

EXEMPLE COMPLET

# COMMENT RENDRE VISIBLE L'INVISIBLE ?

Auteur : Stéphane Vassort, [stephane.vassor@lets-steam.eu](mailto:stephane.vassor@lets-steam.eu)

---



# Étape 1 - Présenter le projet dans son ensemble



Nous vous invitons à travers ce modèle à faire preuve de créativité tout en bénéficiant d'un soutien technique afin de concevoir un projet unique et inclusif ! Vous êtes libre de développer votre propre solution ou de vous inspirer de ressources additionnelles.

## Décrire votre projet



### Nommez votre projet : Comment rendre visible l'invisible ?

Brève introduction de ce qu'est votre projet, le problème abordé derrière, les objectifs pédagogiques.

Ce projet consiste à développer des terrariums communicants pour les grenouilles. Il vise à sensibiliser aux enjeux climatiques par la découverte de l'environnement des dendrobates. Nous proposons de contrôler la température dans un terrarium afin de s'assurer que les conditions idéales (entre 21 et 26°C) sont présentes.

## Réfléchir à l'équité et à l'inclusion



### ASPIRATIONS ET MOTIVATIONS

Comment vous sentez-vous lorsque vous faites des activités STEM ? Qu'est-ce qui vous motive dans les STEM ? Qu'est-ce qui motive vos élèves ? Vos élèves sont-ils tous motivés par la même chose ?

- Trouver des possibilités d'appliquer les connaissances et les compétences dans des projets concrets.
- La créativité comme moyen de promouvoir l'inclusion
- Offrir aux élèves différentes possibilités de développer leurs propres projets pertinents.
- Utilisation de la technologie numérique à des fins ludiques ou des environnements ludiques
- Se sentir motivé·e par la possibilité de créer de nouveaux objets.

### PROBLÈMES ET OBSTACLES

Qu'est-ce qui préoccupe vos élèves ? Quelles sont leurs frustrations ? Existe-t-il des différences qui les désavantageant par rapport aux autres élèves ? Et concernant le numérique dans les activités STEM ?

- Ressources financières pour accéder à une formation sur l'apprentissage assisté par la technologie.
- Fracture numérique générale
- Objectifs différents selon le sexe
- Difficultés potentielles sur le matériel technologique

### MOTS CLEFS

Indiquez 3 mots clefs ou plus qui décrivent la réalité de vos élèves en ce qui concerne les activités STEAM.

- NOUVEAU
- EXCITENT
- EFFRAYANT

## Étape 2 - Collecter des données grâce à la carte et à ses capteurs intégrés 1/2



À ce stade, vous devez trouver une solution de programmation pour collecter vos données, identifier les capteurs à utiliser et comment les programmer sur MakeCode pour que la plateforme communique avec votre carte.

### ORIENTATION



Définissez quel est le problème à résoudre, quelles sont les données à collecter, quels sont les objectifs d'apprentissage derrière le sujet de programmation ?

**Contexte :** Afin de reproduire l'environnement naturel des grenouilles et d'assurer leur survie, différents paramètres de leur milieu de vie doivent être pris en compte. Quelles informations devons-nous connaître pour leur offrir le milieu de vie le plus approprié ?

**Objectifs d'apprentissage :** Identifier les capteurs utiles et la procédure pour les mettre en œuvre avec une carte programmable.

### CONCEPTUALISATION



Formulez une hypothèse pour répondre au problème donné concernant la collecte de données.

Comme le principal paramètre à contrôler pour assurer la survie de la grenouille est la **température**, et qu'elle doit être comprise **entre 21 et 26 °C**, la solution qui semble la plus simple est **d'utiliser le capteur de température intégré à la carte programme STM32**.

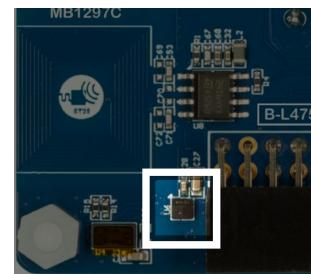
### INVESTIGATION



Décrivez les étapes nécessaires à la collecte des données qui seront utiles à votre projet.

Cette étape peut être réalisée grâce à la fiche d'activité **#R1AS11 - Fabriquer un thermomètre ... très, très visible**. Dans cette activité, nous apprenons à quel point il est facile de lire le capteur de température de la carte et d'afficher sa valeur.

Ce capteur de température est situé à côté du capteur "temps de vol" (*time of flight*) sur la droite, il permet de mettre en œuvre des activités liées à la surveillance de la chaleur ou à l'approche de notions météorologiques. Dans notre cas, il va permettre de surveiller la température à l'intérieur du vivarium.



## Étape 2 - Collecter des données grâce à la carte et à ses capteurs intégrés 2/2



Il est possible d'interroger le capteur de température intégré à la carte avec le logiciel de programmation par blocs disponible dans MakeCode dans la liste de blocs "INPUT".

### Mesurer la température

Pour être pleinement fonctionnel, il est nécessaire que le capteur de température puisse fonctionner au moins jusqu'à 50°C. Afin de vérifier que le capteur sera opérationnel. Après vérification, l'indicateur de température de la carte STM32 indique une plage mesurable par le capteur entre -5°C et 50°C. Ainsi, le choix d'utiliser le capteur intégré semble tout à fait satisfaisant et suffisant.

Fournissez des captures d'écran de la plateforme MakeCode et de votre carte.

The screenshot shows the Microsoft MakeCode interface for the micro:bit. On the left, there's a schematic diagram of the micro:bit board. In the center, a script is being edited:

```
en cas d'appui sur button A0 déclenche clic
button A0 est appuyé
Temperature
on temperature hot at 15 °C
temperature in °C
Humidity
on humidity wet at 50 percent
relative humidity in percent
Pressure
```

A yellow callout box highlights the "Temperature" section of the script with the text "Utiliser les blocs de température".

## DEBRIEFING



Identifiez les connaissances mobilisées pendant cette phase, pensez à votre classe et identifiez les apprentissages possibles, ajoutez les questions de référence qui peuvent se présenter.

Grâce à cette étape, nous avons pu définir que pour obtenir des informations sur l'environnement extérieur, une carte programmable peut utiliser des capteurs.

Si nous voulons programmer la carte STM32, il existe des fonctions pour dialoguer avec le capteur de température intégré et ainsi obtenir la température en degrés Celsius.

Un capteur n'a pas une plage de mesure infinie, il est donc important de vérifier l'adéquation entre sa plage de mesure possible et les mesures attendues.

# Étape 3 - Afficher les données pour obtenir les informations nécessaires 1/2



A ce stade, vous devez trouver une solution de programmation pour afficher vos données, permettant, maintenant que vous avez demandé à un capteur d'obtenir des informations, de faire connaître ces informations à l'utilisateur.

## ORIENTATION



Définissez quel est le défi de l'affichage des données dont vous avez besoin ? Pour vous ? Pour votre classe ? Pour l'utilisateur ?

**Contexte :** Nous avons pu voir dans la partie précédente comment demander à un capteur d'obtenir des informations. Il serait utile maintenant de pouvoir faire connaître ces informations à l'utilisateur.

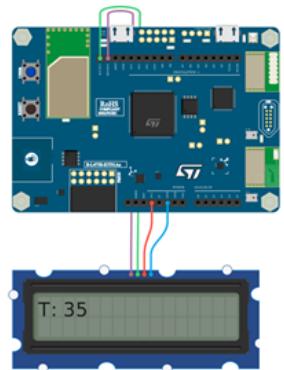
**Objectifs pédagogiques :** Identifier un actionneur et le contrôler afin de pouvoir délivrer une information.

## CONCEPTUALISATION



Formulez une hypothèse pour répondre au problème donné concernant l'affichage des données.

Afin d'informer l'utilisateur de la température mesurée, la première solution qui vient à l'esprit est d'utiliser **l'afficheur de texte LCD externe**.

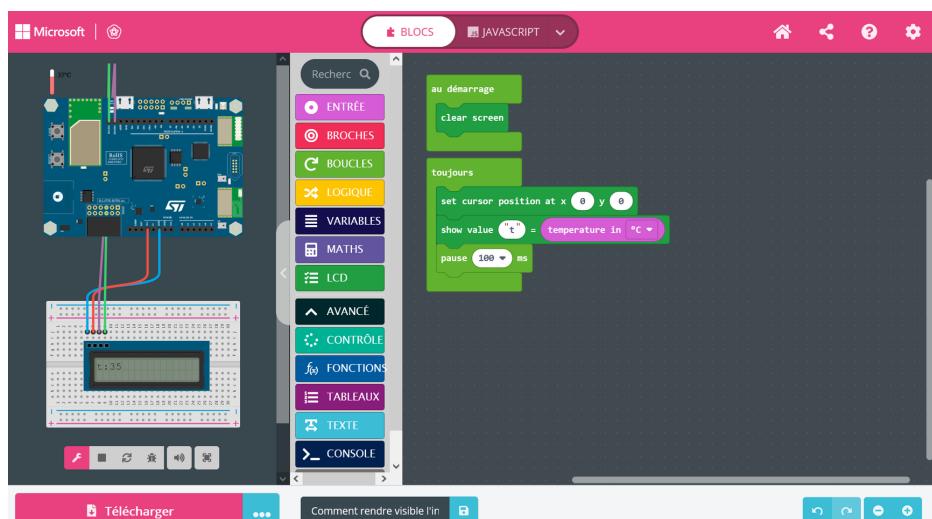


## INVESTIGATION



Décrivez les étapes à suivre pour afficher et montrer les données qui seront nécessaires à votre projet.

Cette étape peut être mise en œuvre grâce à la fiche d'activité **#R1AS10 - Affichage de texte avec un écran OLED**. Dans la documentation de la carte STM32, nous pouvons y voir les fonctions utilisées pour afficher les données sur l'écran LCD : "set cursor position at x: y:" et "show value".



## Step 3 - Display the data to get the needed information - 2/2



### Tester le programme

Afin de vérifier si cela fonctionne, testez un premier programme qui effectue la tâche suivante :

- Nettoyez l'écran LCD,
- Identifiez la position du curseur (sur x=0 et y=0),

Écrivez le mot "**Temp**", affichez la valeur mesurée par le capteur de température et écrivez le mot "C" (pour indiquer que la température est mesurée sur l'échelle Celsius).

Afin de pouvoir appeler ce programme (séquence de blocs) depuis un autre programme, je remplace la boucle "**forever**" par le bloc fonction. La fonction s'appelle "**PrintTemp**".

Fournissez des captures d'écran de la plateforme MakeCode et de votre carte.

The image shows two side-by-side screenshots of the MakeCode editor. Both screenshots show a sequence of blocks:

- A green "fonction" block labeled "PrintTemp".
- A green "clear screen" block.
- A green "set cursor position" block with "at x" set to 0 and "y" set to 0.
- A green "show string" block with the value "Temp".
- A green "show value" block with "t" = "temperature in °C".
- A green "show string" block with the value "C".

The right screenshot shows the same sequence of blocks, but the "show value" block has been converted into a function call, indicating it is part of the "PrintTemp" function definition.

### DEBRIEFING



Identifiez les connaissances mobilisées pendant cette phase, pensez à votre classe et identifiez les apprentissages possibles, ajoutez les questions de références qui peuvent se présenter.

Grâce à cette étape, nous avons pu connecter l'écran LCD à la carte STM32.

**Note sur les types de données :** La donnée est fournie mais la température rendue par le capteur est un entier et la lettre C, pour l'unité, est une chaîne de caractères, c'est pourquoi nous avons utilisé deux blocs différents : "**show number**" et "**show string**". Afin de structurer un programme, il est possible de définir une fonction pour chaque tâche à effectuer.

# Étape 4 - Analyser les données et en tirer des enseignements 1/2



Maintenant que nous sommes capables d'afficher les données instantanément, nous avons besoin de les analyser pour effectuer le suivi de nos informations (suivi de la température, des alertes, du mouvement, de la fréquence...). Cette étape est faite pour permettre cette analyse sur l'éditeur.

## ORIENTATION



Définissez ce qui constitue le défi de cette étape en fonction de votre projet. Quel est votre défi pour analyser et extraire les informations pertinentes appliquées à votre contexte ?

**Contexte :** Nous sommes capables d'afficher des données instantanément. Pour pouvoir analyser les variations des conditions climatiques et identifier quand le niveau de température devient critique pour nos grenouilles et la fréquence de ces alertes, il serait utile de pouvoir effectuer ce suivi sur une longue période.

**Objectifs d'apprentissage :** Analyser des données et extraire des informations pertinentes.

## CONCEPTUALISATION



Formulez une hypothèse pour répondre au problème donné concernant l'analyse des données.

Afin de pouvoir analyser les données du capteur de température sur une longue période, l'utilisation d'un tableau serait une solution simple. Pour cela, il est nécessaire de pouvoir récupérer les données de la carte programmable. La solution que nous allons mettre en œuvre sera d'écrire via le port série les données au format CSV (comma-separated value), exploitable dans un tableau.

## INVESTIGATION



Décrivez les étapes nécessaires à l'analyse et au suivi des données qui seront nécessaires à votre projet.

Vous pouvez utiliser les ressources suivantes pour commencer : [https://fr.wikipedia.org/wiki/Comma-separated\\_values](https://fr.wikipedia.org/wiki/Comma-separated_values). Selon la documentation, un fichier CSV est un format texte ouvert représentant des données tabulaires sous forme de valeurs séparées par des virgules. Les entêtes du tableau sont sur la première ligne, et les données sont ensuite insérées ligne par ligne.

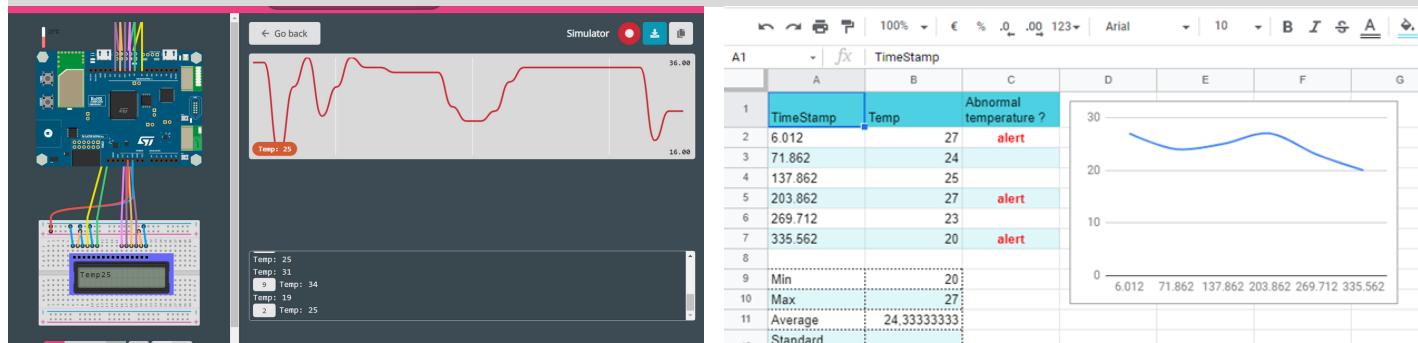
# Étape 4 - Analyser les données et en tirer des enseignements 2/2



## Tester le programme

Afin de récupérer les données pour les analyser, écrire sur la console la température fournie par le capteur embarqué toutes les minutes. Il suffira d'afficher le graphique et de télécharger les données sous forme de fichier CSV. Ce document peut ensuite être ouvert avec un tableur. Il est ainsi possible d'obtenir la moyenne, la température minimale, maximale ou encore l'écart type.

Fournissez des captures d'écran de la plateforme MakeCode et de votre carte.



## DEBRIEFING



Identifiez les connaissances mobilisées pendant cette phase, pensez à votre classe et identifiez les apprentissages possibles, ajoutez les questions de référence qui peuvent se présenter.

Grâce à cette étape, nous avons pu découvrir qu'une carte programmable pouvait également envoyer des informations via une console. Cette fonctionnalité permet d'envoyer des informations plus rapidement qu'en utilisant l'écran intégré mais nécessite un ordinateur connecté.

## Format CSV

La console nous a permis d'envoyer un fichier texte au format CSV qui peut ensuite être ouvert par un tableur pour analyser les données. A partir de ces données, un tableur peut facilement dessiner des représentations graphiques ou effectuer des calculs statistiques.

# Étape 5 - Utiliser les données pour avertir les utilisateurs en cas d'urgence 1/3



Maintenant que nous sommes capables de collecter, d'afficher et de contrôler des données, nous pouvons créer une solution pour utiliser ces données dans la vie réelle dans un but concret. Cette étape supplémentaire du projet permet de fournir un cas d'application réelle à notre activité.

## ORIENTATION



Définissez l'enjeu de cette étape en fonction de votre projet. Quel est l'objectif concret pour l'utilisateur ?

**Contexte :** Nous sommes maintenant capables de mesurer et d'analyser les données issues des capteurs. Il serait utile de pouvoir informer l'utilisateur de la température dans le vivarium et en cas de détection d'une température trop élevée de pouvoir la diminuer.

**Objectifs d'apprentissage :** Identifier une condition et implémenter un bloc conditionnel.

## CONCEPTUALISATION



Formulez une hypothèse pour répondre au problème donné concernant cette étape supplémentaire.

Il y a deux tâches à accomplir ici :

- Informer l'utilisateur de la température dans le vivarium de la manière la plus visible possible, par exemple en changeant la couleur de l'écran LCD ;
- Ouvrir une fenêtre lorsque la température devient trop élevée.

Afin d'identifier automatiquement dans quelle plage de température se trouve l'état actuel, et d'afficher la couleur correspondante de l'écran LCD à l'utilisateur, nous allons utiliser un bloc conditionnel "IF".



## INVESTIGATION



Décrivez les étapes à suivre pour afficher et montrer les données qui seront nécessaires à votre projet.

### Tester le programme

Afin de prévenir l'utilisateur, le programme va changer la couleur de l'écran LCD en fonction de la température de la manière suivante :

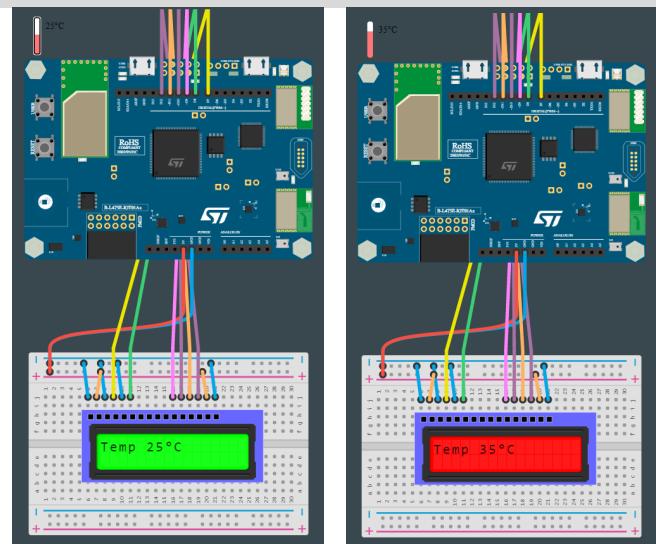
- **-5..21 °C** - lumière **rouge**
- **21 ... 26 °C** - lumière **verte**
- **26..50 °C** - lumière **rouge**

Fournissez des captures d'écran de la plateforme MakeCode et de votre carte.

```

forever
  clear screen
  set cursor position at x 0 y 0
  show string "Temp "
  show number temperature in °C
  show string "°C"
  if temperature in °C < 21 or temperature in °C > 26 then
    set backlight color red
  else
    set backlight color green
  console log value "Temp" = temperature in °C
  pause 100 ms

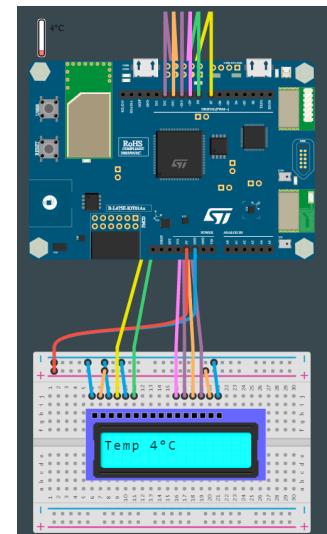
```



Une lumière supplémentaire peut être ajoutée afin d'informer plus précisément l'utilisateur de la température dans le vivarium, et allumer la lumière idoine :

- si la température est **inférieure à 21 °C** - lumière **bleue**,
- si la température est comprise **entre 21 et 26 °C** - lumière **verte**
- si la température est **supérieure à 26 °C** - lumière **rouge**.

Pour cela, j'ai utilisé le bloc conditionnel "**If .. then.. else**". Dans chaque cas, appeler la fonction "**AfficherTemperature**" (créée lors de la première étape de ce projet) pour imprimer la température actuelle sur l'écran LCD.



Afin de pouvoir ouvrir la fenêtre, connecter le moteur pas à pas à la carte STM32 tel que présenté dans la fiche **#R1AS13**. Ensuite, créer la fonction "**EmergencyVentilation**" qui doit être appelée dans le cas où la température est supérieure à 26 °C.



## DEBRIEFING



Identifiez les connaissances mobilisées pendant cette phase, pensez à votre classe et identifiez les apprentissages possibles, ajoutez les questions de références qui peuvent se présenter.

### Boucle conditionnelle

Grâce à cette étape, nous avons pu découvrir ce qu'est une instruction conditionnelle et ses versions : courte "If .. then" et longue "If .. then.. else". Il s'agit d'une structure algorithmique qui n'exécutera une action que si une condition est vérifiée. Dans notre cas, un écran LCD allumera les lumières bleue, verte ou rouge si la température est respectivement dans l'une des plages -5...20 C°, 21...25 C° ou 26...50 C°.

### Ajout de nouveaux dispositifs

Afin de bénéficier de nouvelles fonctionnalités, il est possible d'ajouter des extensions apportant des fonctions supplémentaires. Ici, nous avons ajouté le moteur pas à pas pour activer la ventilation si la température dépasse 26 C°.