

# AFFICHAGE DE TEXTE

AVEC UN ÉCRAN OLED

#R1AS10



**Available on**

## De quoi parle-t-on ?

Un écran qui vous permet d'afficher certaines informations nichées dans vos composants électroniques.



## Prérequis

- R1AS03 - Boutons et affichage LED

## Material

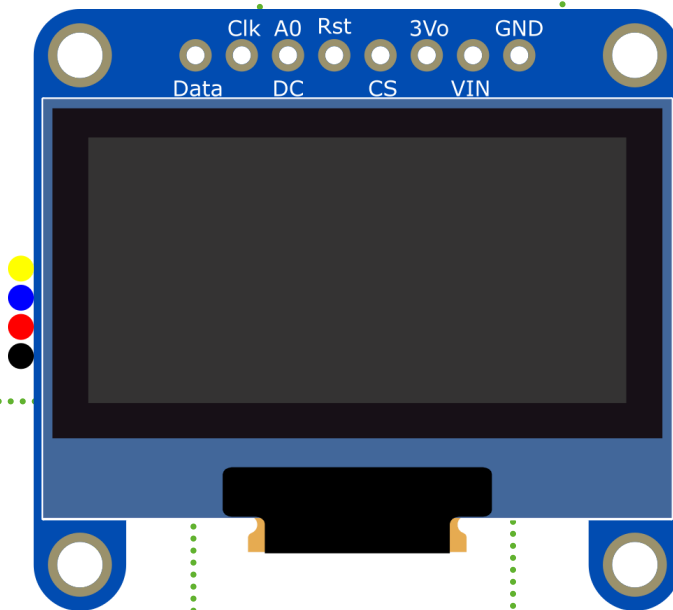
- 1 carte programmable "**STM32 IoT Node Board**"
- 1 câble USB Micro-B
- 1 écran OLED Monochrome 1.3" 128x64 OLED de Adafruit
- 1 câble QT pour connecter l'écran à la carte

## Durée

30 minutes

## Niveau de difficulté

Avancé



## OBJECTIFS D'APPRENTISSAGE

- Connecter un écran LCD à votre carte
- Afficher du texte sur votre écran LCD
- Placer du texte sur un écran
- Afficher l'état actuel de votre programme





La programmation d'une carte électronique est parfois une activité très déroutante. Un microcontrôleur est une boîte noire (nous ne pouvons pas voir ce qui se passe à l'intérieur). Pour éclaircir votre code, vous pouvez utiliser un écran qui vous aide à afficher certaines informations nichée dans vos composants électroniques. Cette fiche d'activité explore comment utiliser les **écrans OLED monochromes à base de SSD1306 avec MakeCode**.

*Ressource : <https://www.electronicwings.com/sensors-modules/ssd1306-oled-display>*



## ÉTAPE 1 - CONSTRUIRE



### Connecter la carte à l'écran

Il y a deux façons de câbler l'**OLED SSD1306** à une carte, soit avec une connexion **I2C** ou **SPI**. Pour notre écran, nous utilisons la connexion **I2C** via le câble **QWIIC/STEMMA** avec la convention suivante :

- **Noir** pour **GND**
- **Rouge** pour **V+ (3V3)**
- **Bleu** pour **SDA (D14)**
- **Jaune** pour **SCL (D15)**

*Ressources : <https://en.wikipedia.org/wiki/I2C>,  
[https://en.wikipedia.org/wiki/Serial\\_Peripheral\\_Interface](https://en.wikipedia.org/wiki/Serial_Peripheral_Interface),  
<https://www.sparkfun.com/qwiic>,  
<https://learn.adafruit.com/introducing-adafruit-stemma-q/what-is-stemma-q>*

### 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](https://makecode.lets-steam.eu). Sur la page d'accueil, créez un nouveau projet en cliquant sur le bouton "New Project". Donnez à votre projet un nom plus expressif que "Sans titre" et lancez votre éditeur.

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

### Installer l'extension

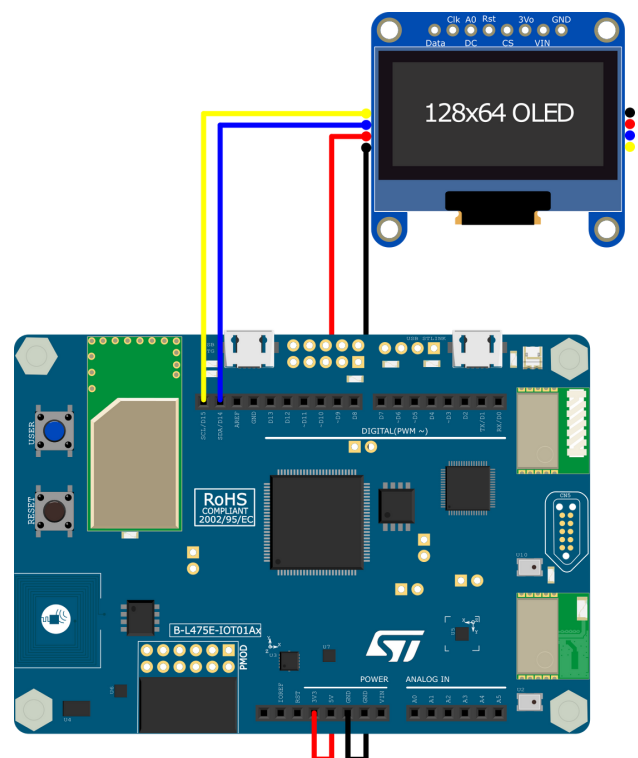
Après avoir créé votre nouveau projet, vous obtiendrez l'écran par défaut "prêt à l'emploi" illustré ici et vous devrez installer une extension.

1

2

3

4



Connecter la carte à l'écran



## ÉTAPE 1 - CONSTRUIRE



Les extensions dans MakeCode sont des groupes de blocs de code qui ne sont pas directement inclus dans les blocs de code de base que l'on trouve dans MakeCode. Les extensions, comme leur nom l'indique, ajoutent des blocs pour des fonctionnalités spécifiques. Il existe des extensions pour un large éventail de fonctionnalités très utiles, ajoutant des capacités de manette de jeu, de clavier, de souris, de servomoteurs, de la robotique et bien plus encore.

Vous voyez le bouton noir **ADVANCED** en bas de la colonne des différents groupes de blocs. Si vous cliquez sur **ADVANCED**, vous verrez apparaître des groupes de blocs supplémentaires. En bas, il y a une boîte grise appelée **EXTENSIONS**. Cliquez sur ce bouton.

Choisissez l'extension "**SSD1306 Display**".

### 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 "**Download**". Copiez le fichier binaire sur le lecteur **DIS\_L4IOT** et attendez que la carte finisse de clignoter et que votre programme affiche du texte !

### 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 va bien, votre carte vous adressera des salutations amicales. Essayez de comprendre l'exemple et commencez à le modifier en modifiant le texte, en ajoutant autant de symboles que possible ou en remplissant l'écran lentement, lettre par lettre.

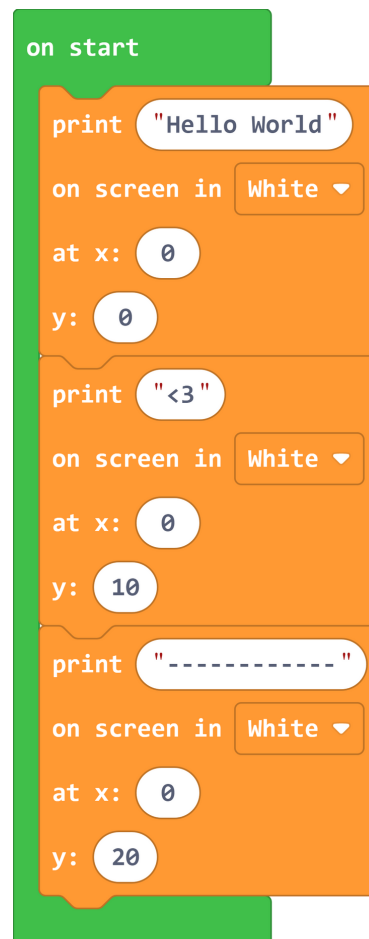
N'hésitez pas à essayer d'afficher n'importe quel élément d'information dans votre programme pour voir l'état actuel de votre carte.



Menu avancé comprenant l'outil "Extensions"

5

6



Blocs permettant l'exécution du programme



## ÉTAPE 2 - PROGRAMMER



```
oled.printString("Hello World", PixelColor.White, 0, 0)
oled.printString("<3", PixelColor.White, 0, 10)
oled.printString("-----", PixelColor.White, 0, 20)
```

### Comment cela fonctionne-t-il ?

Vous pouvez écrire une ligne de texte avec la fonction `printString()`. Cette fonction prend les paramètres suivants :

- Chaîne de texte
- Couleur du texte (PixelColor.Black ou PixelColor.White)
- Position du texte sur l'axe des X
- Position du texte sur l'axe des Y



Sur l'écran du SSD1306, l'origine (la position x=0 et Y=0) se trouve dans le coin supérieur gauche.



## ÉTAPE 3 - AMÉLIORER



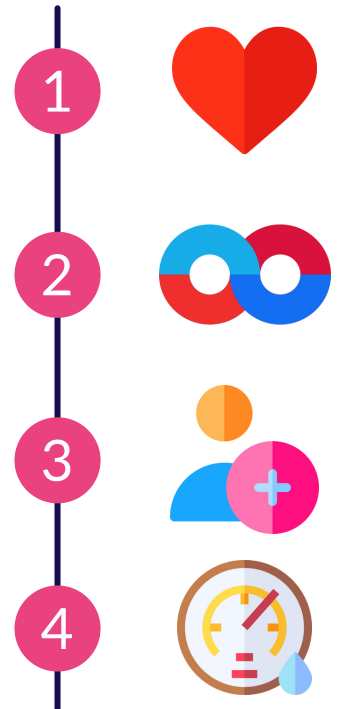
Essayez de **centrer le cœur de la deuxième ligne** en modifiant la coordonnée X du texte.

En ajoutant une boucle, **créez une animation de texte simple** dans l'esprit de **La Linea** en utilisant les **symboles | et \_**. Pour ralentir votre animation, utilisez la fonction `pause()`.

*Ressource : [https://en.wikipedia.org/wiki/La\\_Linea\\_TV\\_series](https://en.wikipedia.org/wiki/La_Linea_TV_series)*

**Montrer l'état courant du bouton USER à chaque instant.** Que se passe-t-il si vous ajoutez un `sleep()` plutôt long dans votre boucle principale ? Comment améliorer la réactivité de votre affichage ?

**Affichez la valeur de tous les capteurs embarqués.** Essayez de positionner chaque valeur à un endroit stratégique pour améliorer autant que possible la lisibilité.



## ALLER PLUS LOIN



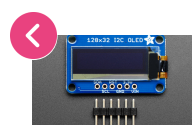
**I2C** - Tutoriel pour tout savoir sur le protocole de communication I2C, pourquoi et comment l'utiliser ainsi que le mettre en œuvre.  
<https://learn.sparkfun.com/tutorials/i2c/all>



**QWIIC/STEMMA** - Utilisez le level shifter et le régulateur de voltage avec les contrôleurs Grove/Gravity/STEMMA/Qwiic.  
<https://learn.adafruit.com/introducing-adafruit-stemma-qt/what-is-stemma-qt>



**Écran OLED** - Ecran à diode électroluminescente organique (OLED ou LED organique).  
<https://en.wikipedia.org/wiki/OLED>



### Explorer d'autres fiches d'activité

**R1AS09 - Capteur d'inclinaison avec l'accéléromètre**



**R1AS11 - Fabriquer un thermomètre ... très, très lisible**



**R1AS15 - Collecter des données**

