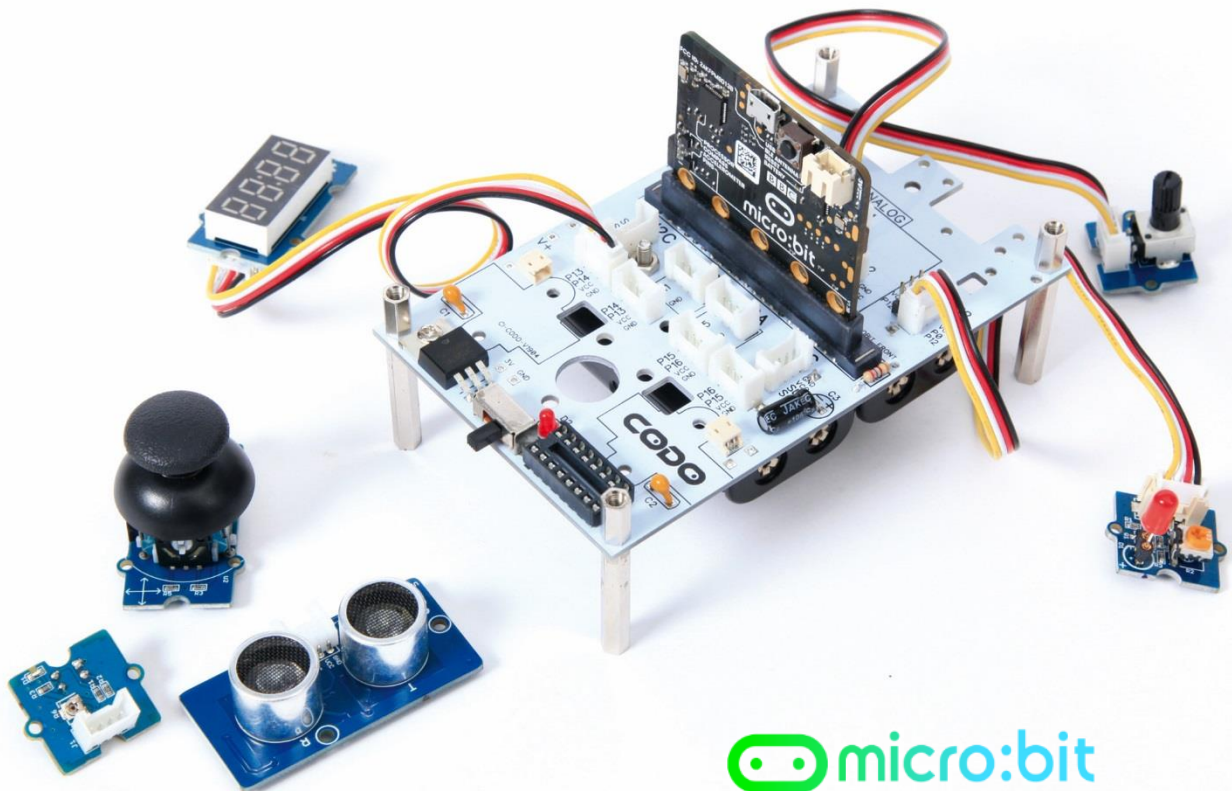
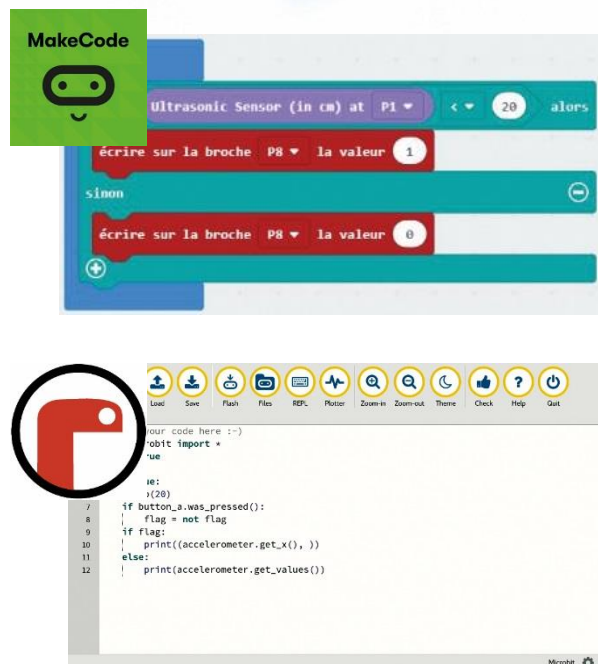


CODO

Grove micro:bit project board



 micro:bit



SOMMAIRE

Introduction	3
La carte CODO	3
Le Robot CODO.....	3
Les environnements de programmation.....	4
Les fiches d'exercices	5
Présentation de la carte CODO.....	6
Options.....	7
Motorisation	7
Porte-stylo.....	7
Plateforme robotique	7
Programmation	8
Activités de programmation portant sur la carte micro:bit seule	8
Exercices avec modules Grove	11
Communication radio et en Bluetooth	13
Options.....	15
Motorisation	15
Porte-stylo.....	16
Plateforme robotique	17
Détection de ligne	18
Joystick.....	19
Procédure de chargement d'un programme.....	22
Le logiciel MakeCode :	22
L'éditeur mu :	24
Programmation avancée	25
Blocs.....	25
Python	26

Introduction

Le projet CODO s'adresse aussi bien à des utilisateurs totalement débutants qu'à des utilisateurs avertis pour créer des scénarios de programmation élaborés allant jusqu'à l'interaction du robot avec un smartphone.

La carte CODO

La carte CODO a été réalisée pour connecter simultanément différents modules Grove à la carte micro:bit.

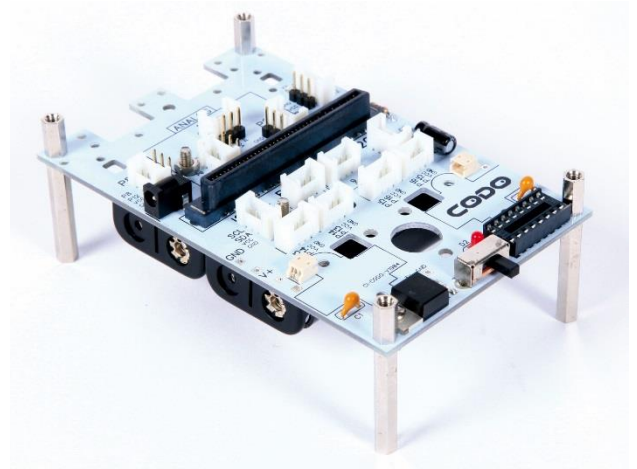
En effet, elle est équipée de 13 entrées/ sorties pour modules Grove.

Elle est conçue pour être alimentée en 6V grâce à 4 emplacements pour piles 1,5V.

La carte micro:bit peut être fixée sur la carte CODO via un connecteur prévu à cet effet.

Cet assemblage est idéal pour être codé en blocs et en python.

Attention toutefois, ce dossier n'explique pas l'utilisation de la carte micro:bit ni celle des modules Grove. Il est donc recommandé d'avoir des notions de bases sur ces éléments avant d'utiliser la carte CODO.



Le Robot CODO

La carte CODO peut accueillir différentes options telles que des moteurs pour fixer des roues et être transformée en robot CODO.

Une fois cette option installée, il est également possible d'ajouter l'option porte-stylo et la plateforme robotique avec pare-chocs.

Enfin, pour encore plus de plaisir à programmer, munissez-vous de deux cartes micro:bit et de deux cartes CODO (une étant le robot et l'autre la carte CODO de base) et modélisez une télécommande à l'aide d'un joystick par exemple.



Modules Grove traités dans les cartes d'activités ExoProg Grove /micro:bit

Bouton-poussoir (S-101020003), potentiomètre (S-101020017), détection de ligne (S-101020009), télémètre à ultrasons (S-101020010), LED rouge (S-104030005), afficheur 4 digits (S-104030003), ruban de LED (S-104020108), joystick (S-101020028), haut-parleur miniature (S-107020001) et capteur de mouvement (S-101020083).

Les exercices proposés sont transposables sur d'autres modules de la gamme Grove.



Les environnements de programmation

Tous les programmes correspondant aux activités menées autour de la carte CODO ont été réalisés sous MakeCode (blocs) et certains avec l'éditeur mu (python).

MakeCode - Environnement de programmation en ligne.



Programmation séquentielle ou événementielle en blocs.

Traduction automatique en JavaScript.

Jeu d'instructions de base en français.

Possibilité d'enrichir le jeu d'instructions.

Espace de simulation.



Mu Python Editor - Environnement (gratuit) de programmation en Python.



Offre des fonctionnalités plus avancées que l'environnement de base MicroPython (flashage direct de la carte micro:bit, fonction d'auto-complétion du code).

Disponible en ligne ou installation sur PC Windows ou Mac OSX.

Permet de programmer en langage Python standard et sur différents matériels.



App Inventor 2 - Environnement de développement pour concevoir des applications pour smartphone ou tablette Android, développé par le MIT pour l'éducation.

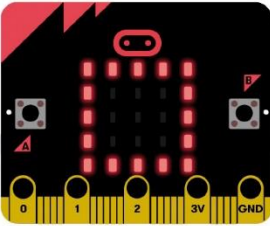
Pour les activités menées avec un smartphone ou une tablette, les programmes et applications ont été réalisés sous AppInventor2

Les cartes activités ExoProg

Pour chaque module ou option, nous vous proposons des exercices avec un fichier de correction qui propose un exemple de programme réalisé sous MakeCode en blocs.

Lorsque le logo python apparait à coté cela signifie qu'une correction en python sous mu est disponible.

MATRICE LED 1



Description : les LED émettent de la lumière.

Caractéristiques : la carte micro:bit possède 25 LED, toutes programmables individuellement, ce qui permet d'afficher du texte, des nombres et des images.

Instruction de base :

montrer l'icône

EXERCICES

NIVEAU 1

Afficher « Hello ! » lorsque le bouton A est pressé.

python™
codo-TEXTE-EX1.hex

NIVEAU 2

Activer la LED en bas à droite pendant 1 seconde lorsque le bouton A est pressé.

python™
codo-TEXTE-EX2.hex

NIVEAU 3

Faire clignoter la LED centrale 8 fois lorsque le bouton B est pressé.

python™
codo-TEXTE-EX3.hex

Deux approches :

1. Avec les corrections de programmes, les utilisateurs découvrent les principes de la programmation en blocs : chargement d'un programme, modification d'un programme et vérification sur le matériel (ex : modification des temps d'attente, etc.).
2. Les utilisateurs conçoivent eux-mêmes le programme pour atteindre l'objectif proposé, ils peuvent ensuite le comparer au fichier de correction.

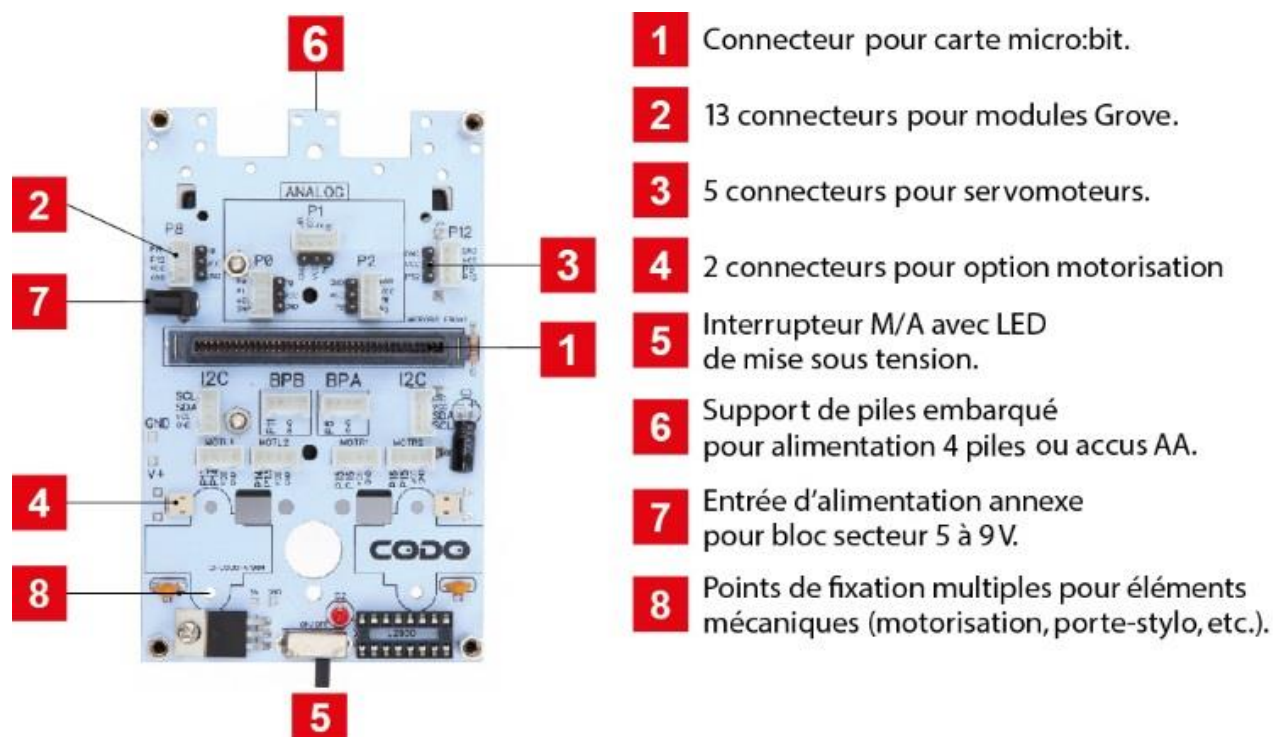
Jeu de cartes activités ExoProg Grove / micro:bit avec MakeCode / Mu Python
(réf. S-EXOPROG-MICROBIT)

Disponibles gratuitement



Présentation de la carte CODO

Dimensions : 72 x 113 mm.



Sur les 13 connecteurs Grove, deux correspondent à une connectique I2C, deux sont reliés aux boutons A et B de la carte micro:bit et trois sont analogiques.

La carte contient 22 perçages pour fixer divers éléments mécaniques (pièces réalisées avec imprimante 3D, découpe laser, etc.)

Embase Ø 1,3 mm pour bloc d'alimentation externe (24 V max).

Livrée avec 4 entretoises FF M3 hauteur 30 mm et 4 entretoises MF M3 hauteur 10 mm.

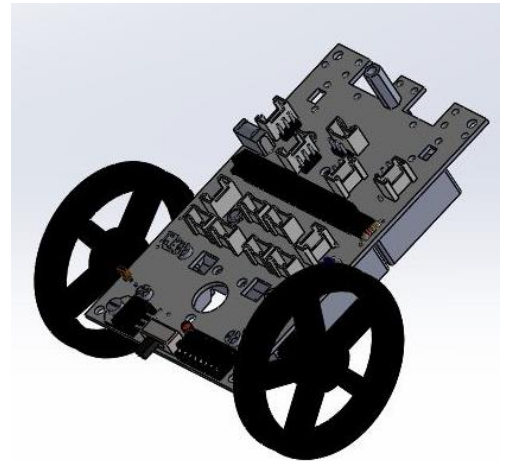
Options

Motorisation

Cette option permet de mettre la carte CODO en mouvement grâce à des moteurs atteignant une vitesse de 200 tours par minute.

Elle est constituée de 2 motoréducteurs 200 tr/min, 100 g.cm, 2 roues Ø 60 mm, 2 supports pour moteurs et 1 boîtier 16 pattes L293 (à fixer sur son support).

Fixation : 4 vis, 4 écrous, 1 entretoise MF Ø 3 x h. 10mm, 1 entretoise MF Ø 3 x h 20 mm, 1 écrou borgne Ø3 mm et 1 entretoise nylon Ø3 x h 4 mm.

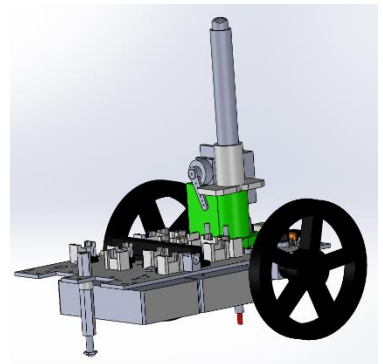


Porte-stylo

Cette option découle de l'option avec moteur. Avec les roues, le robot peut être en mouvement. Le stylo, piloté par un servomoteur, peut écrire et dessiner.

Elle est composée de 1 servomoteur, 2 pièces imprimées en 3D et 1 stylo.

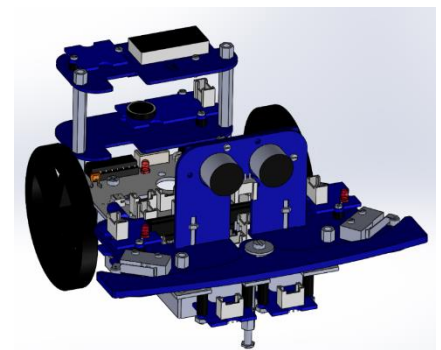
Fixation : 3 vis Ø3 x h10 mm, 3 écrous Ø3 mm et 1 vis Ø2 x h 12mm.



Plateforme robotique

Elle est composée de 6 plaques de PMMA épaisseur 3 mm, 2 microinterrupteurs **et 1 sélection de modules Grove.**

Fixation : 13 vis Ø 2 x h 12mm, 8 vis Ø 2 x h 16mm, 9 vis Ø 2 x h 8 mm, 1 vis Ø 3 x h 10 mm, 30 écrous Ø 2 mm, 1 écrou Ø 3 mm, 1 rondelle Ø 3mm et Ø 12mm, 13 entretoises nylon Ø2.2 x h 3 mm, 4 entretoises nylon Ø 2,2 x h 10 mm, 1 entretoise nylon Ø3,1 x h 4mm, 2 entretoises MF Ø3 x h30mm et 6 entretoises MF Ø3 x h 5 mm.

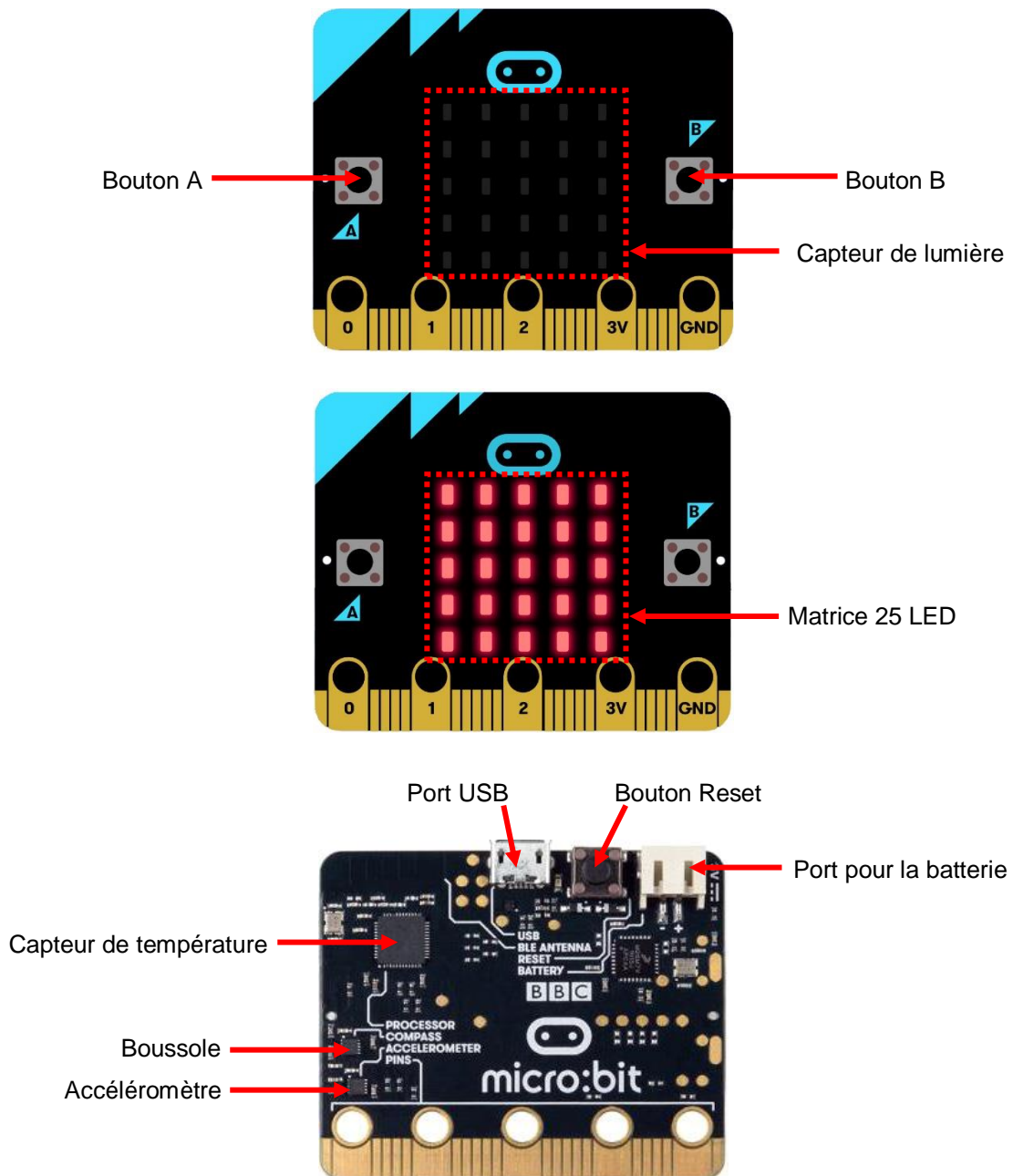



Programmation

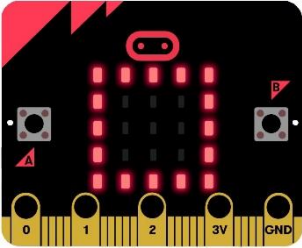



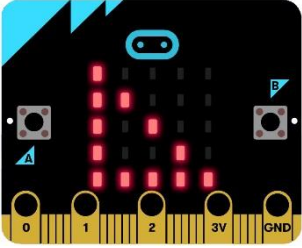



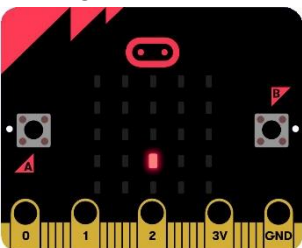







La carte micro:bit est une mini carte programmable conçue pour l'apprentissage de la programmation. Il est possible de réaliser une multitude de programme seulement avec cette petite carte.



Activités de programmation portant sur la carte micro:bit seule

Les activités de programmation proposées ci-après s'appuient sur les fonctionnalités offertes par la carte micro:bit.



Les exercices ayant le logo  ont également une correction proposée en python.



Modules	Enoncés	Corrections
MATRICE LED 1 	Niveau 1 : afficher un carré.	codo-MOTIFS-EX1.hex 
	Niveau 2 : afficher un carré et une seconde après un triangle.	codo-MOTIFS-EX2.hex 
	Niveau 3 : répéter l'affichage d'un carré, d'un triangle, puis d'une croix avec un intervalle de 1 seconde.	codo-MOTIFS-EX3.hex 
BOUTON A ET B 	Niveau 1 : afficher un triangle quand le bouton A est pressé.	codo-BOUTONS-EX1.hex 
	Niveau 2 : afficher un triangle quand le bouton A est pressé et effacer l'écran après 3 secondes.	codo-BOUTONS-EX2.hex 
	Niveau 3 : afficher une croix lorsque le bouton A est pressé, le nombre 15 lorsque le bouton B est pressé et la première lettre de votre prénom lorsque les boutons A et B sont pressés.	codo-BOUTONS-EX3.hex 
MATRICE LED 2 	Niveau 1 : afficher « Hello ! » lorsque le bouton A est pressé.	codo-TEXTE-EX1.hex 
	Niveau 2 : activer la LED en bas à droite pendant 1 seconde lorsque le bouton A est pressé.	codo-MATRICE-EX1.hex 
	Niveau 3 : faire clignoter la LED centrale 8 fois lorsque le bouton B est pressé.	codo-MATRICE-EX2.hex 
CAPTEUR DE TEMPERATURE 	Niveau 1 : afficher la valeur de la température.	codo-TEMP-EX1.hex 
	Niveau 2 : afficher la valeur de la température et allumer la LED en haut à gauche si la température dépasse 28°C.	codo-TEMP-EX2.hex 
	Niveau 3 : faire clignoter la LED en haut à gauche de plus en plus rapidement en fonction de la valeur de la température.	codo-TEMP-EX3.hex 

CAPTEUR DE LUMIERE 	Niveau 1 : tracer le graphe du niveau d'intensité lumineuse.	codo-LUM-EX1.hex
	Niveau 2 : afficher la valeur du niveau d'intensité lumineuse.	codo-LUM-EX2.hex python™
	Niveau 3 : faire clignoter une LED en fonction de l'intensité mesurée par le capteur de lumière.	codo-LUM-EX3.hex python™
ACCELEROMETRE 	Niveau 1 : lorsque l'on secoue la carte, afficher l'icône "fâché".	codo-accel-EX1.hex python™
	Niveau 2 : lorsque l'on incline la carte à droite, afficher une flèche à droite et afficher une flèche à gauche lorsque l'on incline la carte à gauche.	codo-accel-EX2.hex python™
	Niveau 3 : lorsqu'un petit choc est détecté, allumer toutes les LED et les éteindre 1 seconde après.	codo-accel-EX3.hex python™
BOUSSOLE 	Niveau 1 : afficher la direction de la boussole, la faire pivoter et observer les variations.	codo-boussole-EX1.hex python™
	Niveau 2 : si la boussole est pointée vers le nord, afficher N et si la boussole est pointée vers le sud, afficher S.	codo-boussole-EX2.hex python™
	Niveau 3 : si la boussole est pointée vers le nord, montrer la flèche nord et si la boussole est pointée vers le sud, montrer la flèche sud.	codo-boussole-EX3.hex python™

Exercices avec modules Grove

Des modules Grove peuvent être connectés avec un câble Grove à la carte micro:bit grâce à la carte CODO présentée précédemment.

Voici, ci-dessous, quelques exemples d'exercices :

Modules	Enoncés	Corrections
LED 	Niveau 1 : activer la LED pendant 5 secondes, puis l'éteindre.	codo-led-EX1.hex  python™
	Niveau 2 : faire clignoter la LED toutes les 0,5 secondes	codo-led-EX2.hex  python™
	Niveau 3 : faire varier la vitesse de clignotement de la LED.	codo-led-EX3.hex  python™
BOUTON-POUSOIR 	Niveau 1 : lorsque l'on appuie sur le bouton poussoir, montrer une icône au choix, sinon ne rien montrer.	codo-BP-EX1.hex  python™
	Niveau 2 : lorsque le bouton poussoir est pressé, allumer la LED et l'éteindre lorsqu'il est relâché.	codo-BP-EX2.hex  python™
	Niveau 3 : éteindre la LED 3 secondes après avoir relâché le bouton.	codo-BP-EX3.hex  python™
AFFICHEUR 4 DIGITS 	Niveau 1 : afficher "1234".	codo-digits-EX1.hex
	Niveau 2 : afficher la valeur du capteur de température.	codo-digits-EX2.hex
	Niveau 3 : afficher tous les caractères possibles en les faisant défiler chacun leur tour.	codo-digits-EX3.hex
ULTRASONS 	Niveau 1 : activer la LED si la distance mesurée est inférieure à 20 cm sinon la désactiver.	codo-ULTRA-EX1.hex
	Niveau 2 : activer la LED si la distance mesurée est inférieure à 20 cm et la réactiver quand l'objet détecté s'est éloigné de plus de 25 cm.	codo-ULTRA-EX2.hex
	Niveau 3 : faire clignoter la LED à une vitesse variant avec la distance mesurée.	codo-ULTRA-EX3.hex
SPEAKER 	Niveau 1 : actionner le buzzer.	codo-speaker-EX1.hex  python™
	Niveau 2 : actionner le buzzer toutes les secondes.	codo-speaker-EX2.hex  python™
	Niveau 3 : jouer une mélodie une fois en accéléré et quand elle est finie allumer la LED en haut à gauche.	codo-speaker-EX3.hex  python™

POTENTIOMETRE 	Niveau 1 : faire clignoter une LED à une fréquence dépendant de la valeur du potentiomètre.	codo-POT-EX1.hex 
	Niveau 2 : tracer le graphe de la valeur du potentiomètre.	codo-POT-EX2.hex
	Niveau 3 : ramener la valeur du potentiomètre à une échelle de 0 à 100 et l'afficher sur l'afficheur 4 digits.	codo-POT-EX3.hex
CAPTEUR DE MOUVEMENT 	Niveau 1 : lorsque l'on tourne dans le sens des aiguilles d'une montre afficher ☺ et dans le sens antihoraire afficher ☹.	codo-CM-EX1.hex
	Niveau 2 : afficher le nombre de fois qu'un mouvement à droite a été effectué.	codo-CM-EX2.hex
	Niveau 3 : lors d'un mouvement dans le sens horaire, allumer une LED choisie au hasard. S'il s'agit de la LED centrale, attendre 1 seconde et buzzer le son "High F".	codo-CM-EX3.hex
NEOPIXEL 	Niveau 1 : allumer les 30 LED en arc en ciel.	codo-Neo-EX1.hex
	Niveau 2 : afficher les LED en vert lorsque le bouton A est pressé.	codo-Neo-EX2.hex 
	Niveau 3 : allumer 15 LED dans un mélange de rouge, vert et bleu.	codo-Neo-EX3.hex 
NEOPIXEL (suite) 	Niveau 1 : faire une rotation des LED toutes les 0.5 secondes.	codo-Neo2-EX1.hex
	Niveau 2 : allumer le nombre de LED égal au nombre de lettres du mot "Bonjour" en orange.	codo-Neo2-EX2.hex
	Niveau 3 : faire apparaître un arc en ciel des LED et faire varier les couleurs à l'aide du potentiomètre.	codo-Neo2-EX3.hex

Communication radio et en Bluetooth

La communication radio

L'outil **Radio** sur le logiciel MakeCode permet la communication entre cartes micro:bit.

Il est nécessaire de créer deux programmes, un pour chaque carte micro:bit. Elles doivent être programmées selon le même « groupe radio » pour que les programmes fonctionnent.

Voici, ci-dessous, quelques exemples d'exercices :

RADIO	Niveau 1 : Communication de la carte micro:bit 1 vers la carte micro:bit 2. Lorsque l'on appuie sur le bouton A sur la carte 1, afficher 1 sur les deux cartes.	codo-RADIO-EX1A.hex codo-RADIO-EX1B.hex
	Niveau 2 : Communication bidirectionnelle. lorsque l'on appuie sur le bouton A de la carte 1 le chiffre 3 apparait sur la carte 2, lorsque l'on appuie sur le bouton A sur la carte 2 le chiffre 3 disparaît et apparait sur la carte 1.	codo-RADIO-EX2A.hex codo-RADIO-EX2B.hex
	Niveau 3 : Si le bouton A est pressé, envoyer 1 par radio, si le bouton B est pressé envoyer 2 par radio. Si 1 est reçu allumer 30 LED du Neopixel en rouge, si 2 est reçu allumer 30 LED du Neopixel en vert.	codo-RADIO-EX3A.hex codo-RADIO-EX3B.hex

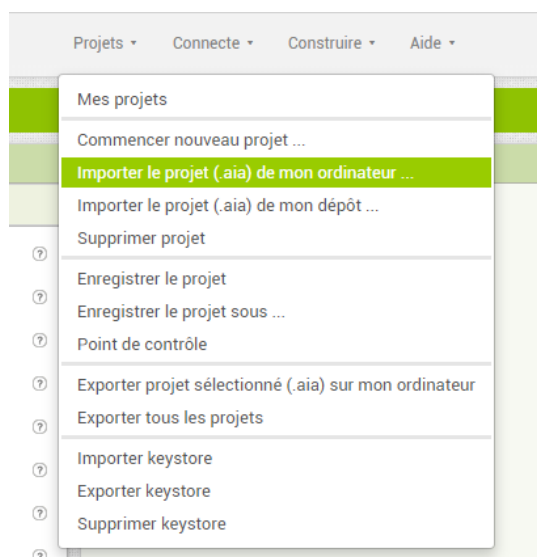
Le Bluetooth

Le mode de communication en Bluetooth permet transmettre des données au travers d'une liaison sans fil entre un smartphone Android et la carte micro:bit.

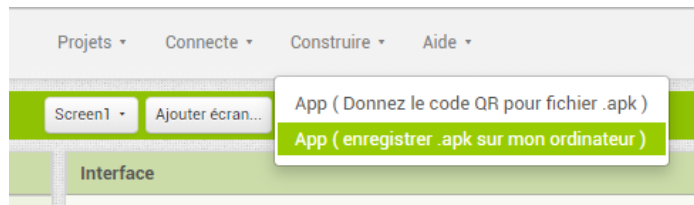
Les applications proposées sont réalisées sous AppInventor 2.

Pour utiliser les programmes / applications de l'option Bluetooth, il faut :

- 1) Activer le Bluetooth et appairer votre smartphone ou tablette à la carte micro:bit.
- 2) Se connecter sur Appinventor2 et charger le fichier « Documentdbase.aia ».



3) Compléter l'application et l'enregistrer en fichier .apk .



- 4) Installer l'application correspondante sur votre smartphone.
- 5) Charger le programme « codo-bluetoothbase.hex » sur le logiciel MakeCode.
- 6) Charger l'extension <https://microbit-blockytalky.glitch.me/> sur MakeCode.
- 7) Compléter le programme suivant l'exercice demandé.
- 8) Télécharger le programme et le copier sur la carte micro bit.

Note : pour pouvoir installer les applications d'A4, il faut autoriser l'installation d'applications de sources inconnues dans les paramètres de sécurité de votre smartphone.

Important : si vous rencontrez des problèmes de connexion, il peut s'avérer nécessaire de relancer l'appairage, notamment si vous utilisez plusieurs robots ou plusieurs smartphones.

Pour créer / modifier une application avec AppInventor :

- avoir un compte Gmail pour la création d'une session.
- Le fichier de modèle « Documentdebase.aia » proposé par A4 pour la communication Bluetooth sert de base pour coder une application sur AppInventor. Le projet de base contient toute la partie connexion/déconnexion de la carte CODO par Bluetooth.

Exercices

Pour chaque exercice, nous proposons des fichiers de correction :

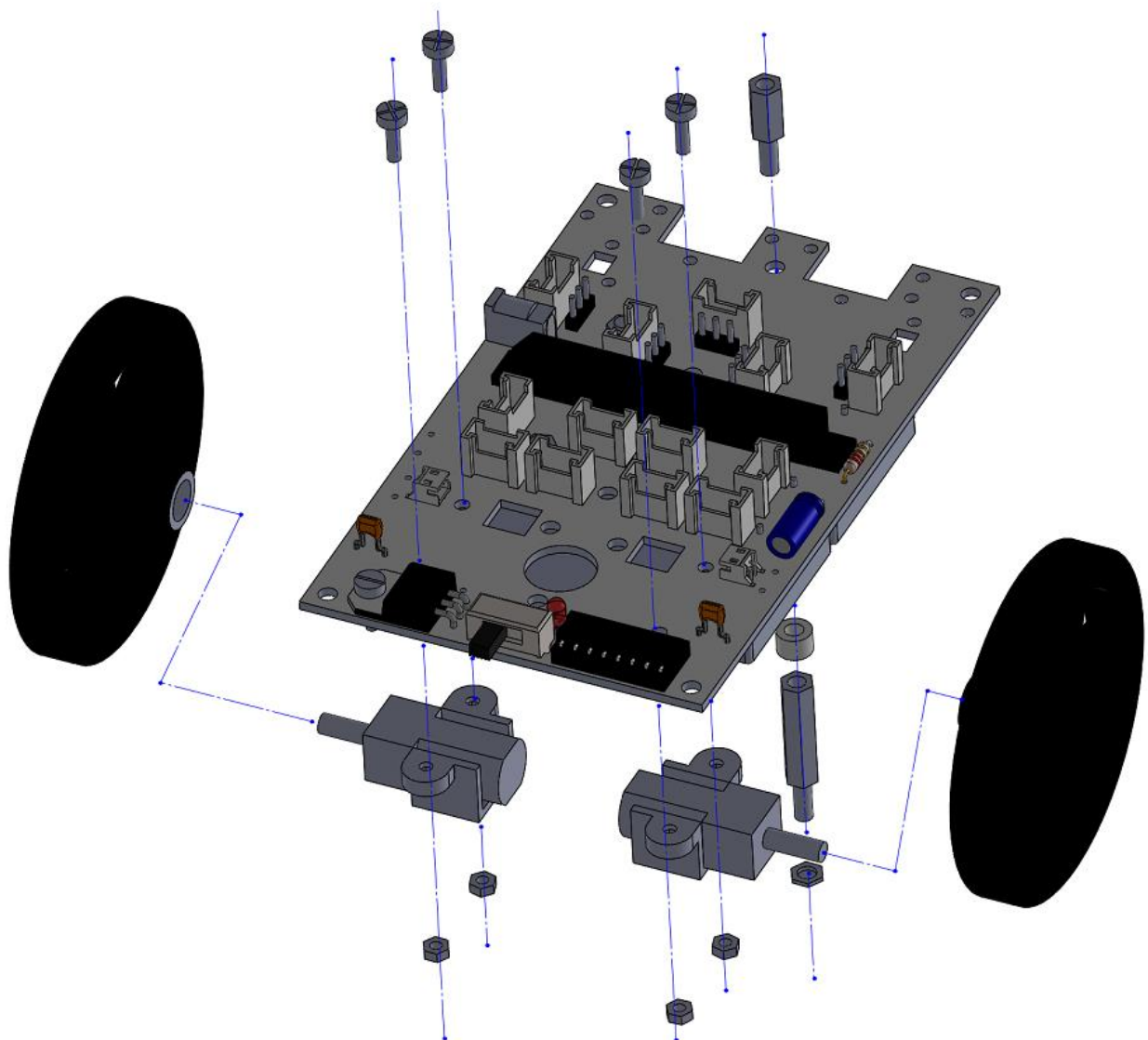
- un fichier **.hex** à charger dans le robot.
- un fichier **.apk** à installer sur le smartphone.
- le fichier **.aia** modifiable avec AppInventor.

Voici quelques exemples d'exercices ci-dessous :

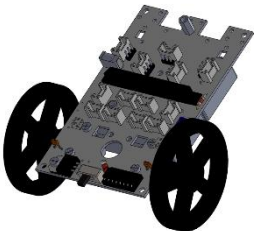



BLUETOOTH	Niveau 1 : Communication de la carte micro:bit vers l'application. Lorsque l'on appuie sur le bouton A afficher "1" sur la carte et sur l'application.	codo-BLUETOOTH-EX1.hex
	Niveau 2 : Communication de l'application vers la carte micro:bit. lorsque l'on appuie sur un bouton dans l'application, un chiffre est envoyé et affiché sur la carte.	codo-BLUETOOTH-EX2.hex
	Niveau 3 Communication bidirectionnelle. lorsque l'on appuie sur le bouton A le chiffre 3 apparait sur l'application et lorsque l'on appuie sur un bouton dans l'application le chiffre 3 disparaît et s'affiche sur la carte.	codo-BLUETOOTH-EX3.hex

Options

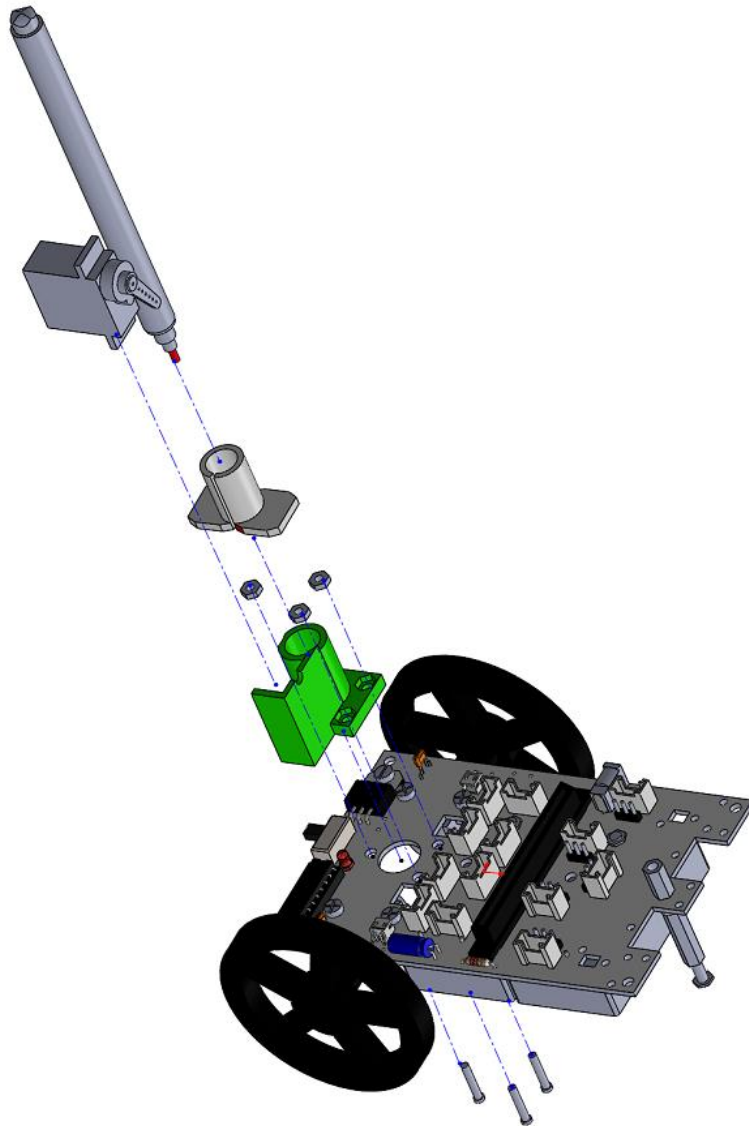
Motorisation



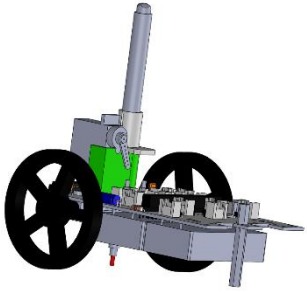



Voici, ci-dessous, quelques exemples d'exercices :

ROBOT CODO Option motorisation 	Niveau 1 : faire avancer le robot pendant 1 seconde à sa vitesse moyenne.	codo-ROBOT-EX1.hex 
	Niveau 2 : faire reculer le robot pendant 1 seconde à sa vitesse moyenne.	codo-ROBOT-EX2.hex 
	Niveau 3 : faire avancer le robot pendant 0.5 seconde puis le faire tourner à droite.	codo-ROBOT-EX3.hex 

Porte-stylo



Voici, ci-dessous, quelques exemples d'exercices :

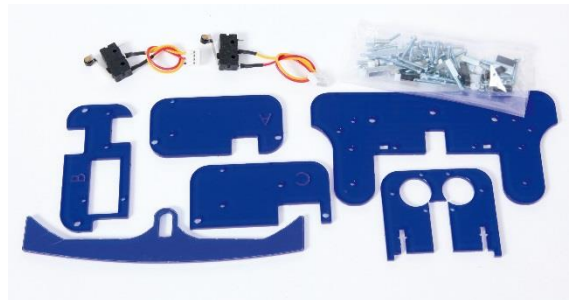
ROBOT CODO OPTION PORTE-STYLO 	Niveau 1 : dessiner un segment.	codo-STYLO-EX1.hex  python™
	Niveau 2 : dessiner des traits en pointillés.	codo-STYLO-EX2.hex  python™
	Niveau 3 : dessiner un rond.	codo-STYLO-EX3.hex  python™

Plateforme robotique

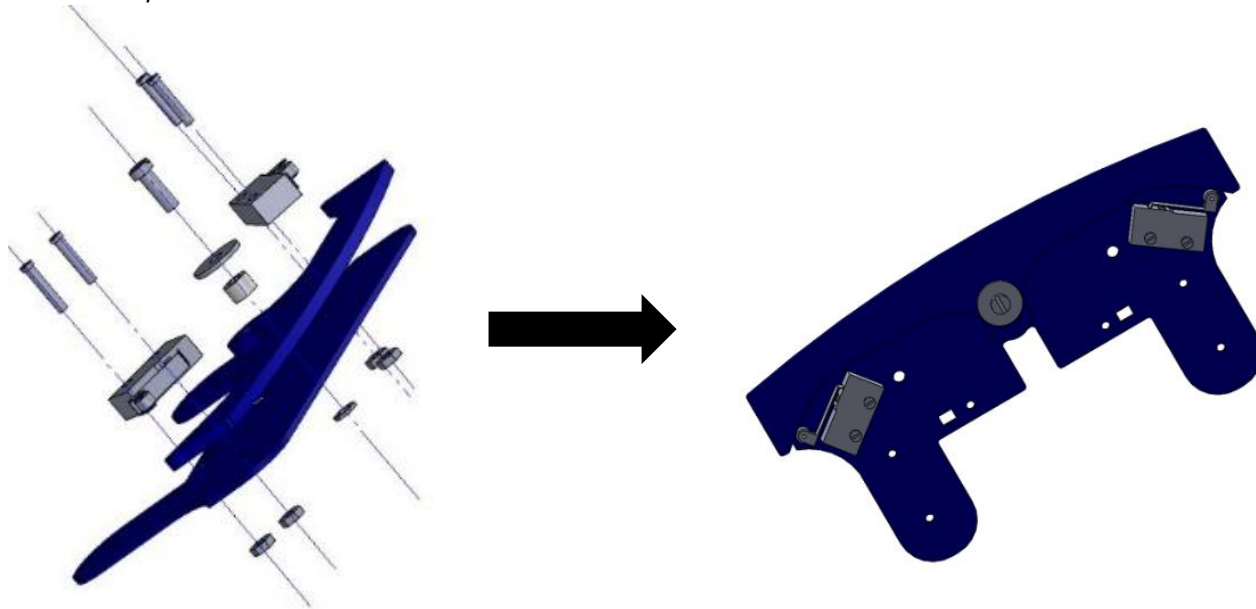
Pare-chocs

Le pare-chocs est accompagné de deux microrupteurs, de 4 vis de 12mm, de 4 écrous, d'un écrou, d'une vis de 10mm, d'une entretoise le diamètre 3mm et d'une rondelle.


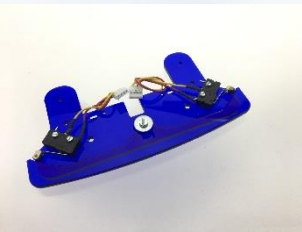


Les microrupteurs se connectent sur les connecteurs BPA et BPB.



Fixation du pare-chocs



Voici, ci-dessous, quelques exemples d'exercices :

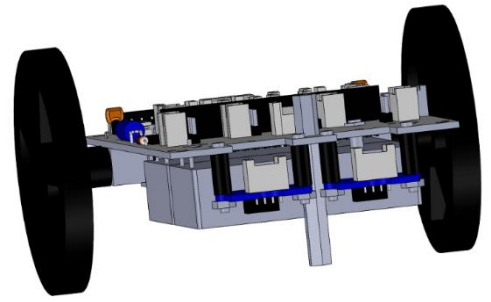
ROBOT CODO OPTION PARE-CHOCs  	Niveau 1 : afficher une flèche à l'opposé du bouton A lorsqu'il est pressé et afficher une flèche à l'opposé du bouton B lorsqu'il est pressé.	codo-PARECHOC-EX1.hex 
	Niveau 2 : faire une manœuvre d'évitement lorsqu'un obstacle est détecté à gauche ou à droite.	codo-PARECHOC-EX2.hex 
	Niveau 3 : quand le robot rencontre un obstacle à moins de 10 cm à l'aide du module ultrasons, la LED s'allume, le speaker sonne et le robot recule et tourne pour éviter l'obstacle.	codo-PARECHOC-EX3.hex

Détection de ligne

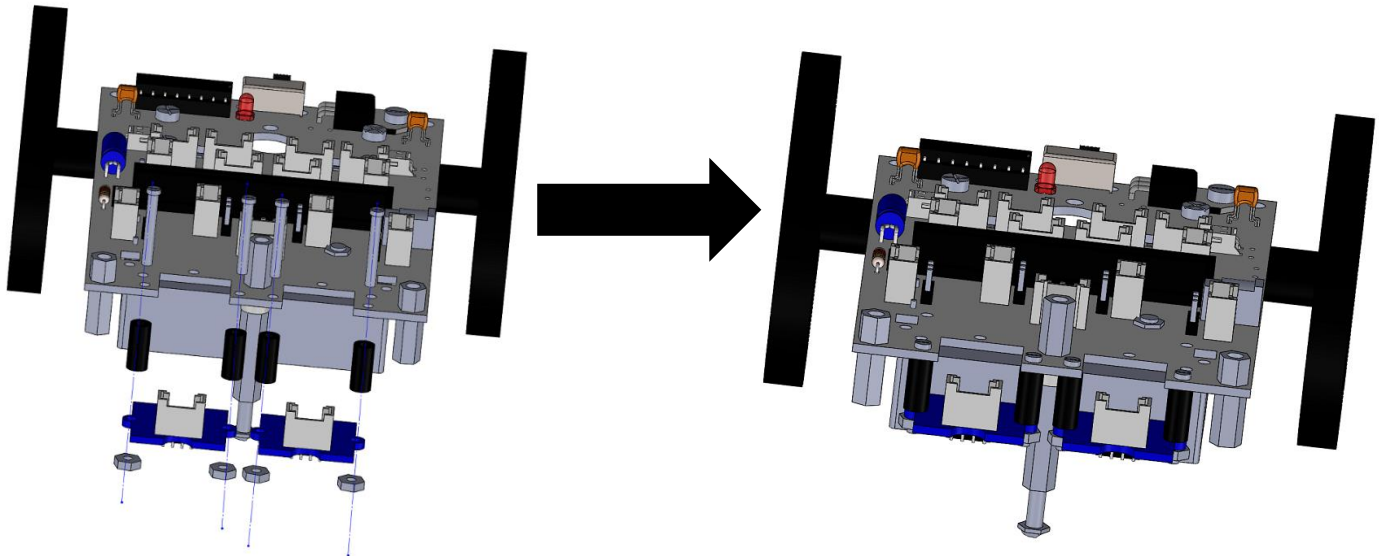
Module Grove de détection de ligne fonctionne lorsqu'il est placé entre 4mm et 15mm de la surface à analyser.

Le capteur infrarouge renvoie 1 si du noir est détecté et 0 si du blanc est détecté. Il est à utiliser sur des bandes noires de 3 cm sur fond blanc.

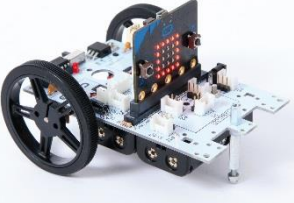



Les deux modules sont accompagnés de 4 entretoises en nylon de 10mm, de 4 vis de 16mm et de 4 écrous.



Fixation de la détection de ligne



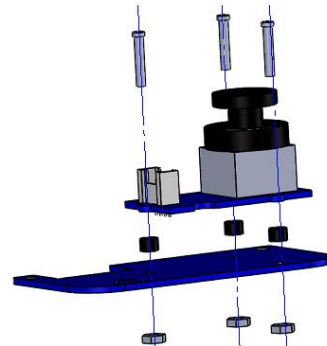
Voici, ci-dessous, quelques exemples d'exercices :

ROBOT CODO DETECTION DE LIGNE  	Niveau 1 : allumer la LED centrale si le capteur détecte du noir et l'éteindre sinon.	codo-LIGNE-EX1.hex 
	Niveau 2 : mettre le robot dans un circuit fermé noir. Lorsque le capteur détecte du noir il change de direction.	codo-LIGNE-EX2.hex 
	Niveau 3 : dessiner une large bande noire à la main et faire suivre le chemin au robot.	codo-LIGNE-EX3.hex

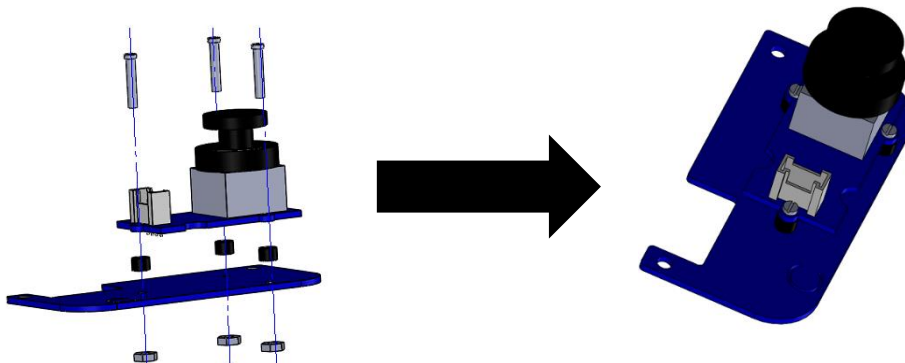
Joystick

Le joystick peut être fixé et connecté sur une deuxième carte CODO, elle-même connectée à une deuxième carte micro:bit. Les deux microcontrôleurs peuvent communiquer en radio et le joystick peut servir par exemple de télécommande.

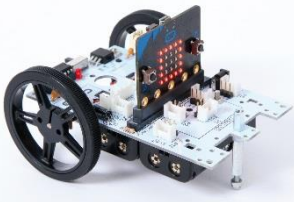


Le module se fixe sur la carte C et est accompagné de 2 vis de 12mm, de 3 entretoises en nylon de diamètre 3mm et de 3 écrous.

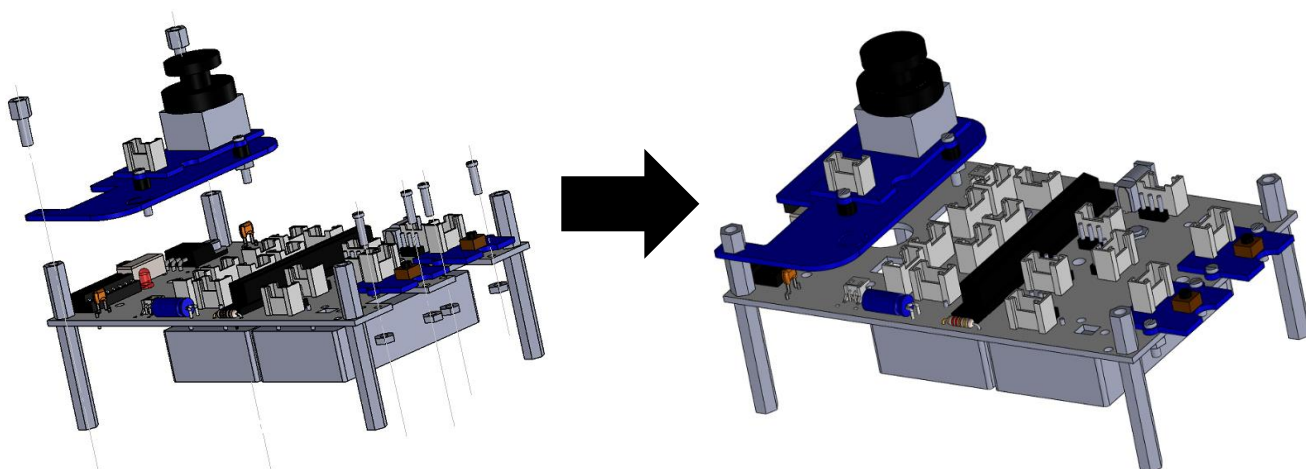


Fixation du joystick



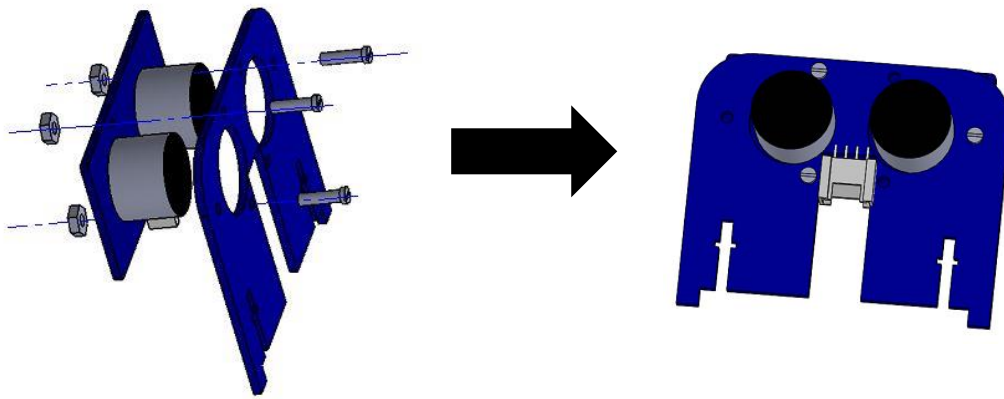
Voici, ci-dessous, quelques exemples d'exercices :

ROBOT CODO JOYSTICK  	Niveau 1 : afficher les angles de rotation sur l'axe X lorsque le bouton A est pressé et afficher les angles de rotation sur l'axe Y lorsque le bouton B est pressé.	codo-JOYSTICK-EX1.hex 
	Niveau 2 : par radio faire avancer ou reculer le robot (carte 1) à l'aide du joystick (carte 2).	codo-JOYSTICK-EX2A.hex codo-JOYSTICK-EX2B.hex
	Niveau 3 : diriger le robot dans toutes les directions par radio avec le joystick connecté à la carte 2.	codo-JOYSTICK-EX3A.hex codo-JOYSTICK-EX3B.hex



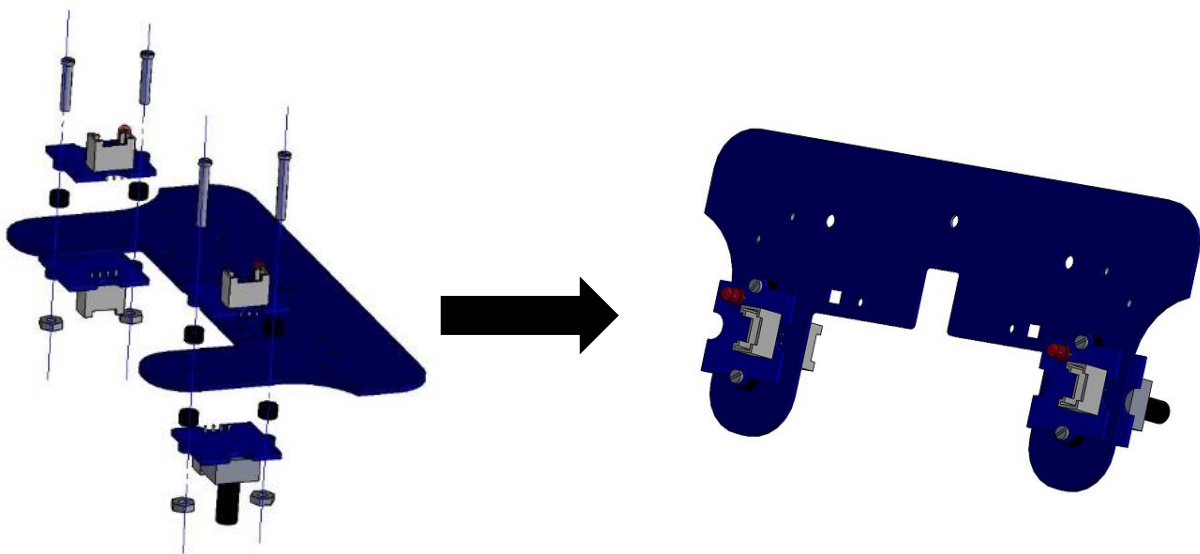
La carte C se fixe sur la carte CODO de base avec deux entretoises MF de 5mm. Pour concevoir une télécommande un peu plus complète deux boutons poussoir Grove peuvent être fixés à l'avant avec 4 vis de 12mm et 4 écrous de diamètre 2mm.

Fixation du télémètre à ultrasons



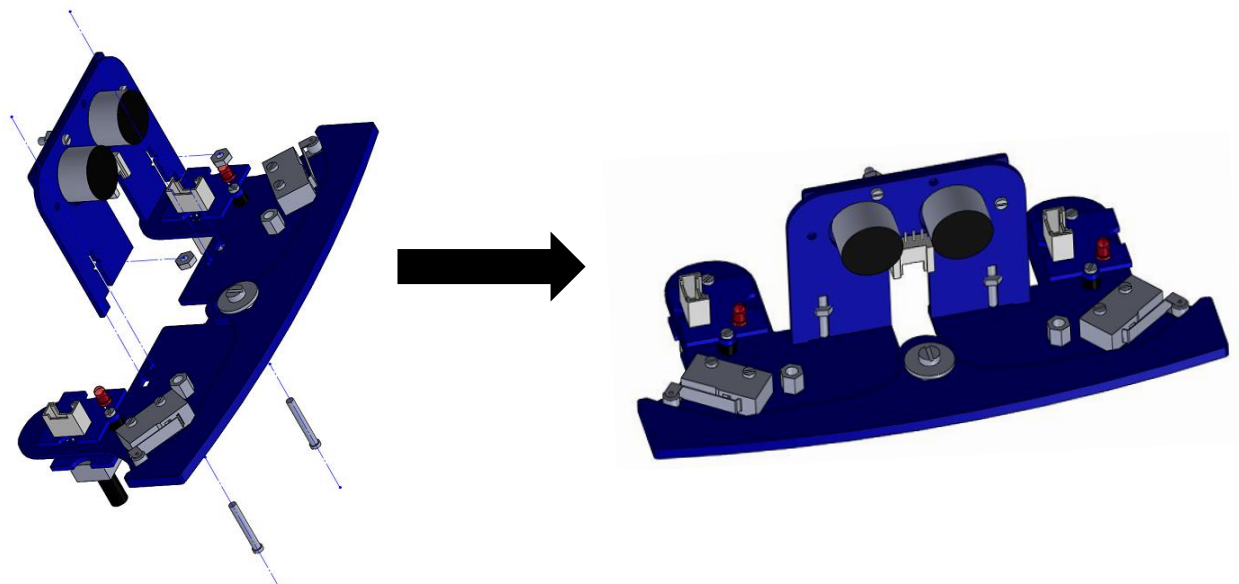
3 vis de 12mm, 3 écrous, un télémètre à ultrasons et la plaque de PMMA correspondante.

Fixation des deux LED, du potentiomètre et du bouton-poussoir



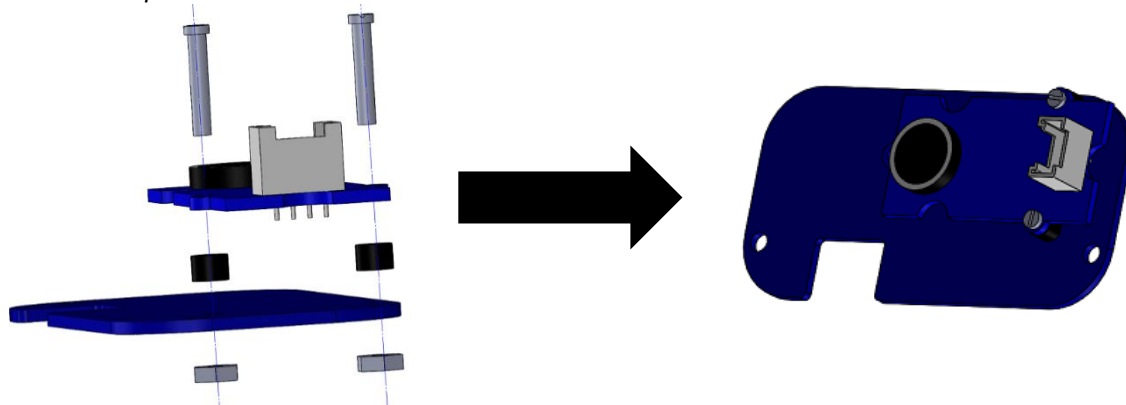
2 vis de 12mm, 2 vis de 16 mm, 2 vis de 12mm, 4 écrous, 6 entretoises nylon, 2 LED, 1 bouton poussoir, un potentiomètre et la plaque du PMMA correspondante.

Fixation des deux plaques précédentes



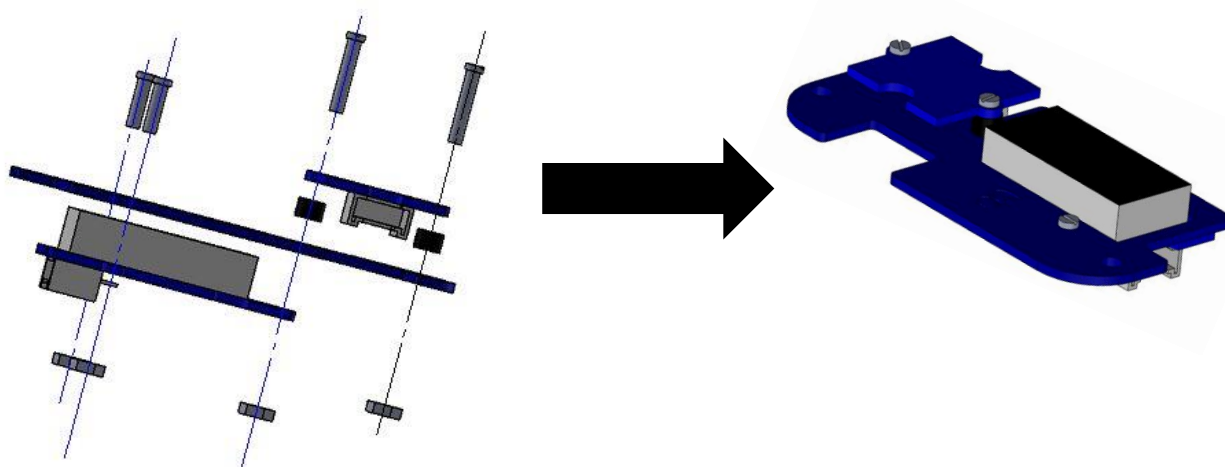
2 vis de hauteur 16 mm, 2 écrous de diamètre 2 mm.

Fixation du speaker



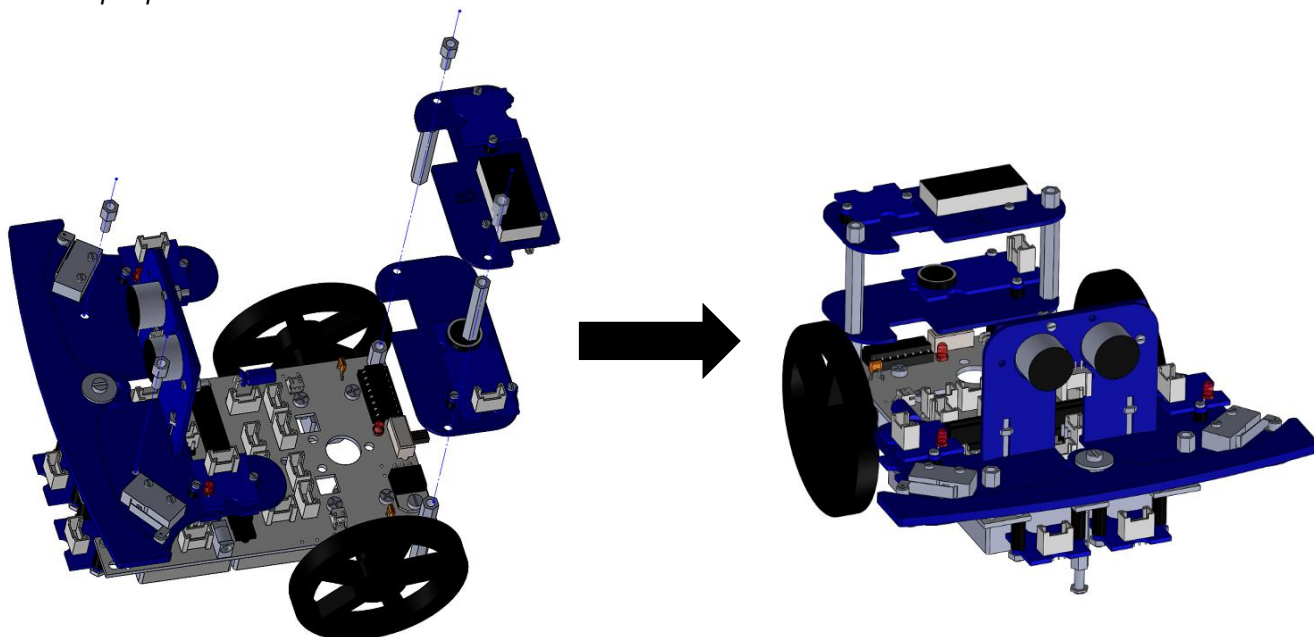
2 vis de 12 mm, 2 écrous, 2 entretoises nylon, 1 speaker et la carte A.

Fixation de l'afficheur 4-digits et du capteur de mouvement



2 vis de 12 mm, 2 vis de 8 mm, 4 écrous, 2 entretoises nylon, 1 afficheur 4 digits, 1 capteur de mouvement et la carte B.

Fixation des plaques sur la carte CODO



2 entretoises MF de 30 mm, 4 entretoises MF de 5mm.

Procédure de chargement d'un programme

La carte CODO est adaptée pour programmer sur le logiciel Makecode pour coder en blocs ainsi que sur l'éditeur mu pour coder en python. Le site ApplInventor2 peut être utilisé pour la communication Bluetooth.

Le logiciel MakeCode :

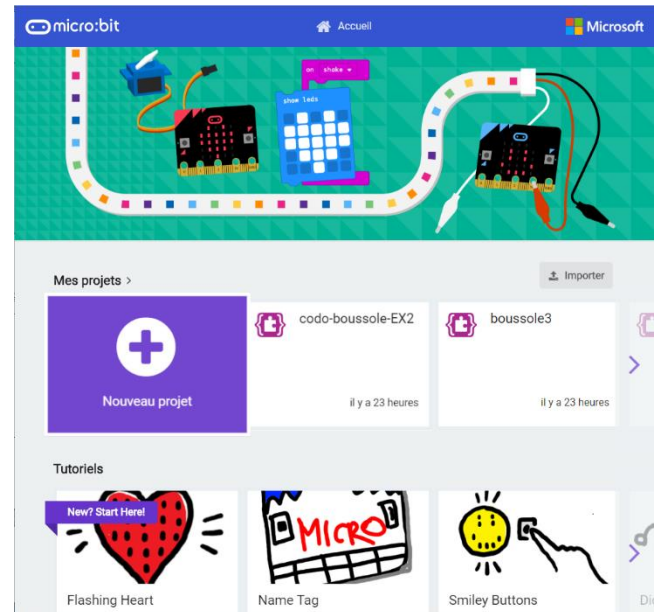
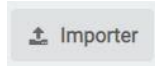
Pour accéder à ce logiciel disponible gratuitement sur internet, copier le lien suivant :

<https://makecode.microbit.org/>

Cliquez sur **Nouveau projet** pour commencer à coder.

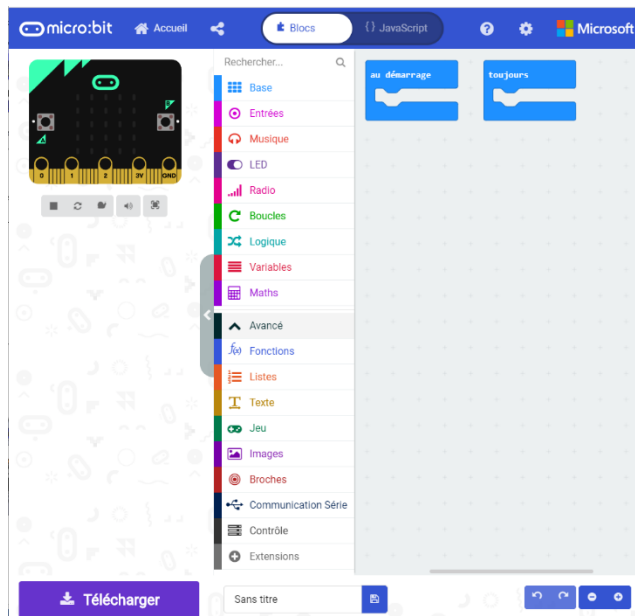


Vous pouvez également importer un fichier depuis votre ordinateur avec **Importer**.



Relier ensuite la carte micro:bit à l'ordinateur avec le câble de programmation USB.

Dans le logiciel MakeCode, créer votre programme et le télécharger. Le copier ensuite sur la carte micro:bit. Votre programme est lancé.



Vous pouvez coder en blocs et en JavaScript, ici nous nous concentrons sur le mode blocs.

Sur la gauche de l'écran, vous trouvez les différents blocs disponibles.

En haut, les blocs de bases sont affichés.

En cliquant sur **Avancé** vous trouvez des blocs complémentaires.

Vous pouvez également ajouter des blocs supplémentaires avec **Extensions**.

Entrez le lien suivant : <https://github.com/Seeed-Studio/pxt-grove.git> pour avoir des blocs correspondant aux modules Grove tels que l'afficheur 4 digits ou le télémètre à ultrasons.

Entrez le lien suivant : <https://github.com/Microsoft/pxt-neopixel> pour obtenir les blocs correspondant au Neopixel.

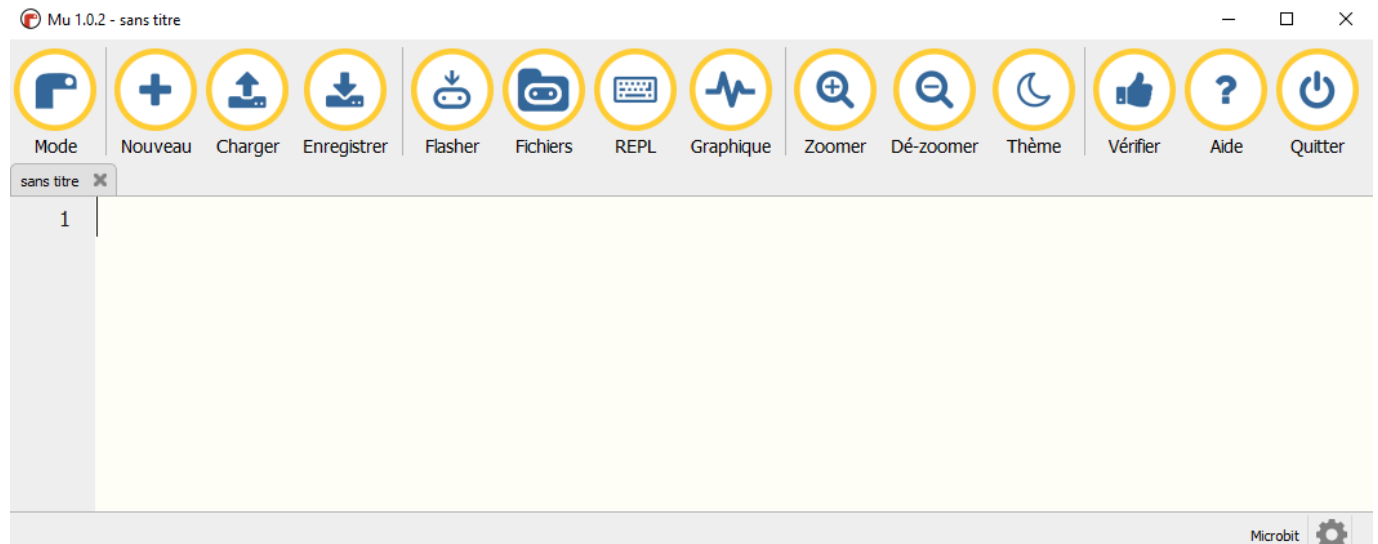
Entrez le lien suivant : <https://github.com/CODOMicrobit/pxt-CODO.git> pour obtenir les blocs correspondant aux moteurs du robot.

La carte micro:bit en haut à gauche de l'écran permet d'avoir une simulation du programme en cours. Lorsque vous cliquez droit sur un bloc et aide, vous obtiendrez une explication sur le bloc correspondant.

L'éditeur mu :

Le logiciel mu permet de coder en python standard et en python pour la carte micro:bit c'est-à-dire en python standard. Le logiciel est à télécharger avec le lien suivant par exemple : <https://codewith.mu/>

Pour lancer un programme, écrivez-le et cliquez sur « flasher » en haut de l'écran. Vous pouvez également importer un fichier existant sur votre ordinateur en cliquant sur « charger ».



Guide les librairies :

Vous trouverez de nombreuses librairies sur le site suivant :

<https://microbit-micropython.readthedocs.io/fr/latest/tutorials/introduction.html>

Blocs

Thèmes	Enoncés	Corrections
Matrice LED	Exercice 1 : allumer les LED une par une de gauche à droite et de haut en bas.	codo-MATRICE3-EX1.hex
	Exercice 2 : reprendre l'exercice 1, en désactivant chaque LED 1 seconde après avoir été activée.	codo-MATRICE3-EX2.hex
	Exercice 3 : activer toutes les LED et éteindre certaines LED une par une de telle sorte qu'un L apparaisse.	codo-MATRICE3-EX3.hex
Minuteur	Exercice 1 : au lancement du programme déclencher un compte à rebours de 10 secondes et jouer un son lorsqu'il est à zéro.	codo-Minuteurs-EX1.hex
	Exercice 2 : déclencher un compte à rebours de 10 s lorsque le bouton A est pressé.	codo-Minuteurs-EX2.hex
	Exercice 3 : reprendre l'exercice 2, et ajouter la possibilité de réinitialiser le compte à rebours lorsque le bouton B est pressé.	codo-Minuteurs-EX3.hex

Python

Les exercices proposés ci-dessous sont des exercices qui permettent de tracer des graphes de la température et de l'accéléromètre, cependant, ils fonctionnent seulement avec Windows 10.

Pour les exercices suivants, il s'agit d'une programmation en python standard. En effet, ceux sont les 6 premiers exercices de la carte micro:bit seule, adaptés en python standard. De ce fait, les motifs ne s'affichent plus sur la carte micro:bit mais à l'écran de l'ordinateur.

Ces corrections permettent d'étudier la librairie « turtle » de python mais également de découvrir les bases du langage python standard.

Thème	Enoncés	Corrections
Graphiques	Exercice 1 : tracer le graphique de la température en fonction du temps.	graph temp.py
	Exercice 2 : tracer le graphique d'accélération suivant les 3 axes en fonction du temps.	graph accel.py
	Exercice 3 : tracer le graphique de l'accélération suivant l'axe X en fonction du temps.	accel suivant x
Matrice LED 1 python standard	Exercice 1 : afficher un carré.	codo-MOTIFS-EX1.py
	Exercice 2 : afficher un carré et une seconde après un triangle.	codo-MOTIFS-EX2.py
	Exercice 3 : répéter l'affichage d'un carré, d'un triangle, puis d'une croix avec un intervalle de 1 seconde.	codo-MOTIFS-EX3.py
Bouton A et B python standard	Exercice 1 : afficher un triangle quand le bouton A est pressé.	codo-BOUTONS-EX1.py
	Exercice 2 : afficher un triangle quand le bouton A est pressé et effacer l'écran après 3 secondes.	codo-BOUTONS-EX2.py
	Exercice 3 : afficher une croix lorsque le bouton A est pressé, le nombre 15 lorsque le bouton B est pressé et la première lettre de votre prénom lorsque les boutons A et B sont pressés	codo-BOUTONS-EX3.py



www.a4.fr

Concepteur et fabricant de matériels pédagogiques