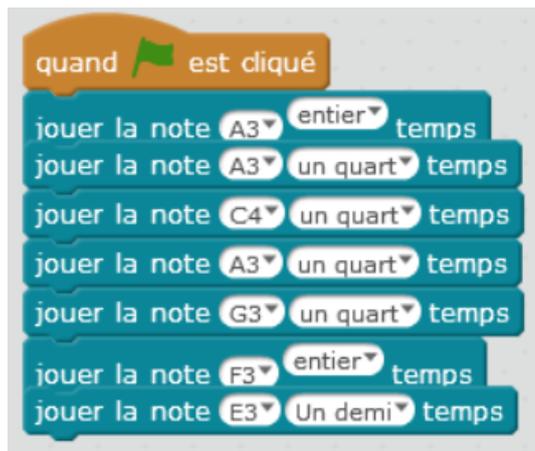
Programme n°3 :

Dans le programme que je propose, si la luminosité chute sous les 500, la sirène et les LED se mettent en route, sinon les LED restent éteintes.

Remarque : on peut isoler la barre « luminosité mesurée sur le capteur de luminosité sur la carte » et double-cliquer dessus afin que la valeur nous soit retournée. C'est très pratique pour déterminer le bon seuil à choisir. On observe classiquement une luminosité d'environ 1000 dans une pièce éclairée normalement, la luminosité chute vers 0 si on masque complètement le capteur de luminosité situé sur la carte (cf photo sur le sujet).



Programme P4 : remplacer la note par une petite mélodie !!!



Voici une mélodie bien connue sur les stades de foot ;-)
Elle est jouée une seule fois au clic du drapeau

Programme n°5 :



Proposition de solution : noter que si vous n'arrêtez pas le robot...il continue à avancer. « Stop tout » interrompt l'exécution du programme...mais n'arrête pas le robot

mBot en « mode autonome » ou comment « Flasher » mBot étape par étape :

Dans un cas général,

il suffira de remplacer dans votre script



par



Cependant, il vous faudra regrouper sous une seule bannière tous votre programme, on ne peut comme avec Scratch fragmenter en plusieurs sous-programmes...c'est parfois gênant et il faut souvent « ruser ».

A noter que les instructions du programme ne pourront évidemment pas faire référence au clavier, aux lutins... puisque le robot ne sera plus en lien avec le PC.

Remarque : pour repasser plus tard en « mode connecté » il vous faudra « **télécharger le microprogramme de communication** » dans le menu « Connecter »

Programme n°6 :

Proposition de solution :



```
mBot - générer le code
répéter 2 fois
  activer le moteur M1 à la puissance 100
  activer le moteur M2 à la puissance 0
  attendre 3 secondes
  activer le moteur M1 à la puissance 0
  activer le moteur M2 à la puissance 100
  attendre 3 secondes
```

Pour les mêmes raisons que vues précédemment, il faudra arrêter le moteur M1 avant de démarrer M2 et inversement : un moteur s'arrête si on le lui demande explicitement sinon...il continue à tourner.

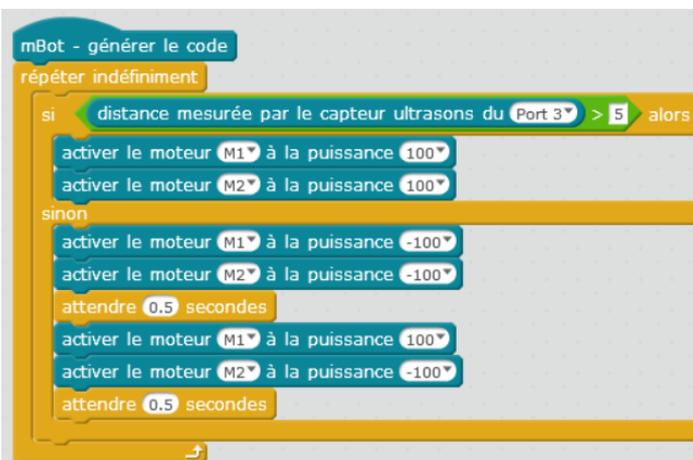
Programme n°7 :

Proposition de solution :



```
mBot - générer le code
répéter indéfiniment
  si distance mesurée par le capteur ultrasons du Port 3 > 5 alors
    avancer à la vitesse 100
  sinon
    reculer à la vitesse 100
    attendre 0.5 secondes
    tourner à droite à la vitesse 255
    attendre 0.5 secondes
```

Noter que la distance est exprimée en cm. On utilise ici la commande « tourner à droite » mais nous aurions pu faire différemment, c'est (presque...) pareil.



```
mBot - générer le code
répéter indéfiniment
  si distance mesurée par le capteur ultrasons du Port 3 > 5 alors
    activer le moteur M1 à la puissance 100
    activer le moteur M2 à la puissance 100
  sinon
    activer le moteur M1 à la puissance -100
    activer le moteur M2 à la puissance -100
    attendre 0.5 secondes
    activer le moteur M1 à la puissance 100
    activer le moteur M2 à la puissance -100
    attendre 0.5 secondes
```

La différence réside dans le fait qu'ici, on peut maîtriser mieux la rotation (dans cet exemple on inverse les vitesse sur les deux moteurs pour tourner, le robot va donc tourner sur place plutôt que de décrire un arc de cercle si nous avons 0 pour l'un et 100 pour l'autre).

Programme n°8 :

Proposition de solution : La vitesse des roues droite et gauche pour les rotations dépendra de votre circuit et du fait que les virages soient plus ou moins serrés. A régler selon les cas.

```
mBot - générer le code
mettre vitesse à 80
répéter indéfiniment
  si suiveur de ligne Port 2 Côté gauche est blanc alors
    activer le moteur M1 à la puissance vitesse
    activer le moteur M2 à la puissance -1 * vitesse
  si suiveur de ligne Port 2 Côté droit est blanc alors
    activer le moteur M1 à la puissance -1 * vitesse
    activer le moteur M2 à la puissance vitesse
  si suiveur de ligne Port 2 Côté droit est blanc et suiveur de ligne Port 2 Côté gauche est blanc alors
    activer le moteur M1 à la puissance 0
    activer le moteur M2 à la puissance 0
  si suiveur de ligne Port 2 Côté droit est noir et suiveur de ligne Port 2 Côté gauche est noir alors
    activer le moteur M1 à la puissance vitesse
    activer le moteur M2 à la puissance vitesse
```

Vous trouverez **d'autres propositions** d'applications sur mon site <http://www.fredtechnocollege.org>

Notes personnelles :