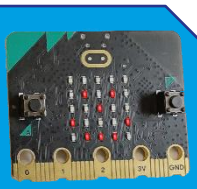
Défis de programmation avec la carte électronique micro :bit



E:\GD-PARENTS\CARTE_MICRO :BIT\FICHE_ANIMATION

Cet atelier permet d'apprendre à :

- Programmer pour automatiser un objet technologique.

- Identifier les objets du quotidien qui sont automatisés.

Vous programmerez par blocs dans MakeCode afin de contrôler des objets technologiques intégrant différents capteurs et actionneurs.

Cet atelier s'adresse principalement aux néophytes et débutants.

Cette séquence a été conçue et produite par :

Ver

SCIENCE UM

Ce document est diffusé sous licence Creative Commons

Vous êtes autorisé·e à partager (copier et redistribuer) le matériel à condition de respecter les termes énoncés ci-dessous.

Attribution — Vous devez créditer l'œuvre de manière appropriée (en indiquant le nom du projet et l'organisme de financement), fournir un lien vers la licence et indiquer si des modifications ont été apportées. Vous pouvez le faire de toute manière raisonnable, mais sans suggérer que l'auteur ou le licencié vous soutient ou approuve l'usage que vous faites de l'œuvre.

Pas d'utilisation commerciale — Vous n'êtes pas autorisé·e à faire un usage commercial de ce matériel, en tout ou partie.

Pas de modification — Si vous modifiez, transformez ou adaptez ce matériel, vous n'êtes pas autorisé·e à diffuser le matériel modifié.

Wallonie recherche SPW

Wallonie Relance

Table des matières

1. Durée estimée de l'activité 4
2. Public cible 4
3. Matériel nécessaire pour réaliser cette activité 4
4. Recommandations destinées à l'animateur·trice de l'atelier 5
5. Déroulement de l'activité 7
Étape 1 : Introduction – Exemples de machines programmables 7
Étape 2 : Présentation de la programmation – Concepts de base 8
Étape 3 : Présentation générale de la carte micro:bit 10
Étape 4 : Défis simples en autonomie 17
Étape 5 : Défis plus complexes en autonomie 19
Étape 6 : Lien avec les métiers et la vie quotidienne 19

Version provisoire

Dans notre quotidien, nous sommes entourés de machines qui semblent fonctionner "toutes seules", mais en réalité, elles obéissent à des instructions précises inscrites dans leur mémoire. Une machine à laver suit un programme de lavage, un feu tricolore alterne les couleurs selon une séquence définie, une voiture moderne utilise des capteurs pour freiner automatiquement ou guider le conducteur, et un robot aspirateur cartographie les pièces pour nettoyer efficacement. Ces exemples montrent que derrière chaque action automatisée se cache un programme.

L'objectif de cet atelier est de faire prendre conscience que la programmation est partout autour de nous, souvent invisible mais essentielle.

Ces machines ne sont pas magiques, elles sont programmées ! Et si on apprenait à leur parler ?

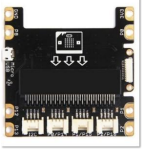
1. Durée estimée de l'activité

2,5 heures

2. Public cible

10-15 ans

3. Matériel nécessaire pour réaliser cette activité

1 ordinateur/tablette avec accès internet	
1 carte micro:bit	
1 Shield Grove	
1 module Grove PIR	
1 module Grove LED	
1 module Grove bouton	

Version provisoire

1 module Grove servomoteur	
1 module Grove relais	
1 module Grove 4-Digit Display	
1 module Grove AHT 20	
1 module Grove ultrasons	
1 module Grove luminosité	

Version provisoire

4. Recommandations destinées à l'animateur·trice de l'atelier

- Dans cette activité, nous présentons la carte micro:bit et les fonctionnalités utiles aux défis proposés. Si toutefois vous souhaitez en découvrir davantage sur cette carte programmable, nous vous conseillons les ouvrages suivants :

54 activités avec la carte micro:bit carte V2 : Programmation avec MakeCode et Vittascience
Dominique Nibart
978-2416019661

micro:bit Programmez la carte avec MakeCode et MicroPython,
Jean-Christophe QUETIN
ISBN : 978-2-409-02500-6

- Beaucoup de sites sont riches d'informations, en voici quelques exemples :
  - https://microbit.org/fr/
  - https://fr.vittascience.com/
  - https://ent2d.ac-bordeaux.fr/disciplines/sti-college/wp-content/uploads/sites/63/2023/01/Notice-dutilisation-micro_bit-v1-et-v2.docx.pdf

- En quelques mots, qu'est-ce que la programmation ?

La programmation, c'est l'art de donner des instructions à une machine pour qu'elle réalise des actions précises. Même si cela peut sembler complexe au premier abord, les bases sont accessibles à tous, et reposent sur quelques concepts fondamentaux.

Le programme : C'est une suite d'instructions que la machine exécute dans un ordre précis. Comme une recette de cuisine, chaque étape doit être claire et logique.

La boucle : Elle permet de répéter une action plusieurs fois. Par exemple, si on veut qu'une lumière clignote, on utilise une boucle pour répéter l'allumage et l'extinction.

La condition : Elle permet à la machine de prendre une décision. On utilise des phrases du type "Si… alors…". Par exemple : *Si le bouton est pressé, alors allumer la LED.*

L'événement : C'est une action extérieure qui déclenche une réaction. Par exemple, Lorsqu'un capteur détecte un mouvement, la machine peut réagir en allumant une alarme ou en affichant un message.

Les variables : Ce sont des "boîtes" dans lesquelles on peut stocker des informations (comme un score, une température, ou un nom), que l'on peut utiliser et modifier au cours du programme.

Les capteurs : Ils permettent à la machine de "sentir" son environnement : température, lumière, mouvement, inclinaison… et d'adapter son comportement en conséquence.

Version provisoire

5. Déroulement de l'activité

- Etape 1 : découverte d'exemples de machines qui nécessitent une programmation. Montrer qu'il y a des cartes programmables. Apprendre à leur "parler"
- Etape 2 : présentation de la programmation et de son mode de fonctionnement grâce au jeu Blockly
- Etape 3 : présentation de la carte micro:bit (interface, connexion, blocs de programmation)
- Etape 4 : des défis simples en autonomie (exemples : déverrouillage, clignotant, allumage radio, essuie-glace)
- Etape 5 : des éfis plus compliqués en autonomie (exemples : phares, barrière, caméra recul (ultrasons), place de parking)
  - Etape 6 : Lien métier, programmation autour de nous

Étape 1 : Introduction – Exemples de machines programmables

Objectif :

Amener à comprendre que de nombreuses machines que nous utilisons ou croisons au quotidien fonctionnent grâce à des programmes intégrés dans des cartes électroniques.

Déroulement :

1. Discussion ouverte

Question de départ : *"À votre avis, quelles machines autour de vous sont automatisées ?"*

2. Présentation d'exemples concrets
   - Machine à laver : elle suit un programme de lavage (durée, température, essorage).
   - Feu tricolore : il alterne les couleurs selon une séquence programmée.
   - Voiture moderne : elle utilise des capteurs pour freiner automatiquement, suivre une route via GPS, etc.
   - Robot aspirateur : il cartographie l'espace, évite les obstacles, revient à sa base.

Pour chaque exemple, poser la question : *"Comment sait-elle quoi faire ?"*
→ amener l'idée de programme.
"Ces machines ne sont pas magiques : elles obéissent à des instructions écrites dans leur carte électronique."
Expliquer brièvement ce qu'est une carte électronique programmable (comme un petit cerveau qui exécute des ordres) et montrer la carte micro:bit que nous programmerons dans la suite de cette activité.

Version provisoire

“Et si on apprenait à leur parler ?”

Étape 2 : Présentation de la programmation – Concepts de base

Objectif :

Introduire les bases de la logique de programmation avec Blockly

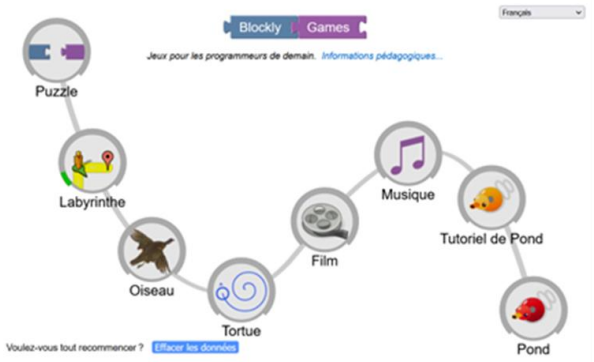
Déroulement :

Le site Blockly Games (https://blockly.games/) vous permet de vous initier à la logique de programmation sans connaissance préalable au travers de différents jeux. Pour cet atelier, les défis “labyrinthes” permettent de poser deux notions utiles à la programmation de la carte micro:bit utilisée par la suite :

- La notion de boucle, qui permet de répéter tout ou partie d’un programme
- Les conditions, qui permettent de réaliser une action SI une condition est rencontrée.

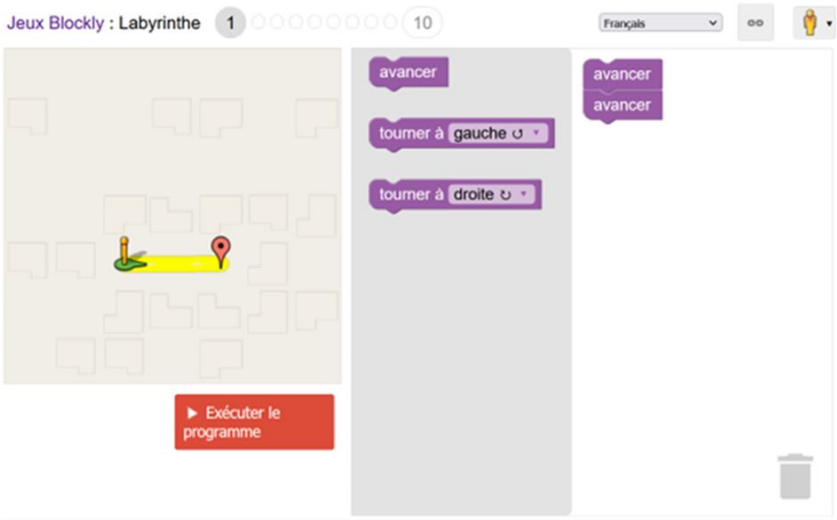
Les actions sont organisées dans une interface graphique dite de “programmation par blocs”, que la programmation de la carte utilisera elle aussi grâce au logiciel MakeCode

Le jeu labyrinthe propose 10 niveaux de difficulté croissante.



Au niveau 1, le but est d’amener le petit bonhomme sur le point orange en programmant ses mouvements à l’aide des blocs disponibles. Ici, “avancer” signifie avancer d’une case. Il faudra donc en cliquer-glisser deux dans la zone de construction du programme (à droite de l’écran) avant de cliquer sur “exécuter le programme” pour le tester.

version provisoire



Au niveau 3, où apparaît le bloc “répéter”.

Les actions qui seront intégrées à l'intérieur du bloc seront répétées encore et encore, ici jusqu’à ce que notre bonhomme ait atteint sa destination finale.

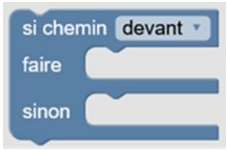
> ℹ *En programmation, la répétition permet d'économiser de la place dans la mémoire en écrivant une seule fois une série d’actions qui sera répétée, c’est ce que l’on appelle une boucle.*

Au niveau 6, où apparaît le bloc “Si”

Les actions qui seront intégrées à l'intérieur du bloc ne seront alors effectuées que si la condition est rencontrée. Ici, c’est “s'il y a un chemin vers la gauche”.

> ℹ En programmation, une condition permet à la machine de prendre une décision. Elle applique le principe du “Si… alors…”. Par exemple : *Si le bouton est appuyé, alors allume la lumière.* Cela signifie que l’action ne se produit que si une certaine situation est vraie. C’est comme poser une question à la machine : *“Est-ce que le bouton est appuyé ?”* — si oui, elle fait quelque chose ; sinon, elle ne fait rien ou fait autre chose.

Version provisoire

Au niveau 9, où l'on ajoute la possibilité d'un "sinon" 

A la manière d'un aiguillage, ce bloc permet de spécifier non seulement les actions à effectuer si une condition est rencontrée, mais aussi que faire dans le cas contraire.

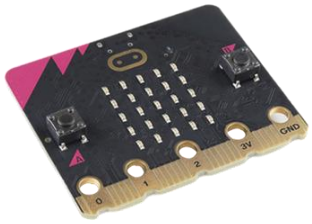
À nouveau, il existe d'autres sites intéressants avec des jeux d'initiations à la programmation :

- L'heure du code : https://code.org/
- Scratch Junior: https://jfo8000.github.io/ScratchJr-Desktop/
- Code Karts (Android) : https://play.google.com/store/apps/details?id=com.edokiacademy.babycoding&hl=en_US&gl=US
- Run Marco : https://runmarco.com/

Étape 3 : Présentation générale de la carte micro:bit

Objectif :

Se familiariser avec l'outil qu'ils vont utiliser.



Déroulement :

1. Présentation de la carte micro:bit

Créée en 2012, cette petite carte électronique se veut un outil de découverte créative et d'initiation à la technologie.
Il existe aujourd'hui deux versions de cette carte microcontrôleur nommées V1 et V2 (la version de la carte est indiquée en bas à droite à l'arrière de la carte).
Si elles se ressemblent beaucoup, la V2 dispose de quelques évolutions :

Fonctionnalités

Cette version embarque plus de mémoire RAM, de mémoire Flash mais également un mini haut-parleur, un micro, un bouton-poussoir tactile capacitif.
Elle comporte, comme sa version précédente, deux boutons poussoirs, une matrice 5x5 LED, 22 broches dont 3 analogiques, un accéléromètre, un magnétomètre et un module Bluetooth 5.0

Connectique

Version provisoire

Les broches se trouvent en bas de la carte :

- 3 entrées/sorties analogiques marquées 0, 1 et 2 ainsi que la broche 3 V et GND. Ces broches sont facilement accessibles avec fiches bananes ou des pinces crocodiles.
- Il est possible d'utiliser les broches supplémentaires (plus petites) à l'aide du dispositif Shield Grove détaillé plus loin.

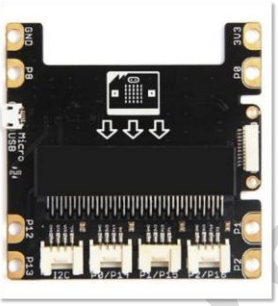
Programmation

La programmation se fait depuis un navigateur web via l'IDE MakeCode pour les débutants ou en JavaScript ou Micro Python pour les utilisateurs plus expérimentés.

Le Bluetooth permet aussi la programmation depuis un smartphone ou une tablette via une application iOS et Android.

Version provisoire

2. Descriptif détaillé des modules électroniques Grove



Ce Shield est une carte d'interface permettant de raccorder facilement, rapidement et sans soudure les capteurs et les actionneurs Grove sur la carte micro:bit V1 ou V2.

Les quatre prises présentes sur le Shield Grove micro:bit proposent les connexions Grove suivantes :

- Une prise I2C à gauche, destinée des composants utilisant ce protocole de communication comme l'écran LCD par exemple.
- Une prise P0(câble jaune) / P14 (câble blanc)
- Une prise P1(câble jaune) / P15 (câble blanc)
- Une prise P2(câble jaune) / P16 (câble blanc)

Certains composants, comme l'afficheur 4-Digit Display, utilisent les deux ports de données de la connectique. Il faudra alors renseigner les deux dans la programmation en entrée ou en sortie.
Pour les autres composants n'utilisant qu'un port, c'est par défaut celle du câble jaune, donc P0, P1 ou P2.
En cas de doute, regardez la prise du composant : l'utilisation de chaque câble y est indiquée. « Nc » signifie *non connecté* et « Sig » qu'un *signal* transite par ce câble. Vous n'avez plus qu'à suivre la couleur du câble jusqu'à la prise du Shield pour voir à quelle prise de la micro:bit ce signal est envoyé.

Le module PIR détecteur de présence compatible Grove basé sur un capteur à infrarouges passifs qui détecte la présence d'une personne ou d'un animal. Il se branche sur les ports P0/P14, P1/P15 et P2/P16 du Shield Grove

Le module Led compatible Grove permet l'utilisation d'une LED à partir d'une sortie digitale. Il se branche sur les ports P0/P14, P1/P15 et P2/P16 du Shield Grove.
Attention au sens de la LED lorsque vous l'insérez dans son support : le coté plat doit être du même côté que le sur le dessin du module.

Le module bouton compatible Grove est un bouton-poussoir qui fait passer un signal de l'état bas à l'état haut lors d'une pression. Il se branche sur les ports P0, P1 ou P2 du Shield Grove

Le module servomoteur doté des palonniers et visserie pour mettre en mouvement vos projets. Il se branche sur les ports P0, P1 et P2 du Shield Grove. Alimentation : 4,8 à 6 Vcc, Couple : 1.8 kg.cm @ 6 Vcc, Vitesse : 0,10 s/60° @ 6 Vcc

Version provisoire

Le module relais agit comme un interrupteur normalement ouvert. Vous pouvez contrôler un circuit de haute tension avec une basse tension, ici 3,3v avec la micro:bit. Il y a un indicateur Led sur la carte, qui s'allume lorsque le relais laisse passer le courant sur le circuit commandé. Il se branche sur les ports P0, P1 ou P2 du Shield Grove

Le module 4-Digit Display comporte 4 afficheurs 7 segments. Il se branche sur les ports P0/P14, P1/P15 et P2/P16 du Shield Grove

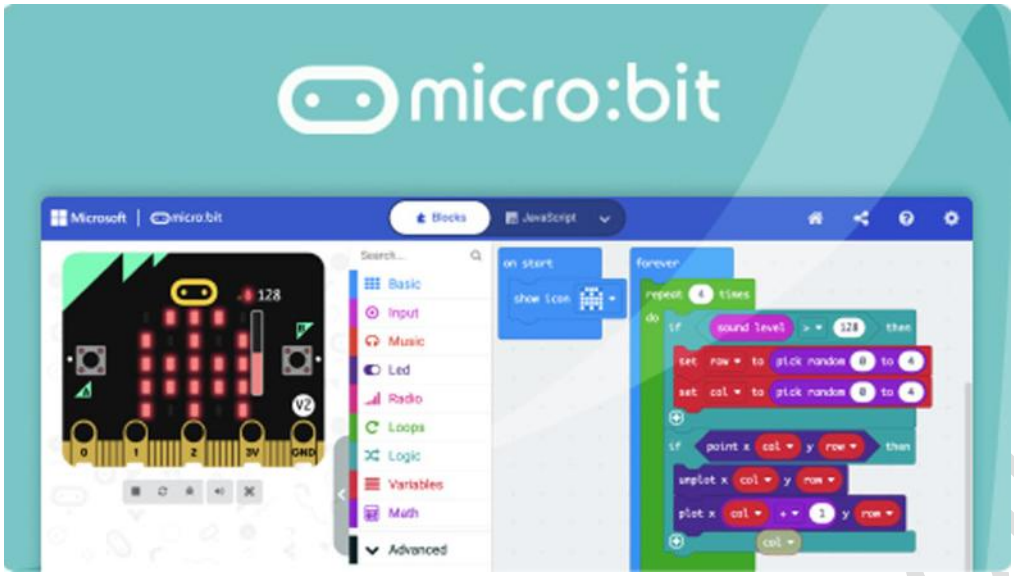
Le module capteur AHT 20 permet la mesure de la température et de l'humidité. Il mesure la température entre -40 à 85 °C et l'humidité de 0 à 100% HR avec une précision de ±0,3°C et de ±2% HR. Il se branche sur le port I2C du Shield Grove.

Le module capteur de distance à ultrasons est un appareil qui utilise des ondes ultrasoniques pour mesurer la distance entre lui-même et un objet. Il a une portée de 3 cm à 350 cm et une précision allant jusqu'à 2 mm

Le module capteur de lumière permet de détecter la présence de lumière. La tension de sortie analogique évolue de 0 à +Vcc suivant l'intensité lumineuse mesurée.

3. Le navigateur web de programmation

Version provisoire

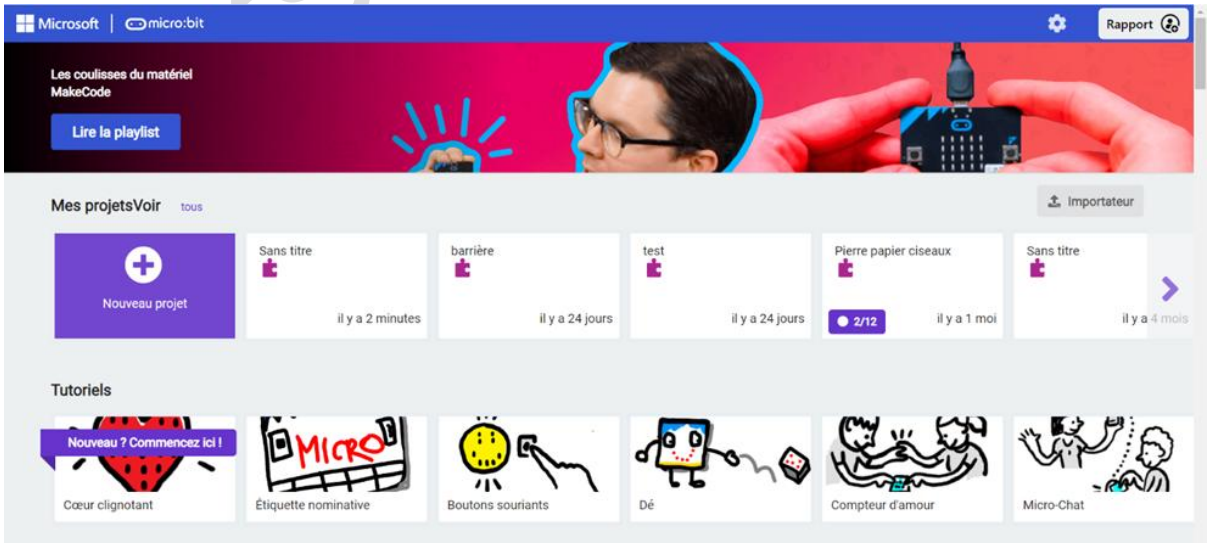


Nous choisissons ici de programmer la carte micro:bit avec la plateforme en ligne Makecode :

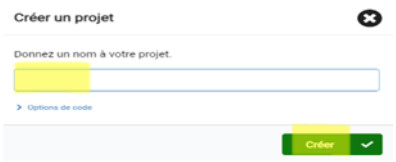
Si vous travaillez sur ordinateur, nous vous recommandons d'y accéder avec le navigateur Chrome, qui permet un transfert simplifié des programmes depuis l'ordinateur vers la mémoire de la carte micro:bit Microsoft MakeCode for micro:bit (microbit.org)

Pour télécharger l'application pour appareils Android : micro:bit – Applications sur Google Play
Pour télécharger l'application pour appareils iOS : micro:bit dans l'App Store (apple.com)

Une fois le logiciel téléchargé (ou plateforme en ligne ouverte), vous accédez à l'interface suivante :

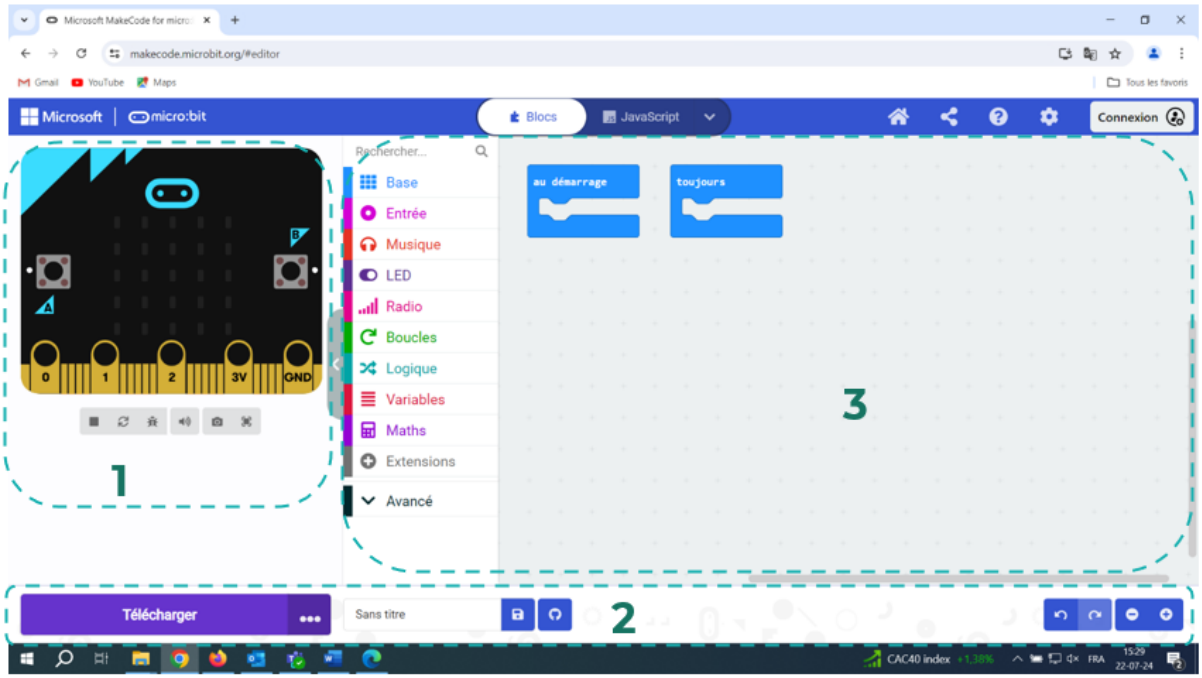


version provisoire



Cliquez sur « **Nouveau Projet** » nommez-le et créez-le.

Vous arrivez alors sur l'interface de programmation suivante :

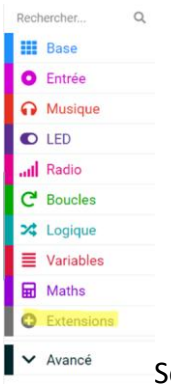


On constate que l'écran est composé de 3 parties, une partie de simulation (1), une partie de configuration (2) et une partie de programmation (3).

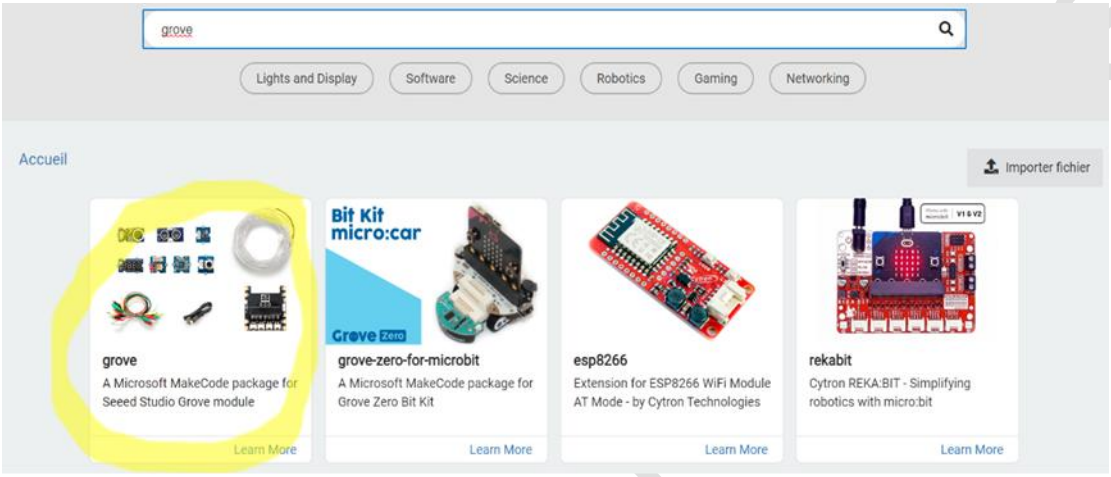
Vous pouvez déjà faire, avec la bibliothèque existante de blocs, un grand nombre de défis de programmation sur la carte micro:bit.

Cependant, si vous voulez programmer les modules Grove cités plus haut, il faudra aller chercher des blocs spécifiques. Pour ce faire :
Cliquez sur "**Extensions**" dans la partie de programmation (3)

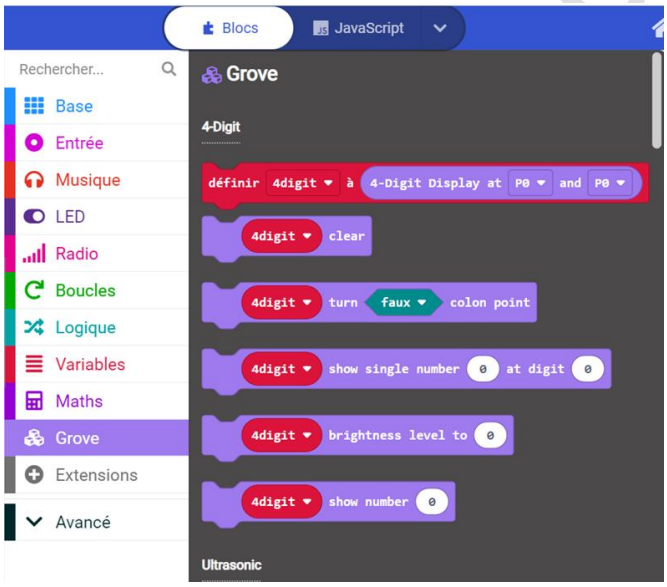
Version



Sélectionnez Grove dans la bibliothèque



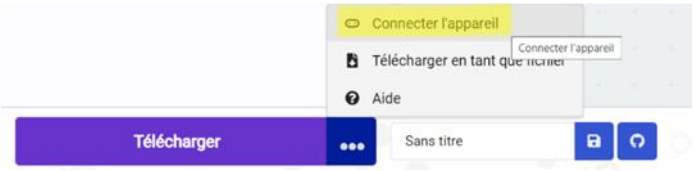
Vous avez maintenant accès à certains blocs de programmation Grove supplémentaires



Pensez aussi à relier la carte micro:bit à votre ordinateur (ou autre appareil) à l'aide du câble USB de branchement ou par Bluetooth

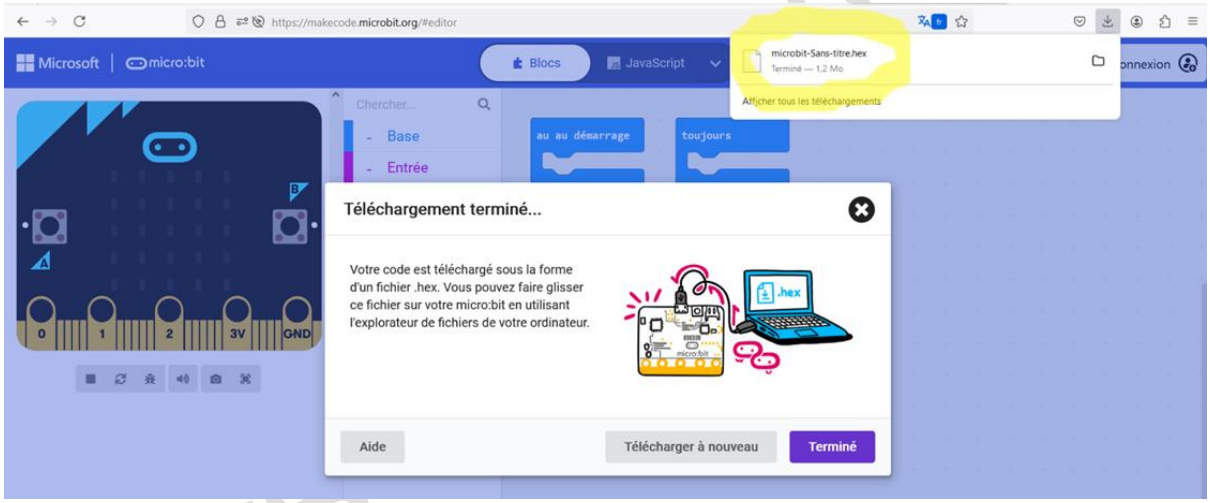
Depuis un ordinateur, connectez la carte micro:bit par USB :

- Si vous utilisez Google Chrome, vous pouvez cliquer sur les « **…** » à droite du bouton télécharger pour sélectionner « **connecter l'appareil »**, et suivez les étapes pour appairer votre carte. Une fois appairé, vous pourrez transférer les programmes en cours en cliquant sur « **télécharger** ».



- Si vous utilisez
  - Windows 7,8 et 10, et pas Google Chrome
  - Linux et Chrome,
  - IOS sous Mac sans Google chrome

  Téléchargez le programme dans votre dossier de téléchargement local, puis cliquez-déposez-le dans la micro:bit à l'aide de l'explorateur de fichiers.



- Si vous utilisez une tablette et l'application micro:bit, suivez la procédure décrite dans l'application pour appairer la carte micro:bit en Bluetooth, puis téléchargez le programme en Bluetooth.

Étape 4 : Défis simples en autonomie

Objectif :

Mettre en pratique les bases de la programmation.

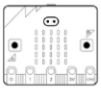
Déroulement :

Proposition de défis :

- Déverrouillage : Affichage d'un message au démarrage de la voiture

Version préliminaire

Défi 1 :



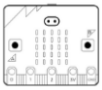
Vous rentrez dans une nouvelle voiture et celle-ci vous accueille.

Programmez la Micro:bit pour :

- **Au démarrage** elle vous **affiche un message** pour dire : **"Bonjour + votre prénom "**

– Allumage radio : Simuler une radio avec sons et pause

Défi 2 :



Vous pouvez supprimer le code précédent.

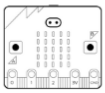
Une fois dans la voiture, vous décidez de brancher la radio :

- Lorsque **sur le Logo appuyé,** demander de **répéter 2 fois ,** lancez la musique en **jouant une mélodie jusqu'à la fin** (vous pouvez réaliser votre propre mélodie).
- Après la répétition**,** faire **une pause de 100 ms** et **arrêter tous les sons.**

Le bloc boucle permet de répéter toutes les actions qui se trouvent à l'intérieur de celui-ci.

– Volant-capteur gyroscopique : Réagir à l'inclinaison de la carte

Défi 3 :



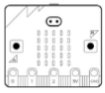
Vous pouvez supprimer le code précédent.

Un fois la voiture en marche, vous devez voir si le volant de celle-ci fonctionne correctement.

- **Lorsque vous inclinez la carte vers la gauche,** demandez de **montrer les LEDs** pour qu'elles fassent **un logo de volant vers la gauche.**
- **Lorsque vous inclinez la carte vers la droite,** demandez de montrer **les LEDs** pour qu'elles fassent **un logo de volant vers la droite.**

– Feux clignotants : Réagir si un bouton est enclenché

Défi 4 :



Vous pouvez supprimer le code précédent.

Une fois en route, vous devez activer vos feux clignotants afin d'indiquer la direction que vous allez prendre :

- Lorsque **le bouton A est pressé,** demandez de **répéter 4 fois , montrer** une **flèche à gauche**, **jouer une tonalité** (au choix) et **montrer les LEDs vides.**
- Lorsque **le bouton B est pressé**, demandez de **répéter 4 fois** , **Montrer** une **flèche à droite**, **jouer une tonalité** (au choix) et **montrer les LEDs vides**.

Provisoire

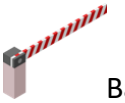
Étape 5 : Défis plus complexes en autonomie

Objectif :

Aller plus loin avec des projets intégrant plusieurs concepts et des capteurs/ actionneurs externes à la carte

Déroulement :

Propositions de défis :

- Phares : Allumer LED selon la luminosité
-  Barrière : Actionner un servomoteur avec bouton
- Caméra de recul : Utiliser capteur à ultrasons pour afficher la distance
- Place de parking : Contrôler la LED selon la détection de présence

Étape 6 : Lien avec les métiers et la vie quotidienne

Objectif :

Montrer l'utilité de la programmation dans le monde réel.

Déroulement :

Présenter brièvement les **métiers concernés** :

- Développeur.se embarqué.e : programme les cartes électroniques dans les objets.
- Roboticien.ne : conçoit et programme des robots pour l'industrie ou la santé.
- Ingénieur.e automobile : travaille sur les systèmes intelligents des véhicules.
- Technicien.ne domotique : installe et programme des systèmes pour la maison (chauffage, sécurité, éclairage).
- Utiliser des capsules vidéo ou des témoignages (si disponibles) pour illustrer ces métiers.

https://drive.google.com/drive/folders/1w0fmy7SQ4oiBmVeN7NYWEwwG7HGQ8QBq

Version provisoire

Prolongements possibles :

Programmer sa station météo :

- https://fr.vittascience.com/shop/173/station-meteo---version-micro-bit
- https://ressources.camexia.org/comprendre-liot-en-samusant-un-atelier-complet-avec-la-carte-microbit-v2/

Ressources complémentaires :

- Vittascience : plateforme éducative pensée pour l'apprentissage du codage, qui propose des outils innovants pour l'enseignement.

https://fr.vittascience.com/

- Gotronic : robotique et composants électroniques

https://www.gotronic.fr/art-kit-inventor-electronic-mb-94004-28479.htm

- 1, 2, 3… Codez ! : 2 tomes qui visent à initier élèves et enseignants aux sciences informatiques

https://fondation-lamap.org/projet/123-codez

Ils sont disponibles gratuitement en prêt sur le site sciences à emporter

- Support FMTTN : site qui répertorie des propositions d'activités d'apprentissage répondant aux attendus du référentiel numérique et technologique

https://support-fmttn.be/

Version provisoire