

CHAPITRE 3

BASES DE PROGRAMMATION LOGICIEL ET MATÉRIEL

LA
-

Auteurs : Jonathan Baudin, Sébastien Nedjar

Grâce aux chapitres précédents, vous avez à l'esprit les piliers pédagogiques de l'approche Let's STEAM (inclusion, équité, approche par expérimentation), nous vous proposons de découvrir les outils d'apprentissage de la programmation qui sont utilisés dans nos propositions d'activités : l'éditeur MakeCode et la carte programmable STM32. Cette présentation vous donnera les informations initiales permettant de démarrer vos projets à partir de ces outils logiciel et matériel.



Les choix technologiques proposés dans ce manuel ont été réalisés car ils ont un réel intérêt pédagogique dans le cadre du déploiement de projets motivants utilisant la programmation dans les écoles secondaires, proposant des échelles de complexité variées. Ce chapitre abordera donc :

- **L'éditeur Microsoft MakeCode** : une plateforme gratuite et open-source permettant de créer des expériences d'apprentissage motivantes. Lien pour accéder à l'éditeur MakeCode Let's STEAM : <https://makecode.lets-steam.eu/>
- **La carte STM32 IoT Node** : une carte programmable embarquant des capteurs et des outils variés et pertinents, utiles pour se lancer dans des projets complexes en classe.

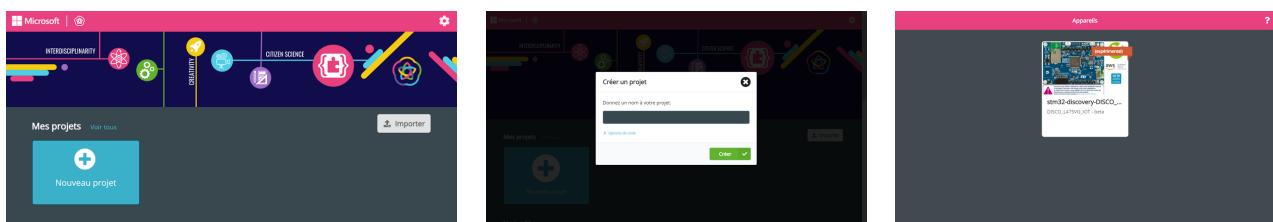


DÉCOUVRIR LA PLATEFORME MAKECODE

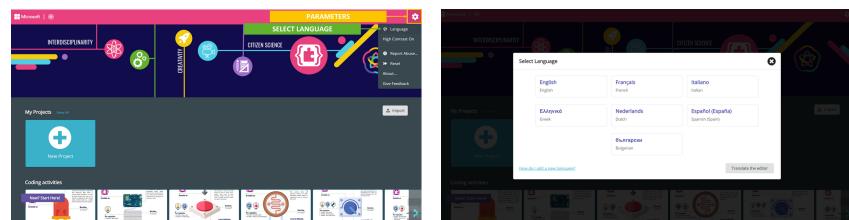
Lorsque vous entrez sur l'interface MakeCode de Let's STEAM, vous arrivez directement sur la page d'accueil. Sur cette page, vous pouvez créer un nouveau projet, ouvrir un projet existant si vous avez déjà travaillé sur l'éditeur, voir les cartes supportées et découvrir des ressources inspirantes.

Lorsque vous créez un projet, il est important de le **nommer avec un titre clair et compréhensible**, vous permettant d'annoncer clairement l'objectif du programme.

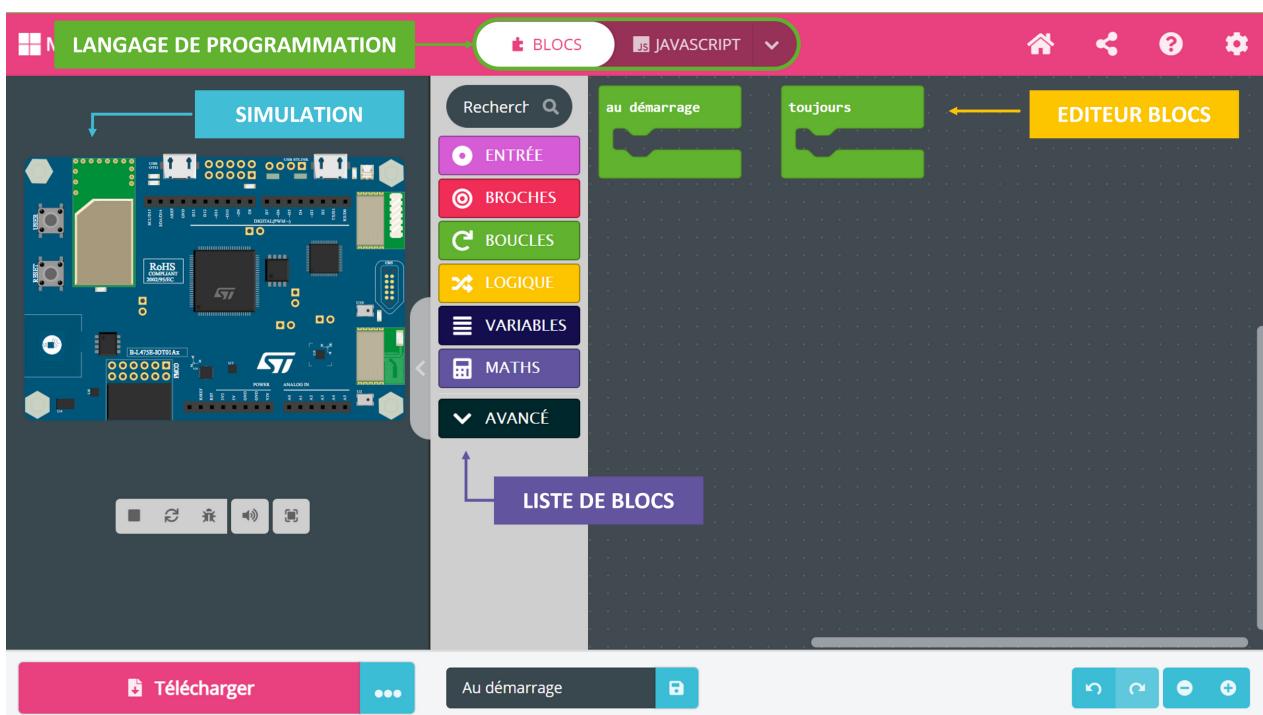
Sur l'écran suivant, vous devrez choisir la **carte sur laquelle vous allez travailler**. Sur les fiches d'activités de Let's STEAM, tous les exemples ont été développés à l'aide de la carte **STM32 IoT Node** (la carte est surlignée en orange dans l'image ci-contre).



Si l'interface chargée est affichée en anglais au lancement de MakeCode, vous pouvez changer la langue en cliquant sur le bouton "paramètre" afin de voir les versions supportées.



Une fois la carte sélectionnée, vous aurez alors accès à l'éditeur, présenté ci-dessous :



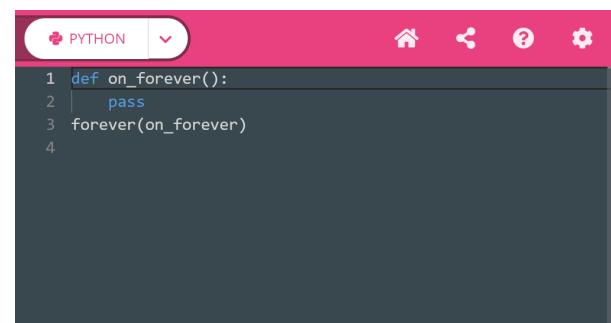
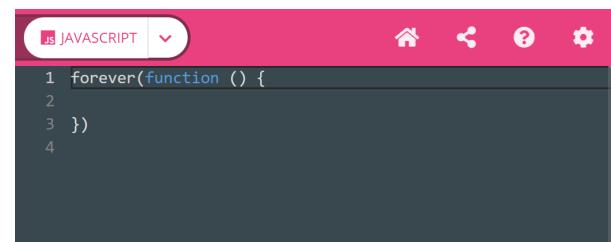
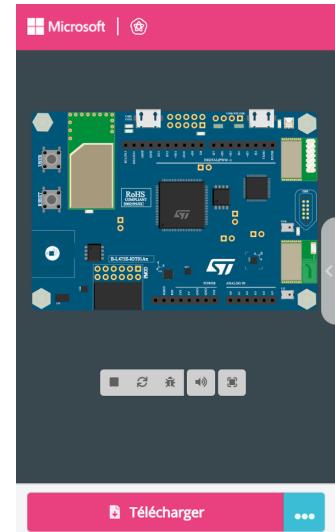


Voici les composants de l'éditeur :

- Le **SIMULATEUR** (à gauche de l'éditeur) : un simulateur interactif fournit aux élèves un retour immédiat sur comment leur programme fonctionne et leur permet de tester et de déboguer leur code.
- La **LISTE DES BLOCS** au centre, pouvant être utilisés dans votre programme, ainsi qu'un champ de recherche.
- **L'ÉDITEUR DE BLOCS** sur la partie droite, qui comprend déjà deux fonctions communes à toutes les activités : “**on start**” et “**forever loop**”. Les élèves qui débutent dans le programmation peuvent commencer par des blocs de couleur qu'ils peuvent glisser et déposer sur leur espace de travail pour construire leurs programmes.

Dans l'éditeur, vous pourrez également choisir le mode de programmation :

- **Par le biais de blocs** (voir la fiche d'activité R1AS1 - Faire clignoter une LED)
- **Par l'intermédiaire d'un éditeur JavaScript** (toutes les fiches d'activité proposées dans ce manuel comprendront le code en JavaScript qui pourra être directement copié-collé dans cet éditeur).
- **Par le langage Python** pour les élèves plus avancés.





Voici la liste des blocs de base disponibles sur l'éditeur MakeCode de Let's STEAM. Vous aurez un aperçu plus précis de la fonction de chaque bloc dans les diverses fiches d'activités proposées dans ce manuel :

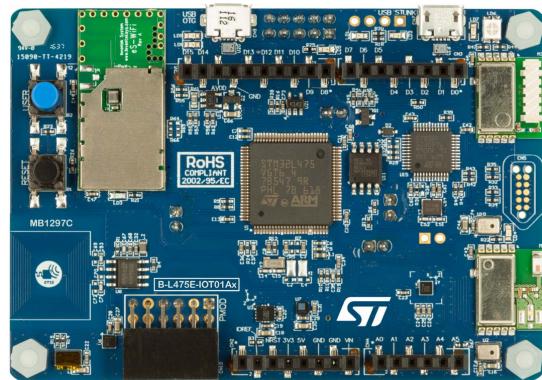
Entrée	ENTRÉE	Utiliser un capteur dans votre programme (comme un bouton, un thermomètre, etc.).
Broches	BROCHES	Interagir directement avec les broches (<i>ou plus fréquemment appelées pin</i> - cf fiches d'activités) et modifier leur état
Control	CONTRÔLE	Gérer l'exécution des événements
Boucles	BOUCLES	Mettre en œuvre les répétitions
Logique	LOGIQUE	Effectuer des tests, des comparaisons et des opérations de logique booléenne.
Variables	VARIABLES	Créer des variables et des compteurs
Maths	MATHS	Effectuer divers calculs mathématiques
Fonctions	FONCTIONS	Créer des sous-programmes
Tableaux	TABLEAUX	Créer une valeur ou un texte dans un tableau
Texte	TEXTE	Modifier les textes
Console	CONSOLE	Afficher les données
Extensions	EXTENSIONS	Accéder à la liste des extensions disponibles dans la version de MakeCode
Magnetics	MAGNETICS	Gérer les communications
Datalogger	DATALOGGER	Créer un jeu de données pour enregistrer les données des capteurs
LCD	LCD	Afficher du texte ou des informations sur un écran (LCD)
OLED	OLED	Afficher du texte ou des informations sur un écran (OLED)
Music	MUSIC	Extension pour jouer de la musique



DÉCOUVRIR LA CARTE STM32 IOT NODE ET SON ENSEMBLE DE CAPTEURS

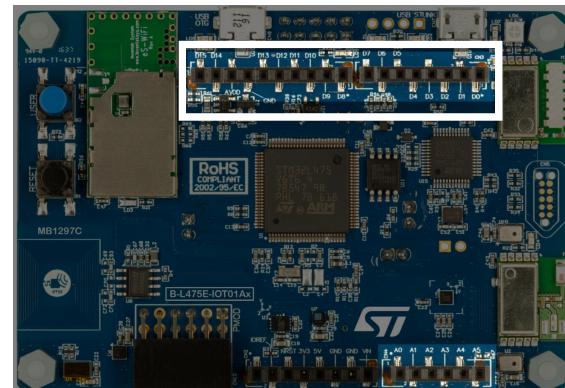
La carte "**STM32 Iot Node**" est une carte programmable, ce qui signifie qu'elle est capable d'exécuter des programmes créés par l'utilisateur.

Pour exécuter ce programme, la carte dispose d'un "**microcontrôleur**", qui est en quelque sorte son cerveau (voir ci-contre). Par exemple, Le nom du microcontrôleur de notre carte présentée ici est : **STM32L475VG**.



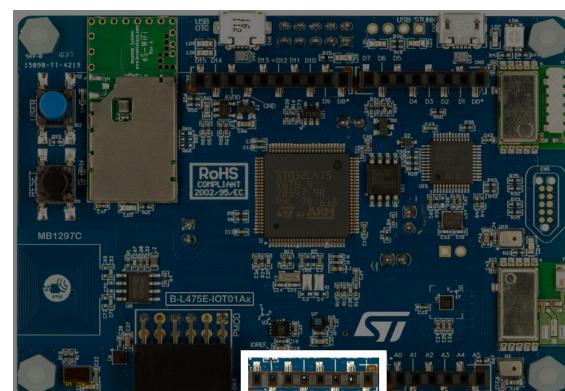
LES GPIO

Comme nous pouvons le constater, il y a beaucoup de "**pattes**" ou de "**broches**" sur la carte, appelées "**General Purpose Input / Output**" (ou **GPIO** en abrégé). Il est possible de les utiliser pour interagir avec des éléments externes. Même s'il y a beaucoup de **GPIO**, il n'est pas possible de tous les utiliser. Les **GPIO** utilisables sont situés en haut et en bas de la carte. Les blocs noirs percés sont appelés "**blocs de broches**". En regardant attentivement, nous pouvons remarquer les noms des **GPIO** inscrits autour (par exemple en bas à droite : "D0, D1, D2, D3, ..., A0, A1, A2, ...").



Nous découvrirons les différences entre les broches Ax (A0, A1, ...) et Dx (D0, D1, D2, ...) plus loin dans les activités.

Il reste un autre bloc de broches, celui-ci est spécial, c'est un "**power pinout block**". Nous pouvons utiliser ces broches pour alimenter des capteurs ou des actionneurs (comme un moteur, une lumière, etc.).



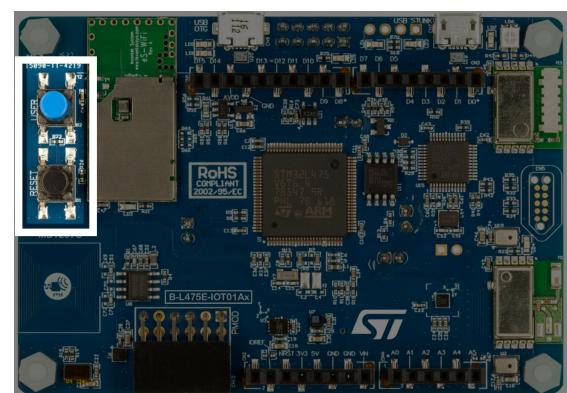
L'inscription sur le dessus du bloc de broches informe sur la manière de l'utiliser. Le "5V" correspond au "+" (pôle positif) d'une batterie et le "**GND**" (abréviation de "Ground") au "-" (pôle négatif).



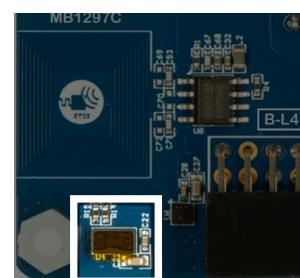
LES PÉRIPHÉRIQUES

La différence entre le nombre de GPIO disponibles via le bloc de broches et le nombre de pattes du microcontrôleur s'explique par la présence de multiples périphériques déjà connectés au microcontrôleur, disponibles sur la carte "STM32 IoT Node" elle-même. La présence de tous ces périphériques rend cette carte particulièrement accessible, car elle permet de mettre en œuvre un large éventail d'activités, des plus simples aux plus complexes, et des plus basiques aux plus ludiques. C'est un véritable atout pour réaliser des activités entraînantes en classe.

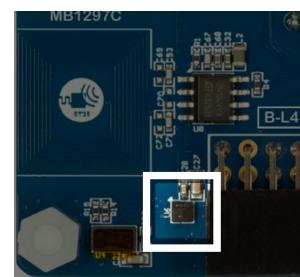
- Boutons:** Sur le côté gauche de la carte, vous trouverez deux boutons. Le bouton noir est le bouton **RESET**, permettant au programme de redémarrer si nécessaire. L'autre (bleu) peut être utilisé dans un programme pour détecter quand l'utilisateur appuie sur ce bouton (appui court, appui long, relâchement, etc.). Il peut être utile pour créer des **interactions simples avec l'utilisateur**, comme un **buzzer** dans le cadre d'organisation de concours à l'aide de cette carte.



- Capteur de distance :** Dans le coin inférieur gauche de la carte, juste à droite de la vis en nylon, vous pouvez trouver un capteur pour mesurer la distance. Il est officiellement appelé "**temps de vol**" (*time of flight*) parce qu'il mesure le temps que met un rayon laser à faire des allers-retours (voler) entre le capteur et un objet.



- Capteur de température et d'humidité :** Sur la droite du capteur "temps de vol", on trouve un capteur à la fois **thermomètre** et **hygromètre** ("2 en 1"). Cela peut être utile pour mettre en œuvre des activités liées à la surveillance de la chaleur ou pour aborder des notions de météorologie.





- Capteur accéléromètre et gyroscope** : Juste au-dessus du "power pinout block", se trouve un capteur à la fois accéléromètre et gyroscope ("2 en 1"). L'accéléromètre est utilisé pour mesurer l'accélération. Vous pouvez l'utiliser pour détecter les mouvements de la carte (par exemple, si la carte est secouée). Le gyroscope donne des informations sur l'inclinaison de la carte. Ce capteur fonctionne sur trois axes (X, Y et Z), ce qui implique qu'il est possible de détecter les mouvements dans l'espace 3D.



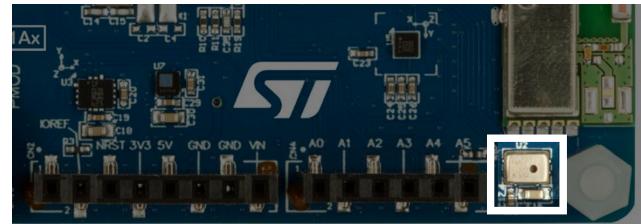
- Capteur de pression atmosphérique** : À côté du capteur accéléromètre et gyroscope, vous trouverez un petit capteur appelé baromètre. Ce capteur nous donne la valeur de la pression atmosphérique.



- Capteur magnétométrique** : À côté du capteur de pression atmosphérique, vous pouvez voir le magnétomètre. Il est utilisé pour récupérer la valeur d'un champ magnétique. Il peut également mesurer des valeurs sur trois axes (X, Y et Z).



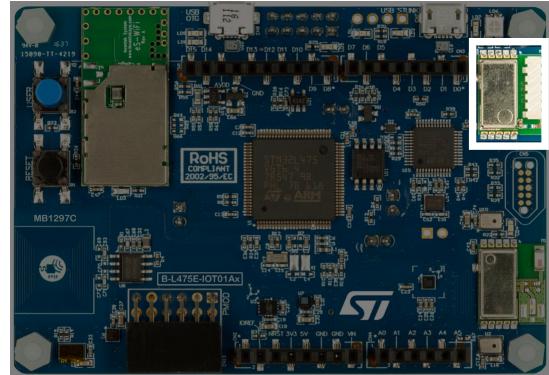
- Microphone** : Dans le coin en bas à droite, vous pouvez voir le microphone, utile pour capturer des sons.





LES MODULES

- **Module Bluetooth** : En haut à droite de la carte, vous pouvez trouver le module bluetooth. Il peut être utilisé pour communiquer et échanger des données avec d'autres appareils (comme une autre carte STM32 IoT Node, ou votre téléphone).



- **Connecteurs Micro-USB** : En haut de la carte, vous pouvez voir deux **connecteurs micro-USB**. Le port USB de droite est celui que vous utiliserez le plus souvent, car il permet de connecter la carte à votre ordinateur et de transférer le programme que vous aurez fait sur MakeCode au microcontrôleur. Le port de gauche, appelé "**port USB OTG**", permet de programmer la carte pour qu'elle agisse et soit reconnue comme un autre dispositif tel qu'un clavier, une souris ou une manette de jeu.

