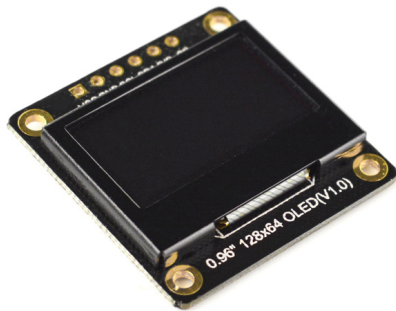


SKU:DFR0650 (<https://www.dfrobot.com/product-2017.html?search=DFR0650&description=true>)




(<https://www.dfrobot.com/product-2017.html?search=DFR0650&description=true>)

Introduction

This 0.96" OLED display module uses the SSD1306 drive chip, has 128x64 self-illuminating white pixels, and adopts two communication methods: SPI/ IIC, IIC in default. With its high contrast, simple wiring and low power consumption, it can be applied to many display applications, such as wearable display devices, Mini small game consoles, desktop widgets, etc.

This 0.96" OLED display module supports both 3.3V and 5V. The display module adopts a high-contrast, low-power OLED display. It is compatible with controllers such as Arduino UNO, Leonardo, ESP32/ESP8266, FireBettle-M0 and so on. Internal power-on reset processing, IIC communication methods, so that your welding line will be more convenient and simpler. Power-on reset is supported, so the Res pin is not elicited. This is a Breakout display module that fits perfectly with the breadboard when you weld the pin. If you want to change IIC communication to SPI, you just need to change the 00hm resistor position.

In addition, the display is in an aluminum frame package that protects the screen from damage, and also prevents you from scratching that caused by the glass edge of the screen.

 **NOTE:** Turn the pixels off when you are not using the display to prevent the use of some pixels too long to darken them, which will cause the brightness uniformity problem.

Features

- IIC/SPI communication, IIC in default
- Self-illuminating OLED display
- Controllable pixels and low power consumption
- Supports power-on reset

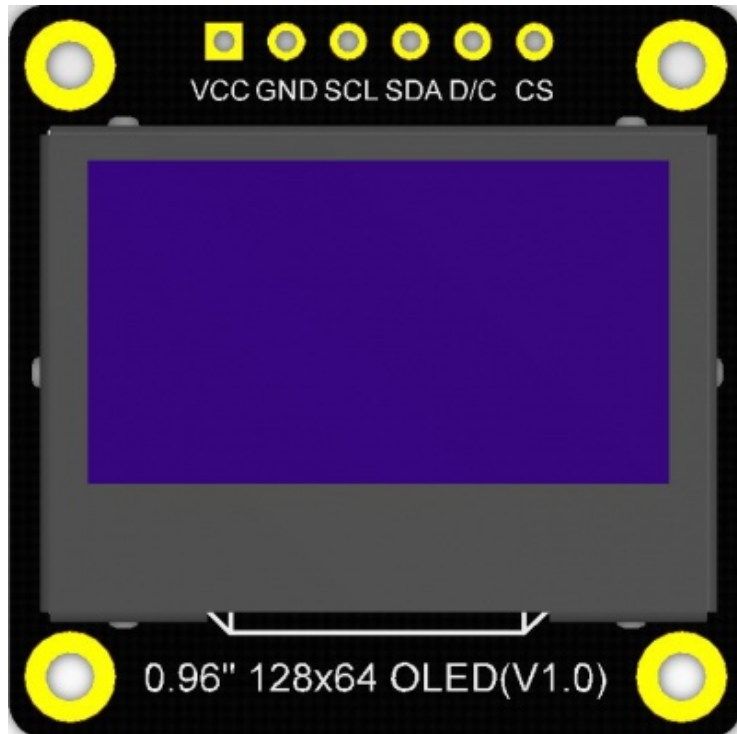
Specification

- Operating voltage: 3.3V to 5V
- Color: white pixels
- Number of pixels: 128 columns x 64 rows
- Interface: IIC (in default) / SPI
- IIC address: 0x3C
- Drive chip: SSD1306
- Brightness: 180 (Typ) cd/m²
- Full-screen lighting power: approx. 24mA-3.3V/5V(IIC), approx. 27mA-3.3V/5V(SPI);
- Full-screen off power consumption: approx. 2mA-3.3V/5V(IIC), approx. 2.87mA-3.3V/5V(SPI)
- Operating temperature: -30°C to 70°C
- Display area: 21.744 x 10.864mm / 0.86" x 0.42"
- Mounting hole diameter: 2.0mm
- Size: 30.00 x 30.00mm / 1.18" x 1.18"

- Weight: 4.2g

This product is supplied with 2.54mm male headers by default, and each connector has a minimum of 10 pins. If the number of pins exceeds your usage needs, please use pliers to break them off yourself.

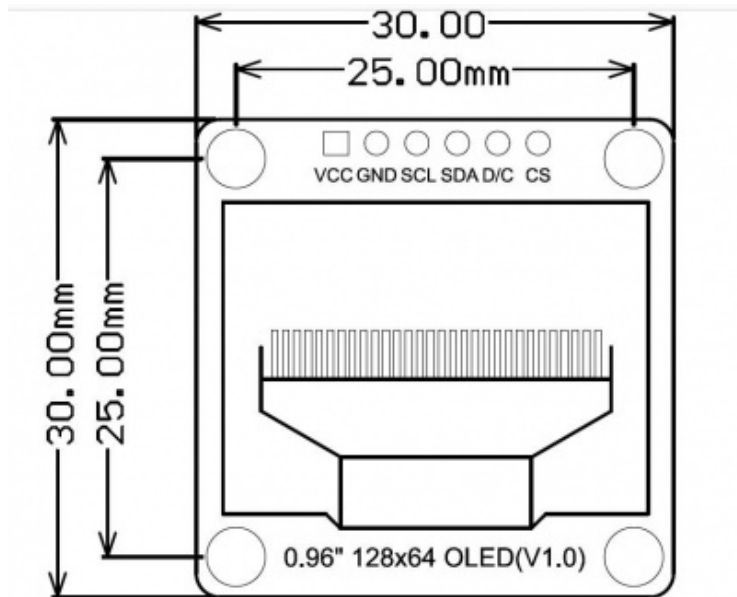
Board Overview



Num	Label	Description
1	VCC	+
2	GND	-
3	SCL	IIC/SPI System Clock Line
4	SDA	IIC/SPI System Data Line
5	D/C	SPI data/command

6	CS	SPI Chip Select
---	----	-----------------

Dimension



Tutorial

The product is a Breakout module that supports both IIC/SPI communication interfaces, which default to the IIC interface. Internal power-on reset, so resnbeu pin is not exposed

NOTE:

- The IIC address is 0x3C
- It is recommended to use Arduino 1.8.9 and above.

Requirements

- **Hardware**

- DFRduino UNO R3 (<https://www.dfrobot.com/product-838.html>) (or similar) x 1
- 0.96inch 128x64 IIC SPI OLED Module x 1
- M-M/F-M/F-F Jumper wires

- **Software**

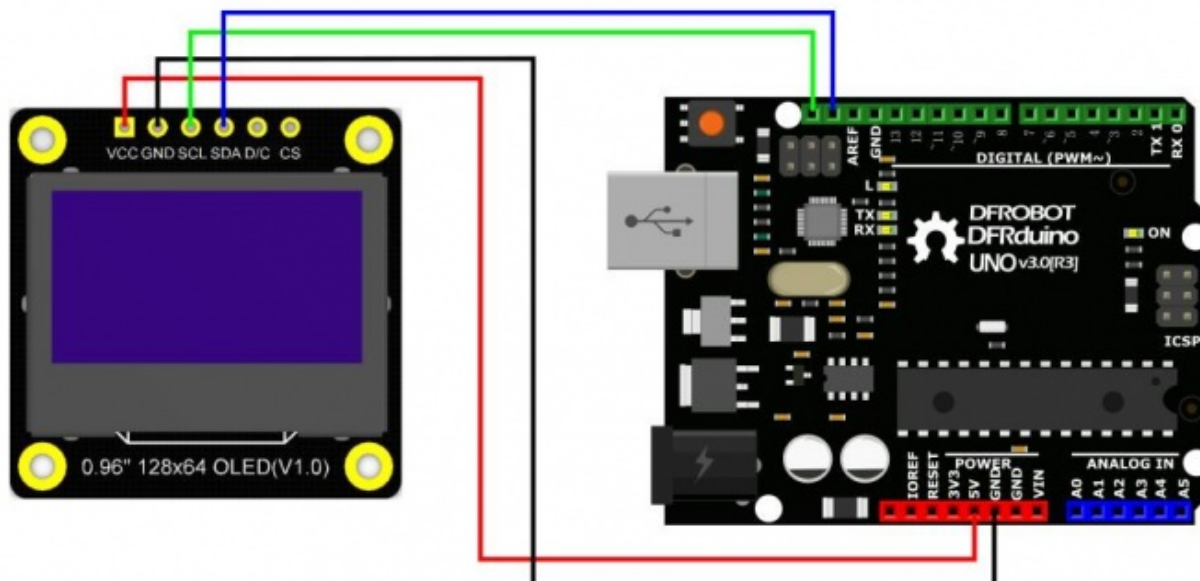
- Arduino IDE (<https://www.arduino.cc/en/Main/Software>)
- Download and install the **0.96" 128x64 u8g2 Library** (https://github.com/DFRobot/U8g2_Arduino/archive/master.zip) (About how to install the library? (<https://www.arduino.cc/en/Guide/Libraries#UxU8mdzF9H0>))
- Download and install the **[0.96" 128x64 DFRobot_GDL Library]**(Not released yet)

1.u8g2 API Functions (<https://github.com/olikraus/u8g2/wiki/u8g2reference#updateddisplayarea=%E5%BA%93%E6%96%87%E4%BB%B6>)

2.u8g2 Font Parameters (<https://github.com/olikraus/u8g2/wiki/fntlistall>)

3.DFRobot_GDL API Functions (<https://www.dfrobot.com/>)

Connection Diagram



Sample Code1 - Hello World!

Let's start with a simple and classic "Hello World!"

```
/*!
 * @file HelloWorld.ino
 * @brief A simple and classic "Hello World!"
 * @n U8G2 supports fonts of multiple sizes, and this demo is just one font size to choose for display.
 * @n U8G2 Font GitHub Link: https://github.com/olikraus/u8g2/wiki/fntlistall
 *
 * @copyright Copyright (c) 2010 DFRobot Co.Ltd (https://www.dfrobot.com)
 * @licence The MIT License (MIT)
 * @author [Ivey](Ivey.lu@dfrobot.com)
 * @version V1.0
 * @date 2019-11-29
 * @get from https://www.dfrobot.com
 * @url https://github.com/DFRobot/U8g2\_Arduino
 */
#include <Arduino.h>
#include <U8g2lib.h>

//#include <SPI.h>
#include <Wire.h>

/*!
 * Display hardware IIC interface constructor
 * @param rotation: U8G2_R0 Not rotate, horizontally, draw direction from left to right
 *                 U8G2_R1 Rotate clockwise 90 degrees, drawing direction from top to bottom
 *                 U8G2_R2 Rotate 180 degrees clockwise, drawing in right-to-left directions
 *                 U8G2_R3 Rotate clockwise 270 degrees, drawing direction from bottom to top
 *                 U8G2_MIRROR Normal display of mirror content (v2.6.x version used above)
 *                 Note: U8G2_MIRROR need to be used with setFlipMode().
 * @param reset: U8x8_PIN_NONE Indicates that the pin is empty and no reset pin is used
 * Display hardware SPI interface constructor
 * @param Just connect the CS pin (pins are optional)
 * @param Just connect the DC pin (pins are optional)
 *
 */
```

```
U8G2_SSD1306_128X64_NONAME_F_HW_I2C u8g2(/* rotation=*/U8G2_R0, /* reset=*/ U8X8_PIN_NONE); // M0/ESP32/ESP8266/mega2560/Uno/Leonard
//U8G2_SSD1306_128X64_NONAME_1_4W_HW_SPI u8g2(/* rotation=*/U8G2_R0, /* cs=*/ 10, /* dc=*/ 9);
// End of constructor list

void setup(void) {
  u8g2.begin();
}

void loop(void) {
  u8g2.clearBuffer();           // clear the internal memory
  u8g2.setFont(u8g2_font_t0_17b_tr ); // choose a suitable font
  u8g2.drawStr(0,10,"Hello World!"); // write something to the internal memory
  u8g2.sendBuffer();           // transfer internal memory to the display
  delay(1000);
}
```


Expected Result1



Sample Code2 - Show beautiful fonts

Do you still use only monotonous English and digital fonts? Do you want a nice fancy font that makes your app attractive, even if it's just "Hello World! "? So try this example.

Please go to [u8g2wiki](https://u8g2wiki.com) for more fonts. There are many good-looking icons, multi-language words, etc.

```
/*!
 * @file Font.ino
 * @brief Show several U8G2-supported fonts
 * @n U8G2 supports multiple fonts, and this demo is just a few fonts to display
 * @n U8G2 Font GitHub https://github.com/olikraus/u8g2/wiki/fntlistall
 *
 * @copyright Copyright (c) 2010 DFRobot Co.Ltd (https://www.dfrobot.com)
 * @licence The MIT License (MIT)
 * @author [Ivey](Ivey.lu@dfrobot.com)
 * @version V1.0
 * @date 2019-11-29
 * @get from https://www.dfrobot.com
 * @url https://github.com/DFRobot/U8g2\_Arduino
 */

#include <Arduino.h>
#include <U8g2lib.h>

// #include <SPI.h>
#include <Wire.h>

/*!
 * Display hardware IIC interface constructor
 * @param rotation: U8G2_R0 Not rotate, horizontally, draw direction from left to right
 *                 U8G2_R1 Rotate clockwise 90 degrees, drawing direction from top to bottom
 *                 U8G2_R2 Rotate 180 degrees clockwise, drawing in right-to-left directions
 *                 U8G2_R3 Rotate clockwise 270 degrees, drawing direction from bottom to top
 *                 U8G2_MIRROR Normal display of mirror content (v2.6.x version used above)
 *                 Note: U8G2_MIRROR need to be used with setFlipMode().
 * @param reset: U8x8_PIN_NONE Indicates that the pin is empty and no reset pin is used
 * Display hardware SPI interface constructor
 * @param Just connect the CS pin (pins are optional)
 * @param Just connect the DC pin (pins are optional)
 */
```

```
*/

U8G2_SSD1306_128X64_NONAME_F_HW_I2C u8g2(/* rotation=*/U8G2_R0, /* reset=*/ U8X8_PIN_NONE); // M0/ESP32/ESP8266/mega2560/Uno/Leonard
//U8G2_SSD1306_128X64_NONAME_1_4W_HW_SPI u8g2(/* rotation=*/U8G2_R0, /* cs=*/ 10, /* dc=*/ 9);

void setup(void)
{
  u8g2.begin(); //Initialize the function
  u8g2.setFontPosTop(); /*When you use drawStr to display strings, the default criteria is to display the lower-left
  * coordinates of the characters.The function can be understood as changing the coordinate position to the upper left
  * corner of the display string as the coordinate standard.
  */
}

void draw(int a )
{
  switch(a)
  {
    case 0:
      u8g2.setFont(u8g2_font_bubble_tr); //Set the font set, which is“u8g2_font_bubble_tr”
      u8g2.drawStr(/* x=*/0,/* y=*/0, "DFR123"); //Start drawing strings at the coordinates of x=0, y=0 “DFR123”
      u8g2.setFont(u8g2_font_lucasarts_scumm_subtitle_o_tf);
      u8g2.drawStr(0, 25, "DFR123");
      u8g2.setFont(u8g2_font_HelvetiPixelOutline_tr);
      u8g2.drawStr(0, 45, "DFR123");
      break;
    case 1:
      u8g2.setFont(u8g2_font_tenstamps_mr);
      u8g2.drawStr(0,0, "DFR123");
      u8g2.setFont(u8g2_font_jinxedwizards_tr);
      u8g2.drawStr(56, 16, "DFR123");
      u8g2.setFont(u8g2_font_secretaryhand_tr);
      u8g2.drawStr(0, 32, "DFR123");
      u8g2.setFont(u8g2_font_freedomr10_mu);
      u8g2.drawStr(56, 48, "DFR123");
      break;
  }
}
```

```
void loop()
{
  for( int i = 0; i <2 ; i++)
  {
    /*
     * firstPage will change the current page number position to 0
     * When modifications are between firstpage and nextPage, they will be re-rendered at each time.
     * This method consumes less ram space than sendBuffer
     */
    u8g2.firstPage();
    do
      {
        draw(i);
      } while( u8g2.nextPage() );
    delay(2000);
  }
}
```

Expected Result2



Sample Code3 - Font display in different directions

The U8G2 supports font display in different directions. It can be used with some gravity sensors for interesting applications.

```
/*!
 * @file FontDirection.ino
 * @brief Display of several font directions supported in U8G2
 * @n U8G2 supports multiple fonts, and this demo shows only four directions (no mirrors are shown).
 * @n U8G2 Font GitHub https://github.com/olikraus/u8g2/wiki/fntlistall
 *
 * @copyright Copyright (c) 2010 DFRobot Co.Ltd (https://www.dfrobot.com)
 * @licence The MIT License (MIT)
 * @author [Ivey](Ivey.lu@dfrobot.com)
 * @version V1.0
 * @date 2019-11-29
 * @get from https://www.dfrobot.com
 * @url https://github.com/DFRobot/U8g2\_Arduino
 */

#include <Arduino.h>
#include <U8g2lib.h>

// #include <SPI.h>
#include <Wire.h>

/*!
 * Display hardware IIC interface constructor
 * @param rotation: U8G2_R0 Not rotate, horizontally, draw direction from left to right
 *                 U8G2_R1 Rotate clockwise 90 degrees, drawing direction from top to bottom
 *                 U8G2_R2 Rotate 180 degrees clockwise, drawing in right-to-left directions
 *                 U8G2_R3 Rotate clockwise 270 degrees, drawing direction from bottom to top
 *                 U8G2_MIRROR Normal display of mirror content (v2.6.x version used above)
 *                 Note: U8G2_MIRROR need to be used with setFlipMode().
 * @param reset: U8x8_PIN_NONE Indicates that the pin is empty and no reset pin is used
 * Display hardware SPI interface constructor
 * @param Just connect the CS pin (pins are optional)
 * @param Just connect the DC pin (pins are optional)

```

```

*
*/

U8G2_SSD1306_128X64_NONAME_F_HW_I2C u8g2(/* rotation=*/U8G2_R0, /* reset=*/ U8X8_PIN_NONE); // M0/ESP32/ESP8266/mega2560/Uno/Leonarc

//U8G2_SSD1306_128X64_NONAME_1_4W_HW_SPI u8g2(/* rotation=*/U8G2_R0, /* cs=*/ 10, /* dc=*/ 9);

//width:30,height:30
const uint8_t col[] U8X8_PROGMEM= {0x00,0xc0,0x00,0x00,0x00,0xe0,0x01,0x00,0x00,0xe0,0x01,0x00,0x00,
                                0xc0,0x00,0x00,0xc0,0x00,0x00,0xc0,0x00,0x00,0xc0,0x00,0x00,0xe0,
                                0x01,0x00,0x00,0xf8,0x07,0x00,0x06,0xfe,0x1f,0x18,0x07,0xff,0x3f,
                                0x38,0xdf,0xff,0xff,0x3e,0xfa,0xff,0xff,0x17,0xf0,0xff,0xff,0x03,
                                0xe0,0xff,0xff,0x01,0xe0,0xff,0xff,0x01,0xe0,0xff,0xff,0x01,0xe0,
                                0xff,0xff,0x01,0x20,0x00,0x00,0x01,0xa0,0xff,0x7f,0x01,0xa0,0x01,
                                0x60,0x01,0x20,0x07,0x38,0x01,0xe0,0x0c,0xcc,0x01,0xe0,0x39,0xe7,
                                0x01,0xe0,0xe7,0xf9,0x01,0xc0,0x1f,0xfe,0x00,0x80,0xff,0x7f,0x00,
                                0x00,0xfe,0x1f,0x00,0x00,0xf8,0x07,0x00,0x00,0xe0,0x01,0x00,0x00,
                                0xc0,0x00,0x00};

void setup() {
  u8g2.begin();
  u8g2.setFontPosTop();/**When you use drawStr to display strings, the default criteria is to display the lower-left
 * coordinates of the characters.The function can be understood as changing the coordinate position to the upper left
 * corner of the display string as the coordinate standard.*/
}

void Rotation() {
  u8g2.setFont(u8g2_font_bracketedbabies_tr          );
  u8g2.firstPage();
  do {
    u8g2.drawXBMP( /* x=*/0 , /* y=*/0 , /* width=*/30 , /* height=*/30 , col );//Draw an image
    /*@brief Set the drawing direction of all strings or glyphs setFontDirection(uint8_t dir)
    *@param dir=0
            dir=1, rotate 0°
            dir=2, rotate 180°
            dir=3, rotate 270°

    */
    u8g2.setFontDirection(0);
    u8g2.drawStr( /* x=*/64 /* y=*/32 " DFR" ); //Draw a string

```

```

u8g2.setFontDirection(1);
u8g2.drawStr(64,32, " DFR");
u8g2.setFontDirection(2);
u8g2.drawStr(64,32, " DFR");

u8g2.setFontDirection(3);
u8g2.drawStr(64,32, " DFR");
}while(u8g2.nextPage());
delay(1500);
}

void loop(void)
{
  u8g2.setFont( u8g2_font_sticker_mel_tr );
  for(int i = 0 ; i < 4 ; i++ )
  {
    switch(i)
    {
      case 0:
        u8g2.setFontDirection(0);
        break;
      case 1:
        u8g2.setFontDirection(1);
        break;
      case 2:
        u8g2.setFontDirection(2);
        break;
      case 3:
        u8g2.setFontDirection(3);
        break;
    }
  }
  /*
   * firstPage will change the current page number position to 0
   * When modifications are between firstpage and nextPage, they will be re-rendered at each time.
   * This method consumes less ram space than sendBuffer
  */
  u8g2.firstPage();
  do
  {
    u8g2.drawStr(64, 32, "DFR");

```



```
    u8g2.drawXBMP( 0, 0, 30, 30, col );  
  }while(u8g2.nextPage());  
  delay(500);  
}  
  
Rotation();  
}
```

Expected Result3





IIC and SPI Wiring

- IIC communication

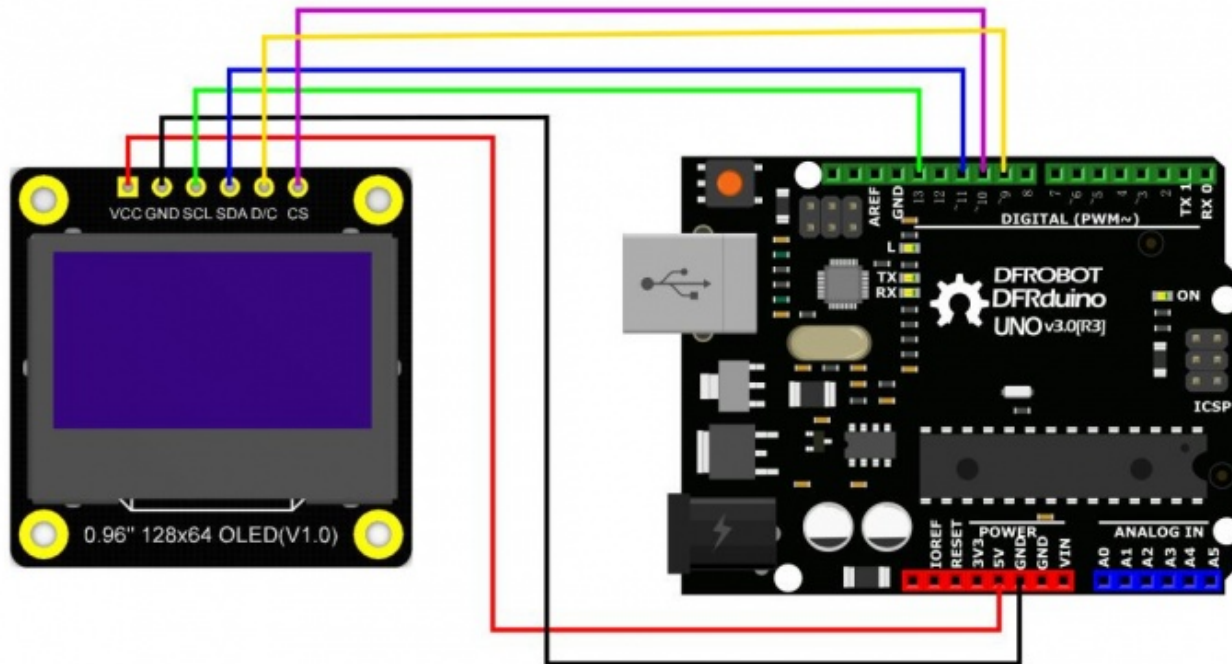


NOTE: When using SPI, the 0 ohm resistance behind the module needs to be replaced to the appropriate position.

- SPI communication



- SPI Wiring



Sample Code4 - GIF

This is a rotating tetrahexon, and for a smooth display, it is recommended that you turn on the solidization function of the SPI.

```
/*!
 * @file Cube.ino
 * @brief Rotating 3D stereoscopic graphics
 * @n This is a simple rotating tetrahexon
 *
 * @copyright Copyright (c) 2010 DFRobot Co.Ltd (https://www.dfrobot.com)
 * @licence The MIT License (MIT)
 * @author [Ivey](Ivey.lu@dfrobot.com)
 * @version V1.0
 * @date 2019-11-29
 * @get from https://www.dfrobot.com
 * @url https://github.com/DFRobot/U8g2\_Arduino
 */

#include <Arduino.h>
#include <U8g2lib.h>

#include <SPI.h>
// #include <Wire.h>

/*!
 * Display hardware IIC interface constructor
 * @param rotation: U8G2_R0 Not rotate, horizontally, draw direction from left to right
 *                 U8G2_R1 Rotate clockwise 90 degrees, drawing direction from top to bottom
 *                 U8G2_R2 Rotate 180 degrees clockwise, drawing in right-to-left directions
 *                 U8G2_R3 Rotate clockwise 270 degrees, drawing direction from bottom to top
 *                 U8G2_MIRROR Normal display of mirror content (v2.6.x version used above)
 *                 Note: U8G2_MIRROR need to be used with setFlipMode().
 * @param reset: U8x8_PIN_NONE Indicates that the pin is empty and no reset pin is used
 * Display hardware SPI interface constructor
 * @param Just connect the CS pin (pins are optional)
 * @param Just connect the DC pin (pins are optional)
 */
```



```
,
/*!
 * @brief Convert the yz in the three-dimensional coordinate system Oxyz into the v coordinate inside
 * the two-dimensional coordinate system Ouv
 * @param y in Oxyz
 * @param z in Oxyz
 * @return v in Ouv
 */
int OxyzTo0v(double y,double z){
    return (int)((y + 35) - z*0.35);
}

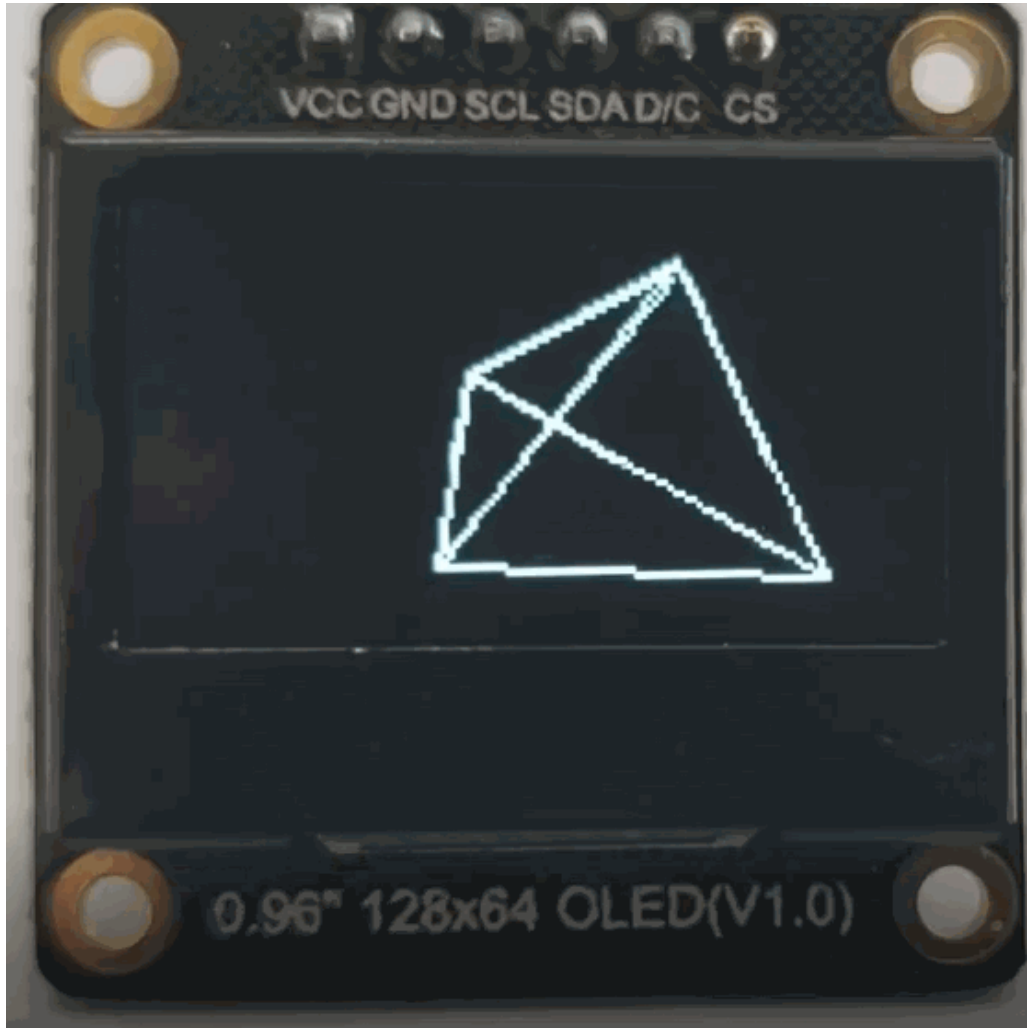
/*!
 * @brief Rotate the coordinates of all points of the entire 3D graphic around the Z axis
 * @param angle represents the angle to rotate
 *
 * z rotation (z unchanged)
 *  $x_3 = x_2 * \cos b - y_1 * \sin b$ 
 *  $y_3 = y_1 * \cos b + x_2 * \sin b$ 
 *  $z_3 = z_2$ 
 */
void rotate(double angle)
{
    double rad, cosa, sina, Xn, Yn;

    rad = angle * PI / 180;
    cosa = cos(rad);
    sina = sin(rad);
    for (int i = 0; i < 4; i++)
    {
        Xn = (tetrahedron[i][0] * cosa) - (tetrahedron[i][1] * sina);
        Yn = (tetrahedron[i][0] * sina) + (tetrahedron[i][1] * cosa);

        //Store converted coordinates into an array of coordinates
        //Because it rotates around the Z-axis, the coordinates of the point z-axis remain unchanged
        tetrahedron[i][0] = Xn;
        tetrahedron[i][1] = Yn;
    }
}
```

```
    digitalWrite(LED_PIN, HIGH);  
}  
}
```

Expected Result4



Compatibility Test

MCU	Pass	Failed	Not Tested	Remark
FireBeetle-ESP32	√			

MCU	Pass	Failed	Not Tested	Remark
FireBeetle-ESP8266	√			
Arduino Uno	√			
Leonardo	√			
Mega2560	√			
Arduino M0	√			


FAQ

Q&A	Some general Arduino Problems/FAQ/Tips
Q1	There are two drives u8g2 and DFRobot_GDL are provided, should I install both of them?
A1	You just need to select one to install. Please note that u8g2 is a powerful lcd library for black and white liquid crystals, and there are functions like built-in capacitive resistance touch, simple UI and u8g2 partial functions in DFRobot_SDL, it supports both black and white, and colorful LCD.
Q2	I have a question!
A2	For any questions, advice or cool ideas to share, please visit the DFRobot Forum (https://www.dfrobot.com/forum/).

More Documents

- DFR0650-Schematic (<https://dfimg.dfrobot.com/nobody/wiki/0eb3b6147b17a8a1b0715205c6357149.pdf>)
- DFR0650-Dimensions (<https://dfimg.dfrobot.com/nobody/wiki/0806c15e1652a6d20dd3d64bc73ea472.pdf>)
- DFR0650-Datasheet (<https://dfimg.dfrobot.com/nobody/wiki/0211ed8b67f6e7ef41990e62e72086bc.pdf>)

- DFR0650-Datasheet (https://wiki.dfrobot.com/Monochrome_0.96_128x64_IIC_SPI_OLED_Display_SKU_DFR0650)

 Get **0.96" 128x64 OLED** (<https://www.dfrobot.com/product-2017.html>) from DFRobot Store or **DFRobot Distributor**.
(<https://www.dfrobot.com/distributor>)

Turn to the Top