

- 1M - Projet Micro-serre d'intérieure automatisée - Architecture des systèmes informatiques
 - @showdialog
 - @showdialog
- Récolter des données via USB @showhint
 - Programmer le Micro:Bit @showhint
 - Téléchargement du programme et visualisation des données @showhint
 - Interprétation et analyser des données @showhint
 - Exercice #1 : Mesure de l'humidité du sol @showhint
 - Schéma de câblage
 - Programmation du Micro:Bit
 - Mesure de la température et de l'humidité de l'air @showhint
 - Changer la couleur d'une led @showhint
 - Étape 1/4 - Vérifier les connexions
 - Étape 2/4 - Vérifier la connexion Micro:Bit dans le bouclier
 - Étape 3/4 - Vérifier la programmation des ports d'entrées-sorties
 - Étape 4/4 - Vérifier la temporalité

1M - Projet Micro-serre d'intérieure automatisée - Architecture des systèmes informatiques

@showdialog

Ce tutoriel est
proposé par




h@rmnia.education



Ce tutoriel est mis à disposition, sauf exception, selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage dans les Mêmes Conditions 4.0 International.

@showdialog

CONSIGNES IMPORTANTES

-  **Sauvegardez TOUS vos documents** dans votre répertoire de travail de groupe sur **OneDrive**.
-  **Prenez des notes** (sous forme manuscrite ou numérique, peu importe comment, **mais prenez des notes !!!** 😊)
-  **Organisez et planifiez** votre travail de groupe. (Qui ? Fait quoi ? quand ?)
-  **Communiquez** entre vous.
-  **Laissez des traces de **TOUTES vos expérimentations** en prenant des **notes**, des **photos** (avec votre téléphone portable) ou des **captures d'écran**.
-  Mettez à jour le journal de bord à **CHAQUE séance**. ⚠
-  Si vous êtes **perdu** servez-vous des **guides** et des **ressources** qui sont à votre disposition dans le **répertoire du projet** sur **OneDrive**.

Récolter des données via USB

@showhint

Le programme envoie chaque seconde la **température** et la **luminosité** sur le port série (**USB**) afin de pouvoir les **visualiser** et les **sauvegarder** sur le PC.

Remarques

- La **température** et la **luminosité** proviennent de deux **variables prédéfinies** dans le Micro:Bit et contiennent les **valeurs mesurées** par les **capteurs internes** correspondants du Micro:Bit (Sonde de température du CPU, matrice de leds)

Programmer le Micro:Bit @showhint

Pour transmettre les données via le port série **USB**, il faut utiliser la **fonction** `||Communication Série:série écrire valeur||` dans la catégorie **Communication Série**. Ceci va définir le nom et la valeur mesurée. Nous trouverons les variables prédéfinies pour la température `||Entrée:température (° C)||` et pour la luminosité `||Entrée:niveau d'intensité lumineuse||` dans la catégorie **Entrée**.

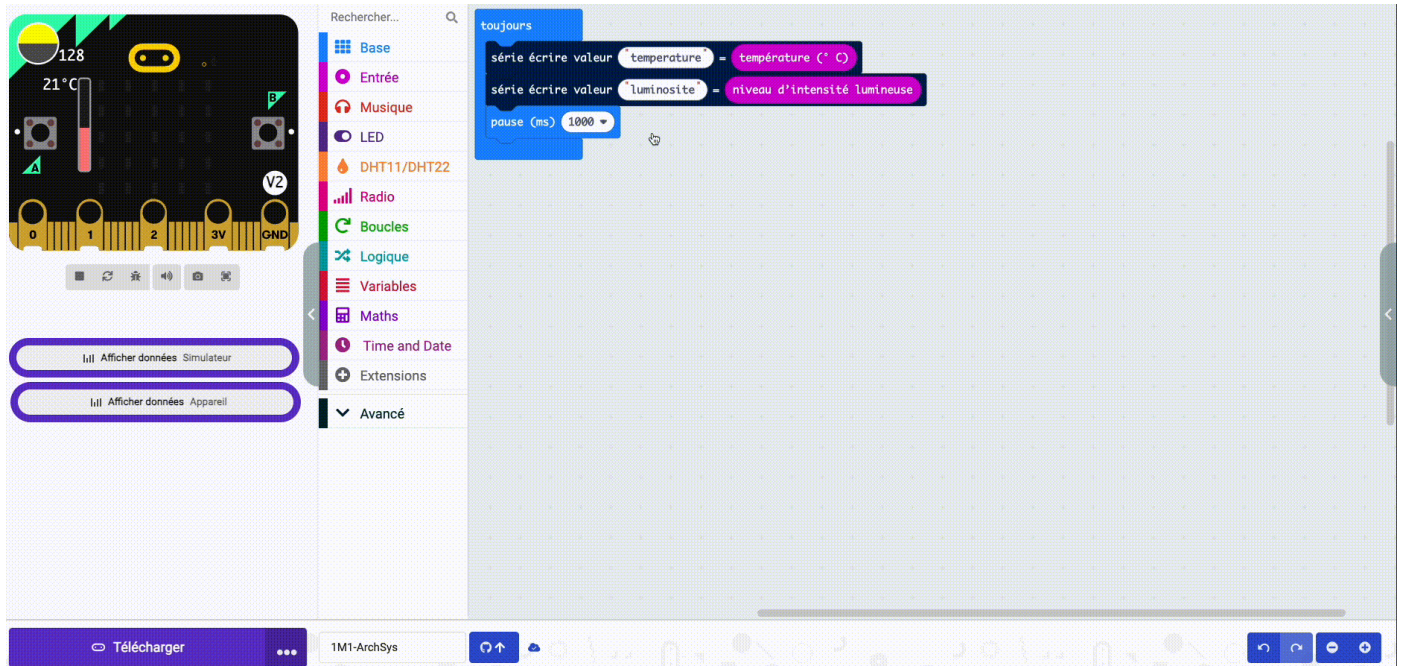
```
basic.forever(function () {
    serial.writeValue("temperature", input.temperature())
    serial.writeValue("luminosite", input.lightLevel())
    basic.pause(1000)
})
```

Téléchargement du programme et visualisation des données @showhint

- Il est important de télécharger le programme à chaque **modification**.

Remarque

- Lors du premier téléchargement, "Makecode" demande de paier le Micro:Bit si cela n'a jamais été fait auparavant.
- Lorsque le téléchargement est terminé, un nouveau bouton "Afficher données Appareil" doit apparaître sur la partie gauche de l'écran.



Astuces : Touchez le CPU pour faire varier la température, et éclairez avec votre smartphone la matrice de led du Micro:Bit.

IMPORTANT : Sauvegardez **toutes vos** données récoltées à chaque fois que vous faites des essais, ceci vous aidera à mieux comprendre ce qui se passe et apporter des corrections si besoin.

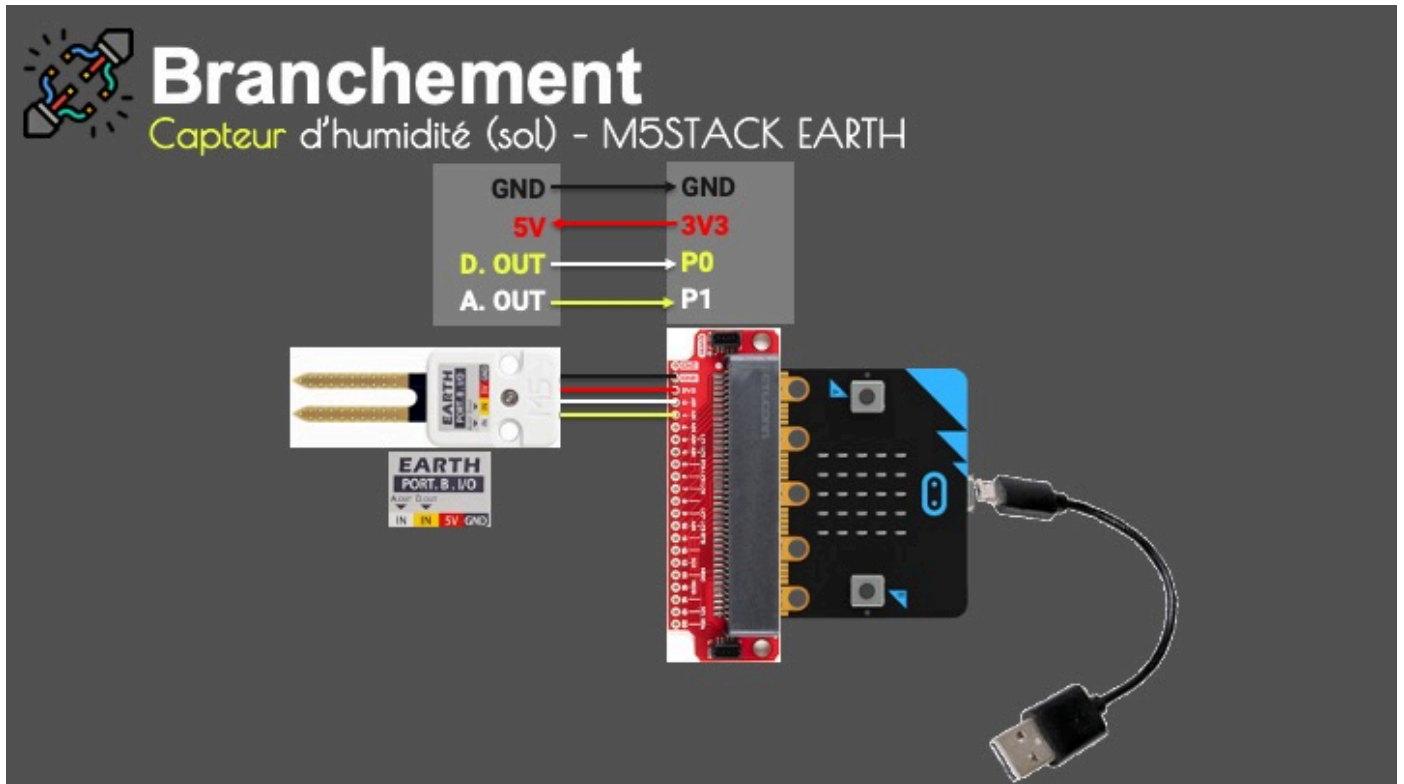
Vous pouvez utiliser une application de traitement de données tel que Excel, Numbers ou Google Sheet pour visualiser et/ou traiter vos données de manière plus approfondies.

Interprétation et analyser des données @showhint

Il est **nécessaire** de valider les données récoltées, et supprimer celles qui sont inutiles ou incorrectes.

Exercice #1 : Mesure de l'humidité du sol @showhint

Schéma de câblage

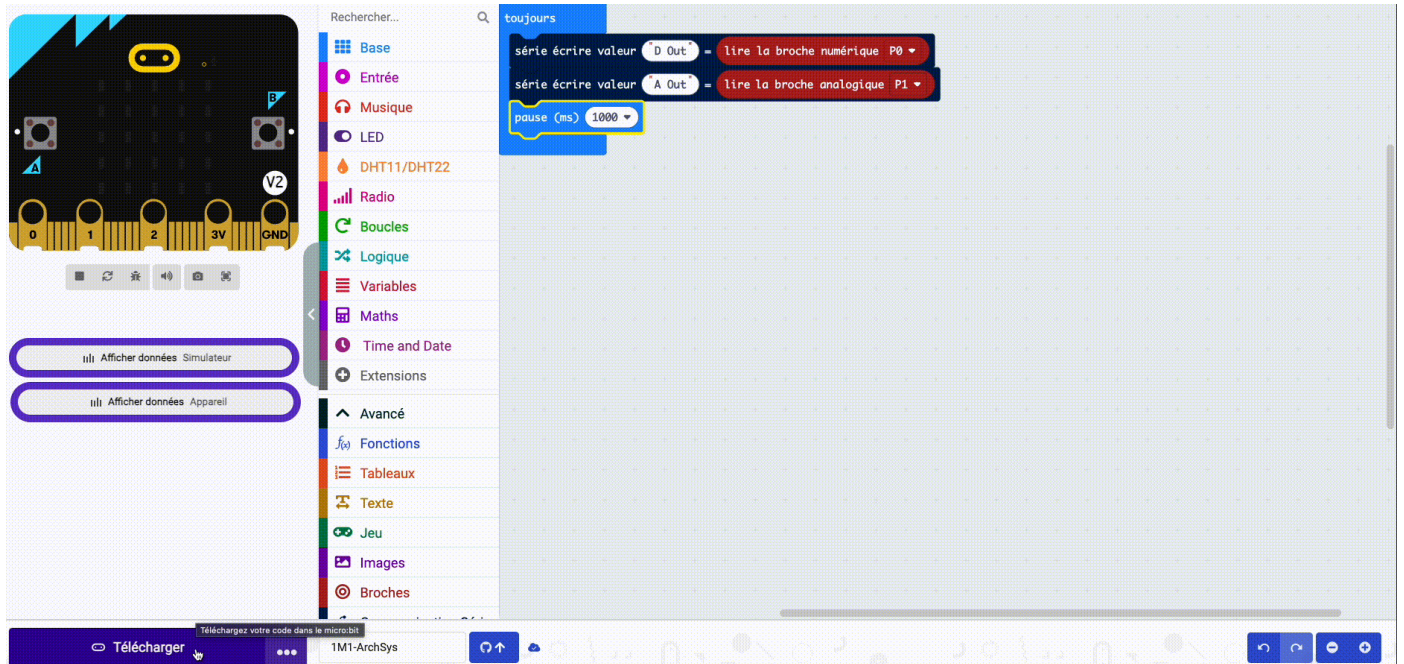


Programmation du Micro:Bit

Le programme utilise la **fonction** `||Communication Série:série écrire valeur||` dans la catégorie **Communication Série** pour envoyer chaque seconde D Out et A Out sur le port série (**USB**).

- D Out : Digital Out -> Valeur numérique (0 ou 1)
- A Out : Analogic Out - > Valeur analogique (0-1023)

```
basic.forever(function () {  
    serial.writeValue("D Out", pins.digitalReadPin(DigitalPin.P0))  
    serial.writeValue("A Out", pins.analogReadPin(AnalogPin.P1))  
    basic.pause(1000)  
})
```

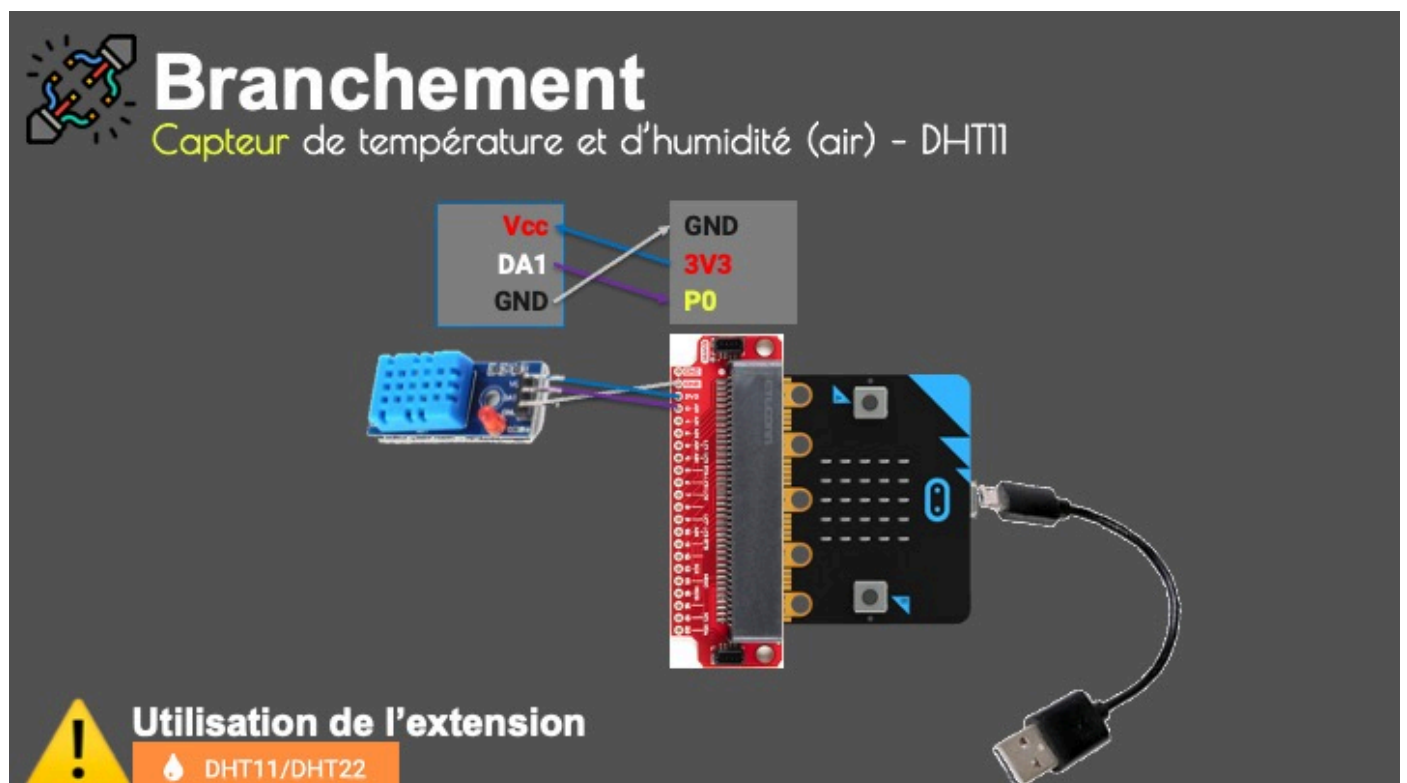


Rapportez votre rapport d'atelier :

1. Les problèmes que vous avez rencontrés lors de vos mesures ?
2. Comment vous les expliqués ?

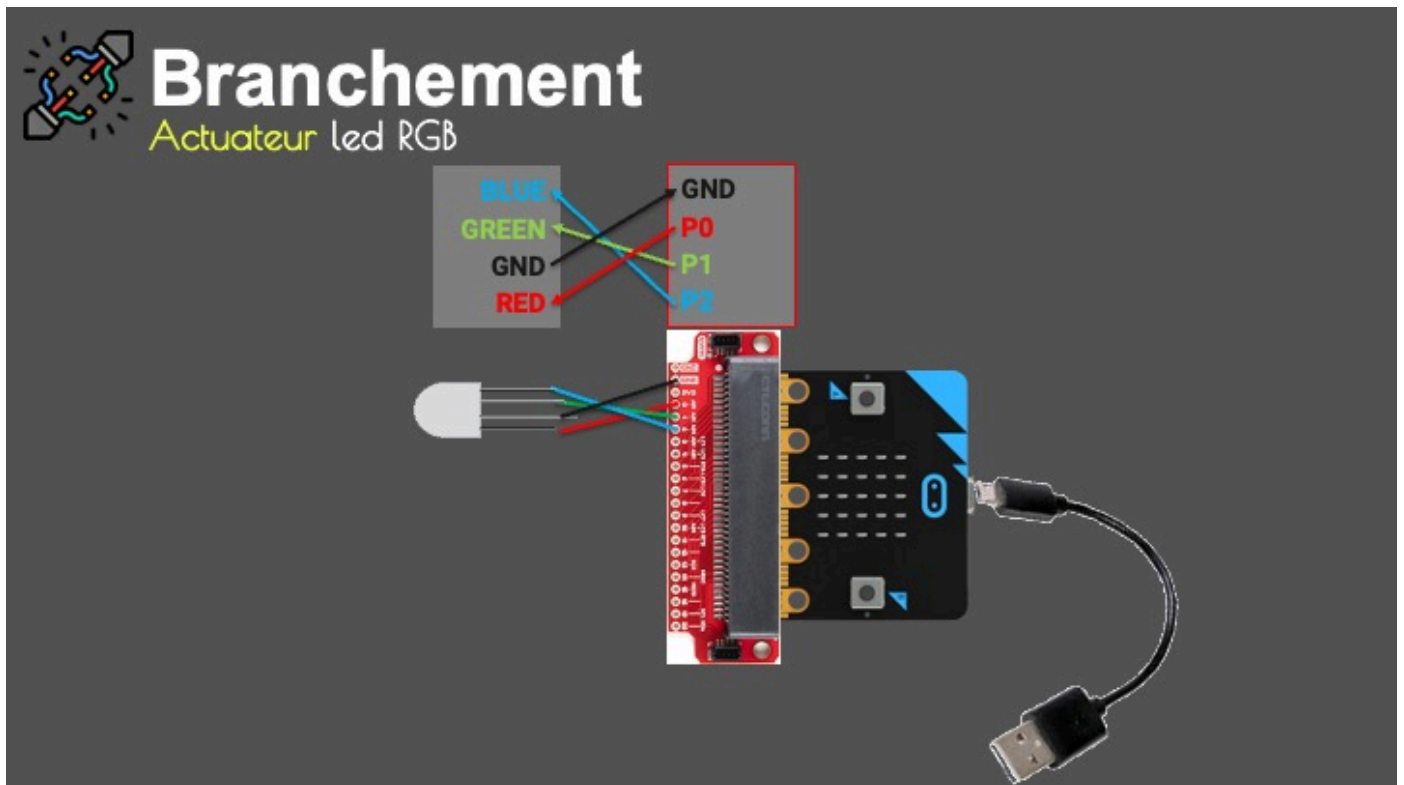
Mesure de la température et de l'humidité de l'air @showhint

Capteur - DHT11



Changer la couleur d'une led @showhint

Actuateur - Led RGB



```
basic.forever(function () {  
  for (let Rouge = 0; Rouge <= 1023; Rouge++) {  
    for (let Vert = 0; Vert <= 1023; Vert++) {  
      for (let Bleu = 0; Bleu <= 1023; Bleu++) {  
        pins.analogWritePin(AnalogPin.P0, Rouge)  
        pins.analogWritePin(AnalogPin.P1, Vert)  
        pins.analogWritePin(AnalogPin.P2, Bleu)  
      }  
    }  
  }  
})
```

Astuces & conseils

- Si vous ne parvenez pas à faire l'exercice, vous pouvez regarder la [vidéo YouTube](#) pour vous aider.
- Vous pouvez sauvegarder votre programme pour chaque étape en copiant le contenu du fichier main.blocks comme le montre l'exemple ci-dessous.

GUIDE DE DÉPANNAGE - (TROUBLESHOOTING)

- [Définition : Troubleshooting](#)

Étape 1/4 - Vérifier les connexions

Vérifiez méticuleusement les connexions selon le schéma de câblage.

Conseils Par moment, il est préférable de tout débrancher et tout rebranché en respectant précisément le **schéma de câblage**.

Étape 2/4 - Vérifier la connexion Micro:Bit dans le bouclier

Assurez-vous que le Micro:Bit soit inséré correctement (et dans le bon sens) dans le bouclier. (shield) Contrôlez que le câble Micro-USB soit correctement branché.

- Si le câble est mal branché, le transfert du programme **ET** la récolte de données ne pourront pas fonctionner, car la liaison entre le Micro:Bit et le PC ne sera pas établie.

Étape 3/4 - Vérifier la programmation des ports d'entrées-sorties

Les données sont soient :

1. Récoltées en **entrée** depuis un **capteur**. (**Input**)
2. Envoyées en **sortie** sur un **actuateur**. (**output**).

Le Micro:Bit dispose de 3 ports standards (**P0, P1, P2**) qui sont prévus pour **récolter** ou **envoyer** des données. **Il est nécessaire** d'indiquer pour chaque ports si ils sont utilisés en **entrée** (**capteur**, ex. capteur de température) ou en **sortie** (**actuateur**, ex. led) et si les données sont **analogiques** (compris entre 0-1023) ou **numériques**. (0 ou 1)

Étape 4/4 - Vérifier la temporalité

Lorsque le Micro:Bit envoie ou reçoit des données, il peut le faire de manière beaucoup trop rapide pour le capteur ou l'actuateur, raison pour laquelle il faut

insérer une pause `||basic:pause(temps)||` dans la boucle qui effectue la mesure.