

# RESSOURCES - PROGRAMMATION - FICHE D'ACTIVITÉ 11

# FABRIQUER UN THERMOMÈTRE

... TRÈS, TRÈS LISIBLE

#R1AS11



**Disponible sur**



**Prérequis**

- R1AS04 - Découverte des capteurs de lumière

**Matériel**

- 1 carte programmable "**STM32 IoT Node Board**"
- 1 câble USB Micro-B
- 1 écran texte LCD Grove I2C
- 1 câble de liaison Grove



**De quoi parle-t-on ?**

Dans cette activité, nous allons apprendre à quel point il est facile de lire le capteur de température de la carte et d'afficher sa valeur.

**Durée**

20 minutes

**Niveau de difficulté**

Intermédiaire

## OBJECTIFS D'APPRENTISSAGE

- Lire le capteur de température
- Utiliser un écran d'affichage LCD





La température est une grandeur physique qui exprime le chaud et le froid. C'est la manifestation de l'énergie thermique, présente dans toute matière, qui est à l'origine de l'apparition de la chaleur. En les mettant en contact, les chocs entre particules font que cette énergie cinétique microscopique se transmet d'un corps à l'autre. C'est ce transfert d'énergie qui, en sciences physiques, est appelé chaleur. Dans cette activité, vous pourrez découvrir l'utilisation du capteur de température, intégré dans la carte. Un capteur de température est un dispositif électronique qui mesure la température de son environnement et convertit les données d'entrée en données électroniques pour enregistrer, surveiller ou signaler les changements de température.



## ÉTAPE 1 - CONSTRUIRE



### Connecter l'écran à la carte

Pour connecter l'écran LCD Grove, nous allons utiliser le bus **I2C**. Pour notre écran, nous utilisons la connexion **I2C** via le câble de liaison Grove avec la convention suivante :

- Rouge pour **V+** (3V3)
- Violet pour **SDA** (D14)
- Vert pour **SCL** (D15)

### Connecter la carte à l'ordinateur

Avec votre câble USB, connectez la carte à votre ordinateur en utilisant le **connecteur micro-USB ST-LINK** (sur le coin en haut à droite de la carte). Si tout se passe bien, vous devriez voir apparaître sur votre ordinateur un nouveau lecteur appelé **DIS\_L4IOT**. Ce lecteur est utilisé pour programmer la carte en copiant simplement un fichier binaire.

### Ouvrir MakeCode

Allez dans [l'éditeur MakeCode de Let's STEAM](#). Sur la page d'accueil, créez un nouveau projet en cliquant sur le bouton "Nouveau projet". Donnez à votre projet un nom plus expressif que "Sans titre" et lancez votre éditeur.

Ressource : [makecode.lets-steam.eu](http://makecode.lets-steam.eu)

### Programmer la carte

Dans l'éditeur JavaScript de MakeCode, copiez/collez le code disponible dans la section "**Programmer**" ci-dessous. Si ce n'est pas déjà fait, pensez à donner un nom à votre projet et cliquez sur le bouton "**Télécharger**". Copiez le fichier binaire sur le lecteur **DIS\_L4IOT** et attendez que la carte finisse de clignoter et votre datalogger est prêt !

### Exécuter, modifier, jouer

Votre programme s'exécutera automatiquement chaque fois que vous le sauvegarderez ou que vous réinitialiserez votre carte (appuyez sur le bouton intitulé RESET). Si tout fonctionne bien, votre carte changera l'état des LED pour montrer que la collecte de données est en cours. Essayez de comprendre l'exemple et commencez à le modifier en changeant la période entre deux mesures, en ajoutant d'autres données provenant d'autres capteurs de la carte. Essayez d'afficher autant de données que vous le souhaitez à plusieurs endroits pour comprendre comment la température évolue

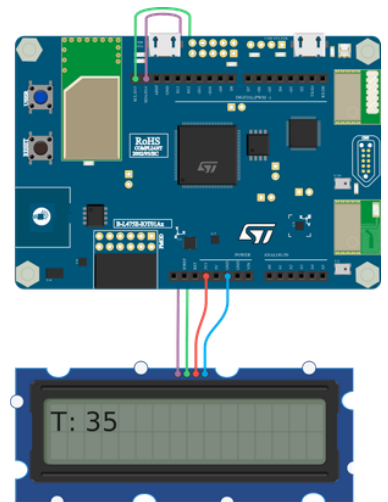
1

2

3

4

5



Connecter l'écran à la carte



## ÉTAPE 2 - PROGRAMMER



```
lcd.clear()  
forever(function () {  
  lcd.setCursor(0, 0)  
  lcd.ShowValue("T", input.temperature(TemperatureUnit.Celsius))  
  pause(500)  
})
```

### Comment cela fonctionne-t-il ?

Le code se compose de :

- un bloc **clear screen**
- un bloc **forever**
- un bloc **set cursor position**
- un bloc **show value**



L'écran LCD mémorise la position du prochain emplacement d'insertion. Lorsque nous voulons afficher quelque part à l'écran, nous devons toujours commencer par définir la position du curseur.

Avant d'écrire à l'écran, nous effaçons l'écran en appelant la fonction **LCD.clear()**.

À chaque itération de la boucle, avant d'écrire quelque chose, nous plaçons le curseur au début de l'écran (au premier caractère de la première ligne).

**input.temperature(TemperatureUnit.Celsius)** renvoie la valeur entière de la température en degrés Celsius. La valeur est affichée à l'écran avec la fonction **LCD.ShowValue()**. Le premier paramètre de cette fonction donne le nom de la valeur et le second, la valeur à afficher.

### Simulation du capteur de température

Vous pouvez jouer avec le capteur simulé en appuyant sur la **petite icône de thermomètre** affichée sur le simulateur de la carte. Vous pouvez modifier la valeur détectée (par exemple, en touchant avec le doigt le capteur physique de la carte), ce qui modifie en conséquence la valeur affichée sur l'écran LCD.



## ÉTAPE 3 - AMÉLIORER



Essayez de modifier le programme de cette activité pour **lire chaque capteur un par un et afficher sa valeur sur l'écran LCD**. Familiarisez-vous avec les différents capteurs disponibles. Essayez également d'utiliser des blocs supplémentaires de **LOGIC** ou **LCD** pour afficher du texte ou des valeurs.

**Ajoutez une condition qui modifie le rétroéclairage en fonction de la valeur de la température**. Par exemple, vous pouvez régler le rétroéclairage en bleu lorsque la température est inférieure à 10° et en rouge lorsque la température est supérieure à 20°.

Placez votre carte à différents endroits dans votre classe pour **créer un jeu de données comparables**. Si vous le souhaitez, vous pouvez également entrer en contact avec d'autres écoles dans votre pays ou à l'étranger pour élargir votre jeu de données et travailler sur des sujets météorologiques.

1



2



3



## ALLER PLUS LOIN



**Écran à cristaux liquides** - En savoir plus sur l'histoire et les caractéristiques des écrans à cristaux liquides (LCD).

[https://en.wikipedia.org/wiki/Liquid-crystal\\_display](https://en.wikipedia.org/wiki/Liquid-crystal_display)



**Réveil LCD avec de nombreux visages** - y compris beaucoup d'autres horloges LCD1602 trouvées sur des sites de fabricants.

<https://www.hackster.io/john-bradnam/lcd-alarm-clock-with-many-faces-new-version-9352a2>



**Le jeu Chrome Dino sur un écran LCD.**

[https://create.arduino.cc/projecthub/Unsigned\\_Arduino/the-chrome-dino-game-on-an-lcd-shield-883afb](https://create.arduino.cc/projecthub/Unsigned_Arduino/the-chrome-dino-game-on-an-lcd-shield-883afb)



**Luminomètre** - Mesurez et affichez les niveaux de lumière.

<https://learn.adafruit.com/light-meter>



## Explorer d'autres fiches d'activité

**R1AS10 - Affichage de texte avec un écran OLED**



**R1AS15 - Collecter des données**

