Comprendre l’IoT en s’amusant : un atelier complet avec la carte Microbit V2

Et si vous pouviez visualiser en temps réel la température de votre salon, de votre balcon ou même d’un terrarium, directement depuis votre smartphone ? Bonne nouvelle : ce projet est à la portée de tous, même si vous n’avez jamais programmé une carte électronique de votre vie.

Dans cet article, nous allons vous guider pas à pas pour prendre en main la carte Microbit V2, une petite carte ludique et puissante conçue pour l’apprentissage de l’électronique et du code. À l’aide d’un capteur et d’une connexion Wi-Fi, vous apprendrez à envoyer des données vers ThingSpeak, une plateforme gratuite qui permet d’afficher vos mesures en ligne, sous forme de graphiques consultables partout, y compris sur votre téléphone.

Pas besoin d’être ingénieur ou geek confirmé : ici, pas de jargon inutile, pas de lignes de code obscures. Juste une méthode simple, pensée pour les curieux, les enseignants, les jeunes makers et tous ceux qui veulent découvrir le monde des objets connectés en douceur.

Prêt à transformer votre Microbit en mini station météo connectée ? Suivez le guide.

Le matériel dont vous aurez besoin

Avant de commencer, voici la liste des éléments à rassembler pour suivre ce projet :

- Une carte Micro:bit V2, de préférence dans un kit de démarrage (avec câble USB et support piles)
- Un module IoT:bit EF03426 : il permet de connecter facilement des capteurs et d’ajouter une connexion Wi-Fi à la Microbit
- Une sonde de température étanche DS18B20 (réf. WPSE324) : idéale pour mesurer la température de l’air, de l’eau ou du sol
- Un câble Grove vers Octopus (réf. 11096) : pour relier la sonde au module IoT:bit sans soudure
- Une source d’alimentation externe (pack de piles ou batterie USB) : nécessaire pour que le projet fonctionne sans être relié à l’ordinateur

Coût total : Entre 30€ et 70€ selon les fournisseurs.

Avec tout ça, vous êtes prêt à passer à l’étape suivante : visualiser comment tout se connecte ensemble.

Aperçu du projet

Avant de tout brancher, prenons une minute pour comprendre comment le projet fonctionne dans son ensemble.

L’idée est simple :

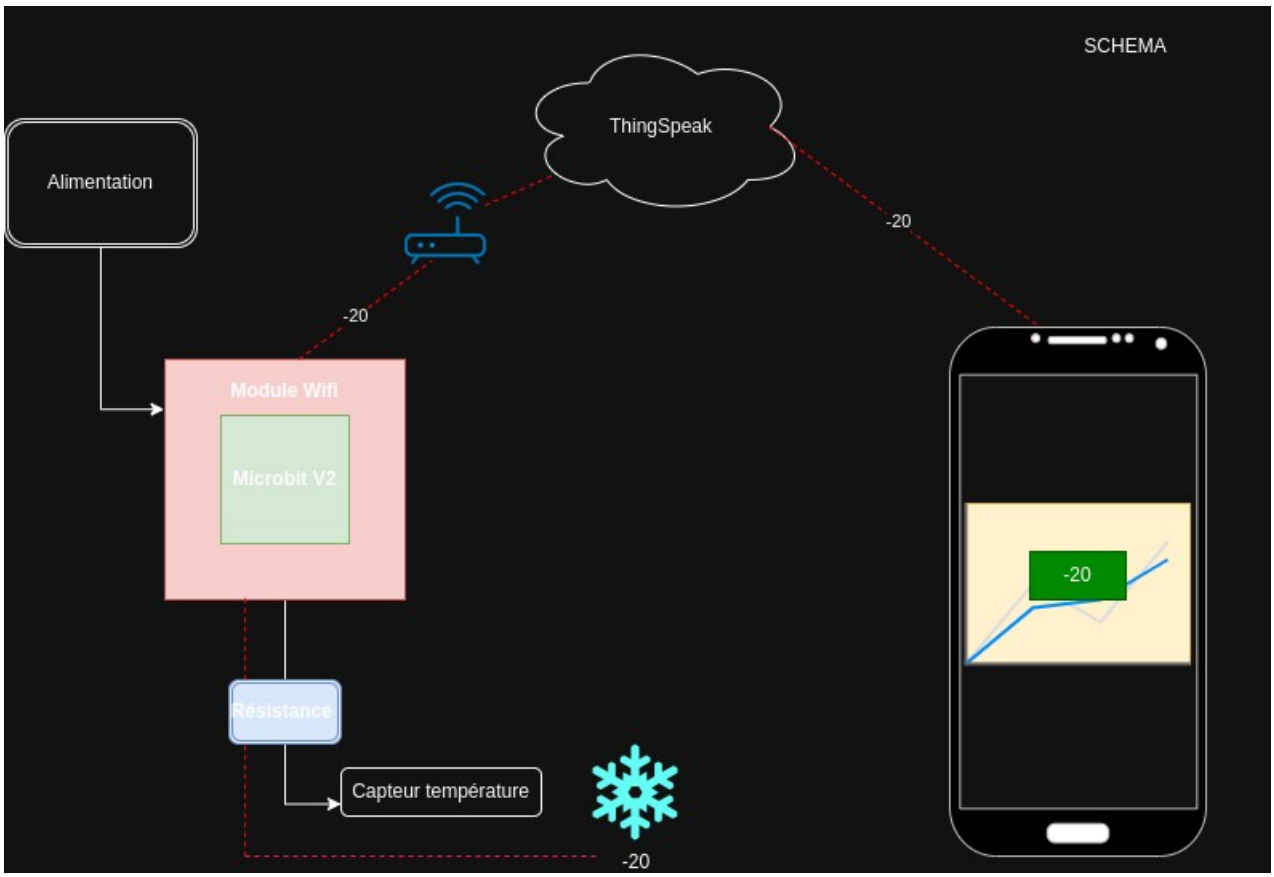
➡ La sonde DS18B20 mesure la température.

➡ La carte Microbit, connectée au module IoT:bit, lit cette température.

➡ Grâce au Wi-Fi intégré au module, la donnée est envoyée vers ThingSpeak, une plateforme en ligne.

➡ Sur le smartphone ou ordinateur, vous pouvez voir la température s’afficher en temps réel, sous forme de graphique.

Schéma de montage



Qu’est-ce que ThingSpeak ?

ThingSpeak est une plateforme open source développée en Ruby. Elle permet à des objets connectés (ce que l’on appelle l’Internet des Objets ou IoT) de communiquer entre eux et d’envoyer des données vers Internet.

À quoi sert ThingSpeak ?

ThingSpeak est un outil en ligne qui vous permet de :

- Collecter des données envoyées par vos capteurs ou objets connectés.
- Visualiser ces données en temps réel sous forme de graphiques.
- Analyser les données directement dans le cloud.
- Déclencher des alertes ou des actions automatiques en fonction des données reçues.

1-Créer un compte ThingSpeak

Pour créer un compte, rendez-vous sur le lien suivant :

https://thingspeak.mathworks.com/login?skipSSOCheck=true

Vous devrez renseigner les informations suivantes :

- Votre adresse e-mail
- Votre pays
- Votre prénom et nom

Suivez ensuite les instructions affichées à l’écran pour finaliser votre inscription.

Créer un “Channel” (canal de données)

Une fois connecté, vous devrez créer un Channel.

Un Channel est un espace de stockage dans lequel vous allez enregistrer les données envoyées par vos capteurs.

Étapes pour créer un Channel : Cliquez sur « New Channel ».

1. Dans le champ Name, indiquez le nom de votre projet.

   Exemple : `Sonde Température`
2. Vous pouvez ajouter une description (facultatif).
3. Dans Field1 (variable), indiquez la première donnée que vous souhaitez enregistrer (par exemple : température).
4. Si vous souhaitez ajouter d’autres capteurs (humidité, GPS, etc.), cochez les cases à droite pour activer Field2, Field3, etc.

   Vous pouvez configurer jusqu’à 8 champs de données.

Qu’est-ce qu’une variable ?

En informatique, les variables sont des éléments qui associent un nom (l’identifiant (température)) à une valeur (température envoyée par notre capteur).

Afficher l’emplacement du capteur (optionnel)

L’option « Show Channel Location » permet d’afficher un point sur une carte représentant l’emplacement géographique de votre capteur.

Cette fonctionnalité est facultative, mais elle peut être utile si vous souhaitez visualiser où se trouvent vos dispositifs.

Finaliser la création du Channel

- Cochez l’option « Show Status » pour activer l’affichage de l’état du canal.
- Cliquez ensuite sur « Save Channel » pour enregistrer votre configuration.

Votre Channel est maintenant créé !

Il est prêt à recevoir et afficher les données envoyées par vos objets connectés.

2. Programmation de la carte Microbit V2

Une fois votre channel ThingSpeak prêt, vous pouvez passer à la programmation de la carte Microbit V2.

Pour cela, vous utiliserez un environnement très visuel, pensé pour les débutants.

Accès à MakeCode

Rendez-vous sur le site officiel de programmation de la Microbit :

https://makecode.microbit.org

> Il n’est pas nécessaire de créer un compte pour utiliser l’éditeur.

Étape 1 : Créer un nouveau projet

1. Cliquez sur « Nouveau projet ».
2. Donnez un nom à votre projet (ex : *Microbit_ThingSpeak*).
3. Cliquez sur Créer.

L’interface de MakeCode

Une fois le projet ouvert, vous verrez une interface divisée en trois zones :

- **Zone 1 : Simulation**

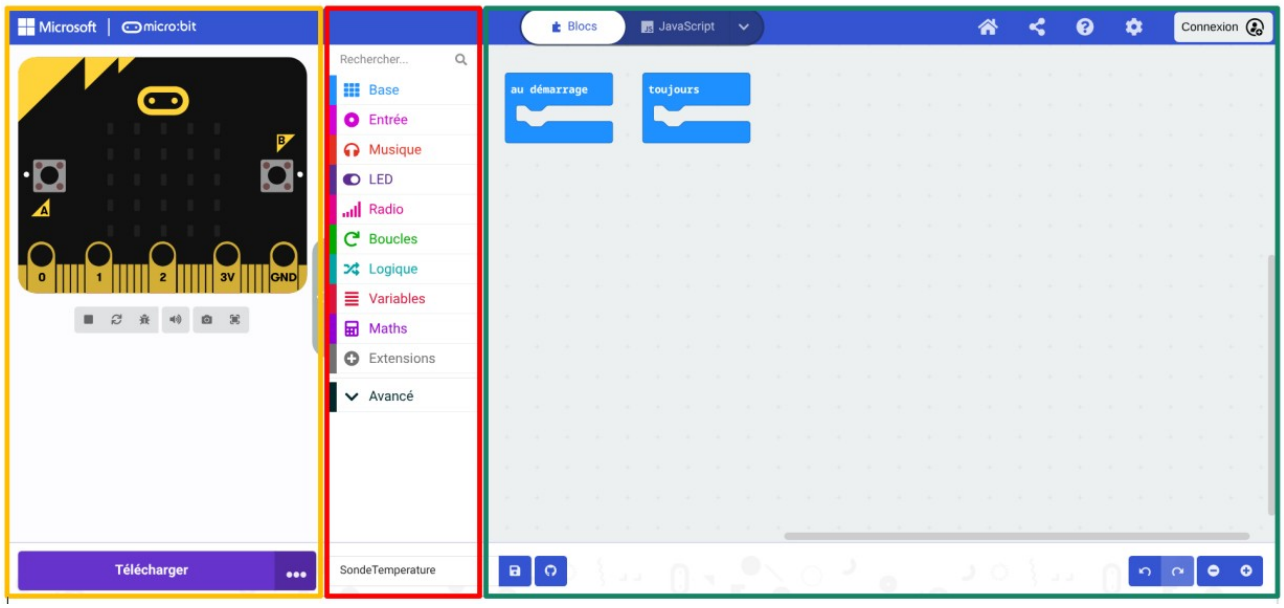
  En haut à gauche, une Microbit virtuelle simule ce que fera votre carte. Elle se met à jour automatiquement selon les blocs que vous assemblez.

- **Zone 2 : Programmation**

  Au centre, vous trouverez des blocs de couleurs représentant différentes actions (entrée, sortie, logique, Wi-Fi, etc.). Il vous suffit de les glisser-déposer pour créer un programme.

- **Zone 3 : Transfert du code**

  À droite, cette section vous permet de télécharger le fichier à transférer sur la carte Microbit. Une fois le programme terminé, vous cliquerez sur Télécharger, puis glisserez le fichier `.hex` sur votre Microbit reconnue comme une clé USB.



Étape 2 : Ajout des extensions nécessaires

Pour que votre Microbit puisse se connecter à Internet, envoyer des données vers ThingSpeak et lire la température via un capteur, il est nécessaire d’ajouter des extensions dans MakeCode.

Pourquoi des extensions ?

Les extensions ajoutent de nouvelles fonctionnalités à l’éditeur. Elles permettent d’utiliser du matériel spécifique, comme le module Wi-Fi ou la sonde DS18B20, sous forme de blocs faciles à manipuler.

Comment ajouter les extensions ?

1. Dans l'éditeur MakeCode, cliquez sur l'onglet « Avancé« dans le menu de gauche.
2. Puis sélectionnez « Extensions« tout en bas de la liste.
3. Dans la barre de recherche qui apparaît, tapez iot, appuyez sur la touche Entrée, puis sélectionnez le premier résultat (lié au module IoT:bit).
4. Répétez l'opération en tapant DS18B20 (la référence du capteur de température), puis sélectionnez le deuxième résultat.

Si tout se passe bien, deux nouveaux menus apparaîtront dans la colonne des blocs :

Un menu IoT pour gérer la connexion Wi-Fi et l'envoi de données.

Un menu DS18B20 pour lire la température via la sonde.

Principe de la programmation par blocs

La programmation sur MakeCode fonctionne comme un jeu de construction. Il suffit de **choisir un bloc** dans le menu (à gauche) et de le glisser–déposer dans l'espace de travail (à droite).

Exemple simple : afficher un texte

Imaginons que vous souhaitiez afficher un message à l'écran de la Microbit :

1. Allez dans le menu « Base«
2. Sélectionnez le bloc « afficher le texte 'Hello' »
3. Glissez ce bloc dans le bloc bleu « au démarrage«

Résultat : à chaque fois que vous allumerez la Microbit, le mot HELLO s'affichera sur l'écran LED.

2A-Programme de la sonde de température et envoi vers ThingSpeak

La création du programme complet se fera en deux grandes parties :

1re partie : Au démarrage

Dans cette section, vous allez :

- Connecter la Microbit au réseau Wi-Fi via le module IoT:bit
- Initialiser la sonde DS18B20
- Afficher un message de confirmation si la connexion réussit

2e partie : Envoi régulier des données

Ici, vous ajouterez un bloc qui :

- Lit la température via la sonde
- Affiche brièvement la valeur sur la Microbit (optionnel)
- Envoie la température vers ThingSpeak via Wi-Fi, en utilisant votre clé API

Ce processus pourra être répété automatiquement toutes les X secondes grâce à une boucle.

Partie 1 : Initialisation au démarrage

Lors du démarrage de la carte Microbit, plusieurs actions doivent être réalisées pour préparer le bon fonctionnement du projet : vérifier que tout est connecté, afficher des messages utiles et tester les composants. Voici comment procéder, étape par étape.

1. Afficher un symbole au démarrage

Pour signaler que la carte est bien alimentée, vous pouvez afficher une icône sur l'écran LED.

- Rendez-vous dans le menu Base.
- Choisissez le bloc « Montrer l'icône« (par exemple une maison , mais vous pouvez aussi choisir un autre symbole, un son ou un texte).
- Glissez ce bloc dans la zone « au démarrage« .

2. Préparer la connexion Wi-Fi

Ensuite, il faut initialiser le module Wi-Fi pour permettre l'envoi des données sur Internet.

- Allez dans la section ESP8266_IoT.
- Prenez le bloc « set ESP8266…« et placez-le juste sous l'icône d'accueil.
- Ajoutez une courte pause pour éviter que le Microbit ne plante :
  - Dans Base, sélectionnez le bloc « pause (ms) 100« .
  - Glissez-le sous le bloc précédent.

3. Connexion au réseau Wi-Fi

Vous allez maintenant vous connecter à votre réseau sans fil.

- Toujours dans ESP8266_IoT, sélectionnez « connect Wifi SSID et Key« .
- Glissez ce bloc sous la pause.
- Remplacez SSID par le nom de votre réseau Wi-Fi, et **Key** par le mot de passe (évitez les caractères spéciaux — un partage de connexion mobile est souvent plus simple).

Pour indiquer que la tentative de connexion est en cours :

- Dans Base, ajoutez « afficher texte 'Wi-Fi…' » juste en dessous.

4. Vérification de la connexion Wi-Fi

Vous allez maintenant vérifier si la connexion a réussi ou échoué.

- Allez dans Logique, sélectionnez le bloc « si… alors… sinon… ».
- Placez-le sous le message 'Wi-Fi…'.
- Dans ESP8266_IoT, prenez le bloc « Wi-Fi connected faux » et insérez-le dans la condition logique. Changez faux en vrai.

Si la connexion est réussie :

- Ajoutez « afficher texte 'Wi-Fi OK' ».
- Puis, allez dans la section ThingSpeak et insérez « connect ThingSpeak » pour établir le lien avec la plateforme.

Sinon (connexion échouée) :

- Ajoutez « afficher texte 'Wi-Fi FAIL' ».
- Puis, ajoutez « montrer l'icône croix » (dans Base) pour signaler l'erreur.

5. Test du capteur de température DS18B20

Enfin, testons si la sonde de température est bien reconnue :

- Utilisez un bloc « si… alors… sinon… » comme précédemment.
- Allez dans DS Temp 2-Wire et choisissez « temperature on P0 ». Glissez cette valeur dans la condition pour vérifier si la température est supérieure à -300 (une valeur arbitraire pour détecter si le capteur répond).
- Si le capteur fonctionne :
  - Affichez le texte « Capteur OK » et un logo (ex.).
- Sinon :
  - Affichez le texte « DS18B20 ? » et un symbole d'erreur (ex. ou un point d'interrogation).

Remarque importante :

Le capteur de température DS18B20 est connecté sur P2, ce qui signifie Pin 2 (ou broche 2) de la carte Microbit.

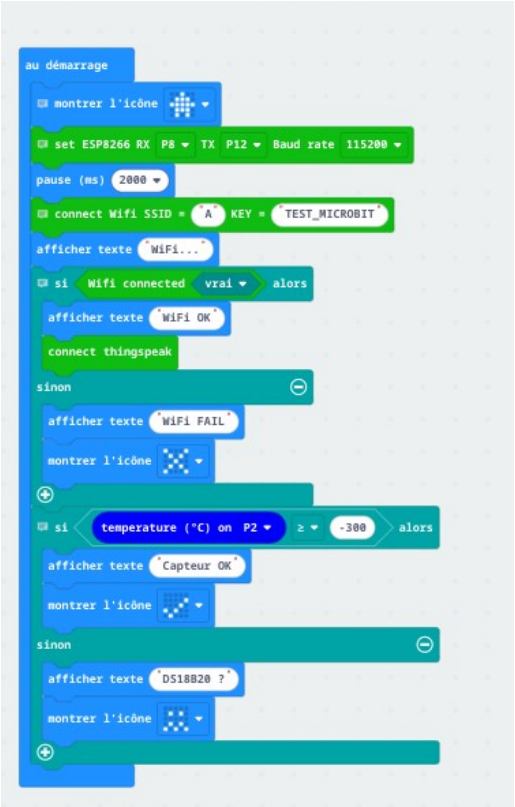
Une pin (ou broche, en français) est un petit connecteur sur la carte qui permet d'envoyer ou de recevoir des signaux (comme des données, de l'électricité ou des commandes). Elle sert à relier la carte à des composants externes comme des capteurs, des LED ou des moteurs.

Grâce au module d'extension IoT:bit, vous disposez de 20 broches numériques : de P1 à P19.

Cependant, attention : P8 et P12 sont réservées pour le fonctionnement du module Wi-Fi.

Évitez donc d'y connecter un autre composant, cela pourrait provoquer un conflit ou empêcher le bon fonctionnement du Wi-Fi.

Voici le code au démarrage :



Partie 2 : Envoi régulier des données vers ThingSpeak

Maintenant que la Microbit est connectée au Wi-Fi et que la sonde DS18B20 fonctionne, nous allons envoyer les données de température vers la plateforme ThingSpeak de manière régulière.

Pourquoi utiliser le bloc “Toujours” ?

Le bloc “toujours” permet d’exécuter en boucle le même ensemble d’instructions.

Dans notre cas, il va servir à mesurer la température en continu et à mettre à jour les données sur ThingSpeak toutes les quelques secondes.

1. Création d’une variable pour la température

Commençons par stocker la valeur de température mesurée dans une variable :

- Dans le menu Variables, cliquez sur “Créer une variable” et nommez-la `Température`.
- Glissez le bloc “définir Température à 0” dans le bloc “toujours”.
- Remplacez le 0 par la température lue depuis le capteur :
  - Allez dans DS Temp 2-Wire, puis prenez le bloc “température on P0”.
  - Glissez-le à la place du 0.
  - Remplacez le P0 par P2

Ce bloc permet de récupérer la température actuelle en degrés Celsius à partir du capteur connecté sur la broche P2.

2. Vérification de la connexion avec ThingSpeak

Avant d'envoyer les données, il faut vérifier que la carte est bien connectée à ThingSpeak :

- Dans Logique, sélectionnez le bloc "si... alors... sinon..." et placez-le sous le bloc de définition de la variable.
- Dans ThingSpeak, utilisez le bloc "ThingSpeak connected" pour remplacer la condition vrai.

3. Envoi des données à ThingSpeak

Si la connexion est bien établie :

1. Glissez le bloc "set data to send to ThingSpeak" dans la partie "alors".
2. Juste en dessous, ajoutez le bloc "upload data to ThingSpeak".

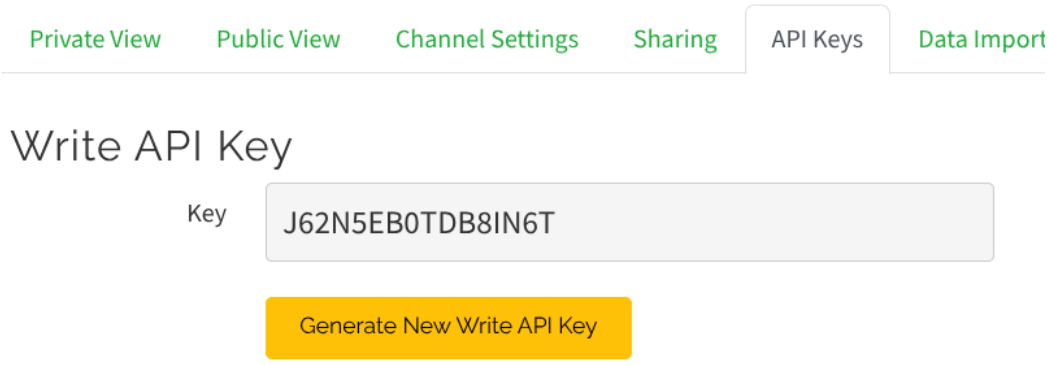
Le champ Field1 correspond au premier champ de votre channel ThingSpeak — ici, il s'agit de la température.

Pour y envoyer la valeur correcte :

- Glissez la variable Température dans la case Field1 =.

Ajout de la clé API

- Dans l'interface de ThingSpeak, allez dans My Channels → API Keys.
- Copiez la clé appelée "Write API Key".
- Collez cette clé dans le bloc "upload data to ThingSpeak > Write API Key", pour que vos données soient bien envoyées vers le bon canal.



4. Que faire en cas d'erreur ?

Si la connexion échoue, vous pouvez afficher un message :

- Dans la section "sinon" de votre condition, insérez un bloc "afficher le texte 'NO'".

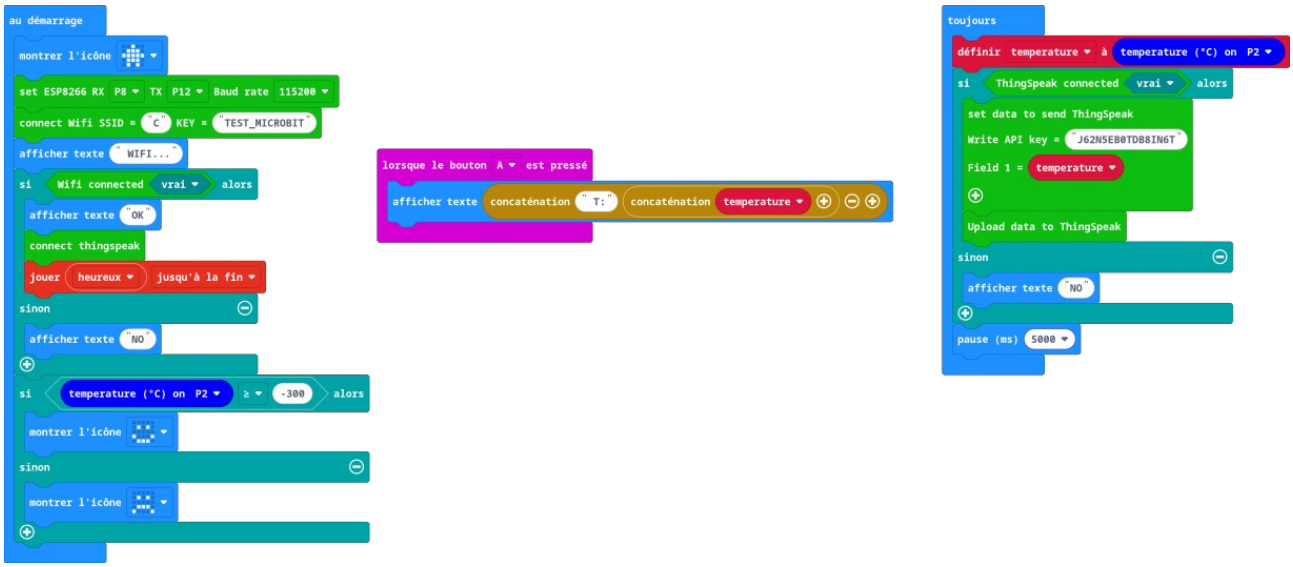
Cela permet de repérer rapidement un problème de connexion.

5. Ajouter une pause entre chaque envoi

Il est essentiel d'ajouter une pause d'au moins 5 secondes après chaque envoi de données. Sans cette pause, le Microbit risque de surchauffer ou de se bloquer.

- Dans Base, utilisez le bloc "pause (ms)".
- Réglez-le à 10 000 ms (10 secondes) pour un bon équilibre entre régularité et stabilité.

Voici le code final :



Que se passe-t-il au démarrage de la Microbit ?

Lorsque vous allumez une carte Microbit pour la toute première fois, un petit jeu interactif se lance automatiquement.

Il s'agit d'un programme de démonstration préinstallé qui permet de découvrir quelques fonctions de base de la carte, comme les boutons, l'affichage LED ou les mouvements.

> Ce mini-jeu est documenté dans le livret fourni avec le kit de démarrage, ou bien via un QR code présent dans la boîte.
> Vous pouvez le tester librement, mais ce programme sera remplacé dès que vous téléchargerez votre propre projet.

Une fois votre programme installé, le jeu ne se relancera plus : la carte exécutera automatiquement le code que vous aurez écrit dans MakeCode.

2B-Télécharger votre programme sur la Microbit

Maintenant que votre projet est prêt, voici comment le transférer sur la carte :

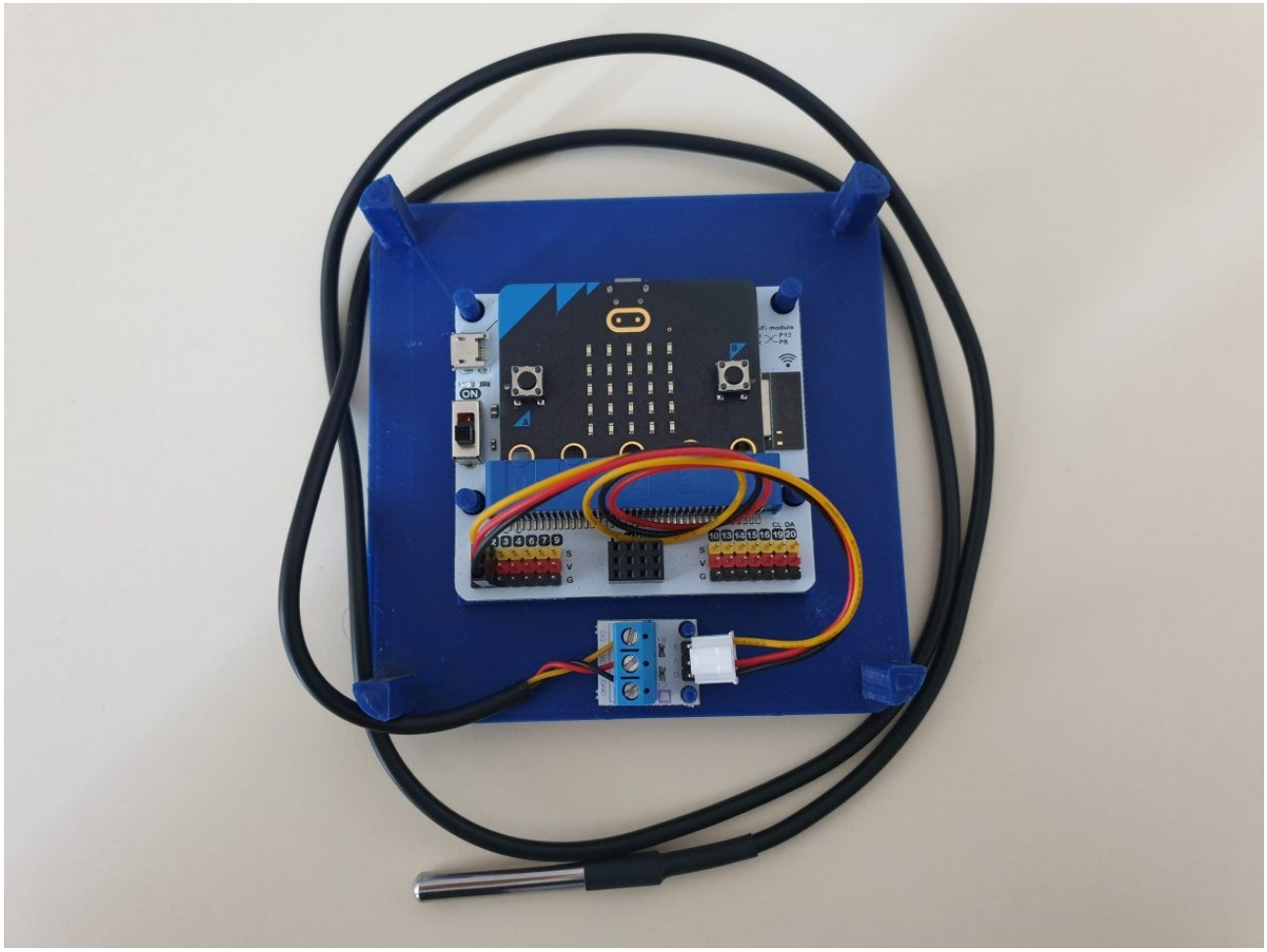
1. Branchez votre Microbit à l'ordinateur via le câble USB.
2. Elle apparaîtra comme un disque externe (comme une clé USB).
3. Dans MakeCode, cliquez sur le bouton "Télécharger".
4. Un fichier `.hex` va être généré.
5. Glissez ce fichier dans le disque Microbit.

   Une LED orange va clignoter : c'est le signe que le transfert est en cours.

> Une fois le fichier copié, la Microbit redémarre et exécute automatiquement votre programme.

3-Schéma de montage du projet

Avant de commencer à mesurer la température et à envoyer des données vers ThingSpeak, il est important de bien comprendre le montage de votre circuit. Voici les explications associées au schéma que vous allez insérer.



Comprendre les connexions : GND, VCC, Signal

Les modules et capteurs comme la sonde DS18B20 possèdent généralement trois fils :

Fonction	Nom de la broche	Couleur du fil (standard)	Rôle
Masse	GND	Noir	Sert de référence électrique (0 Volt)
Tension	VCC / 3.3V ou 5V	Rouge	Fournit l'énergie au capteur (alimentation)
Signal	SIG / Data	Jaune	Transmet les données mesurées (température)

> Il est essentiel de respecter ces couleurs pour éviter les erreurs de branchement. Une inversion pourrait endommager le matériel.

Pourquoi une résistance est nécessaire sur le capteur DS18B20 ?

Le capteur DS18B20 fonctionne avec une liaison de type OneWire (un seul fil pour les données), qui nécessite une résistance de "pull-up" entre la ligne de signal (jaune) et la ligne de tension (rouge).

- Cette résistance a généralement une valeur de 4,7 kΩ.
- Elle permet de stabiliser le signal transmis au microcontrôleur.

> Si vous utilisez une sonde DS18B20 déjà intégrée dans un boîtier étanche pour éducation, cette résistance est souvent déjà incluse. Vérifiez bien les spécifications ou demandez confirmation lors de l'achat.

Dans ce projet, l'alimentation joue un rôle fondamental. Voici les deux connexions à ne pas confondre :

1. Connexion USB à la Microbit

- Sert uniquement à transférer votre code depuis l'ordinateur vers la carte.
- Une fois le programme installé, vous pouvez la débrancher.
- Cette connexion ne suffit pas pour alimenter tout le circuit.

2. Alimentation externe via le module IoT:bit

- Le module IoT:bit (votre carte d'extension) dispose d'un port d'entrée pour une alimentation 5V (via USB ou batterie).
- C'est cette alimentation qui fournira l'énergie à :
  - la carte Microbit,
  - le module Wi-Fi,
  - et la sonde de température.

> Sans cette alimentation externe, le système ne pourra pas fonctionner durablement : ni connexion au Wi-Fi, ni envoi de données, ni lecture du capteur.

Conclusion

Bravo ! Si vous êtes arrivé jusque-là, vous avez franchi toutes les étapes nécessaires pour créer un petit système connecté capable de relever la température et d'envoyer les données dans le cloud grâce à ThingSpeak. Ce projet vous a permis de découvrir les bases de la programmation visuelle, de l'internet des objets (IoT) et de l'électronique appliquée avec la Microbit V2.

Bonus : Visualiser vos données sur smartphone

Pour suivre l'évolution de vos mesures directement depuis votre téléphone, vous pouvez télécharger gratuitement l'application ThingView (disponible sur Android et iOS).

Il suffit d'entrer l'identifiant de votre canal pour consulter vos courbes en temps réel, où que vous soyez.

Ressources complémentaires

Voici quelques liens utiles pour aller plus loin ou améliorer votre projet :

- **Fichiers 3D pour imprimer le boîtier de votre montage (format STL et source)**

  → Free CAD

  → zipModel
- **Documentation officielle Microbit MakeCode**

  → https://makecode.microbit.org/
- **Détails techniques sur le capteur DS18B20**

  → https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf
- **Aide sur la plateforme ThingSpeak**

  → https://fr.mathworks.com/help/thingspeak/

Une question, un souci ?

Ce projet est aussi un atelier éducatif, alors n'hésitez pas à poser vos questions, à partager vos remarques ou même à proposer des idées d'amélioration.