



# Adafruit 1,3" 128x64 OLED FeatherWing - SH1107 + 3 buttons (I2C)

[Mise à jour le 23/6/2022]



- **Ressources**

- [Adafruit 1,3" 128x64 OLED FeatherWing - SH1107 + 3 buttons](#)
- **Distribué** par [Mouser](#)

- **Lectures connexes**

- [Les afficheurs graphiques](#)
- [Bibliothèque - Adafruit GFX Graphics Library](#)
- [0,96" 128x64 OLED 2864 Display module - SSD1306 \(I2C\)](#)
- [Adafruit 1.8" 128x160 Color TFT LCD display with MicroSD Card v2 - ST7735R \(SPI\)](#)

---

## 1. Description

Le module complémentaire **Adafruit FeatherWing 128×64 OLED** pour **Feather** est un petit **écran monochrome de 1,3 pouces** de diagonale avec **trois boutons** pouvant être utilisés sur n'importe quelle carte mère Feather. Cet afficheur utilise un bus **I2C** pour se connecter et dispose d'un circuit de réinitialisation automatique et d'un bouton de réinitialisation sur le dessus. Il est composé de 128×64 pixels OLED blancs individuels et aucun rétroéclairage n'est requis car l'écran produit sa propre lumière. Le connecteur STEMMA QT compatible **SparkFun Qwiic** est inclus pour le bus I2C qui fournit le plug and play pour les appareils STEMMA QT, Qwiic et Grove I2C.



- **Caractéristiques**

- 128×64 OLED
- Écran monochrome de 1,3 pouces de diagonale
- Dimensions PCB 22,9 mm x 50,9 mm
- ~Zone d'affichage de 25,8 mm
- Connecteur STEMMA QT compatible SparkFun Qwiic pour le bus I2C

- **Bibliothèques à installer dans l'IDE Arduino ou dans PlatformIO (VSCode)**

- Adafruit gfx library
- Adafruit SH110X

## 2. Exemples

### 2.1 Un premier exemple pour tester l'afficheur avec l'IDE Arduino

- Fichier → Exemples → Adafruit SHT110X → **OLED\_featherwing.ino**

### 2.2 Mise en oeuvre de l'afficheur avec un capteur BME280

- **Description** : mesure de de la température, de l'humidité et de la pression à l'aide d'un capteur **Sparkfun BME280**, test des boutons-poussoirs et affichage sur un écran Oled **Adafruit SH1107**. L'écran et le capteur sont reliés via le système **Qwiic** de Sparkfun.



- **Matériels**

- Carte à microcontrôleur : [Adafruit Feather Huzzah ESP8266 + Support Particle](#)
- Afficheur : [Adafruit OLED SH1107](#)
- Capteur (température, humidité et pression) : [Sparkfun BME280](#)

- **Bibliothèques à installer dans l'IDE Arduino ou dans PlatformIO (VSCode)**

- Adafruit GFX Library by Adafruit [\[GitHub\]](#)
- Adafruit SH110X by Adafruit [\[GitHub\]](#)
- SparkFun BME280 by SparkFun Electronics [\[GitHub\]](#)



## • Code Arduino

\*.cpp

```
// Matériels : Adafruit Feather Huzzah ESP8266 + Support Particle,
// Adafruit OLED SH1107, Sparkfun BME280, câble Qwiic
// Logiciel : Arduino

#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SH110X.h>
#include "SparkFunBME280.h"

#define BUTTON_A 0
#define BUTTON_B 16
#define BUTTON_C 2

// Constructeurs
Adafruit_SH1107 display = Adafruit_SH1107(64, 128, &Wire);
BME280 bme_280; // L'adresse du circuit BME280 est 0x77 par défaut

void setup()
{
    // Bus I2C
    Wire.begin(); // Initialisation
    Wire.setClock(400000); // Fast I2C
    display.begin(0x3C, true); // L'adresse de l'afficheur est 0x3C par
    défaut

    // Configuration de l'affichage
    display.setRotation(1); // Affichage horizontal
    display.setTextSize(1); // Horizontal
    display.setTextColor(SH110X_WHITE);
    display.clearDisplay(); // Pour ne pas afficher le logo Adafruit
    chargé

    // automatiquement à la mise sous tension
    // Test de la communication avec le capteur
    if (bme_280.beginI2C() == false)
    {
        display.println("DEFAULT(s)");
        display.println("1. Le capteur BME280 ne repond pas ! ");
        display.println();
        display.print("BLOPAGE du PROGRAMME");
        display.display(); // Transfert du buffer sur l'écran
        while (1)
            delay(10); // Blocage du programme
    }
}
```

```
}

// Connexion des boutons-poussoir
pinMode(BUTTON_A, INPUT_PULLUP);
pinMode(BUTTON_B, INPUT_PULLUP);
pinMode(BUTTON_C, INPUT_PULLUP);
}

void loop()
{
    // Efface le buffer
    display.clearDisplay();

    // Test des boutons
    display.setCursor(0, 0);

    if (!digitalRead(BUTTON_A))
        display.print("[A]");
    if (!digitalRead(BUTTON_B))
        display.print("[B]");
    if (!digitalRead(BUTTON_C))
        display.print("[C]");

    // Titre
    display.setCursor(20, 0);
    display.println("Sparkfun BME280");

    // Humidité
    display.setCursor(0, 12);
    display.print("Humidite : ");
    display.print(bme_280.readFloatHumidity(), 0);
    display.println("%");

    // Pression en hPa
    display.setCursor(0, 22);
    display.print("Pression : ");
    display.print(bme_280.readFloatPressure() / 100, 0);
    display.println("hPa");

    // Température
    display.setCursor(0, 32);
    display.print("Temp : ");
    display.print(bme_280.readTempC(), 1);
    display.print("C");

    // Infos
    display.setCursor(5, 52);
    display.print("Appuyer sur A, B, C");

    // yield();
}
```

```
display.display(); // Transfert du buffer sur l'écran  
delay(10);  
}
```



Télécharger le projet PlatformIO pour VSCode.

From:

<http://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:

<http://webge.fr/dokuwiki/doku.php?id=materiels:afficheurs:ada1107&rev=1655979037>

Last update: **2022/06/23 12:10**

