

# Unit OLED

SKU:U119



## Description

**Unit OLED** is a 1.3-inch OLED expansion screen unit. It adopts SH1107 drive scheme, supports black/white display, and the resolution is 128\*64. The driver chip selects the I2C communication interface, and the user can mount it to the I2C bus of the existing device when in use, which saves IO. The back of the screen is integrated with a magnetic design, which can easily be fixed on the metal surface by adsorption. The OLED screen extension is suitable for being embedded in various instruments or control devices that need to display simple content as a display panel.

## Product Features

- OLED display panel
- SH1107 drive scheme
- Display color: black/white
- I2C communication interface
- 128\*64 resolution
- Viewing angle: full viewing angle
- Magnetic back design

## Include

- 1x Unit OLED
- 1x HY2.0-4P Cable

## Applications

- Information display

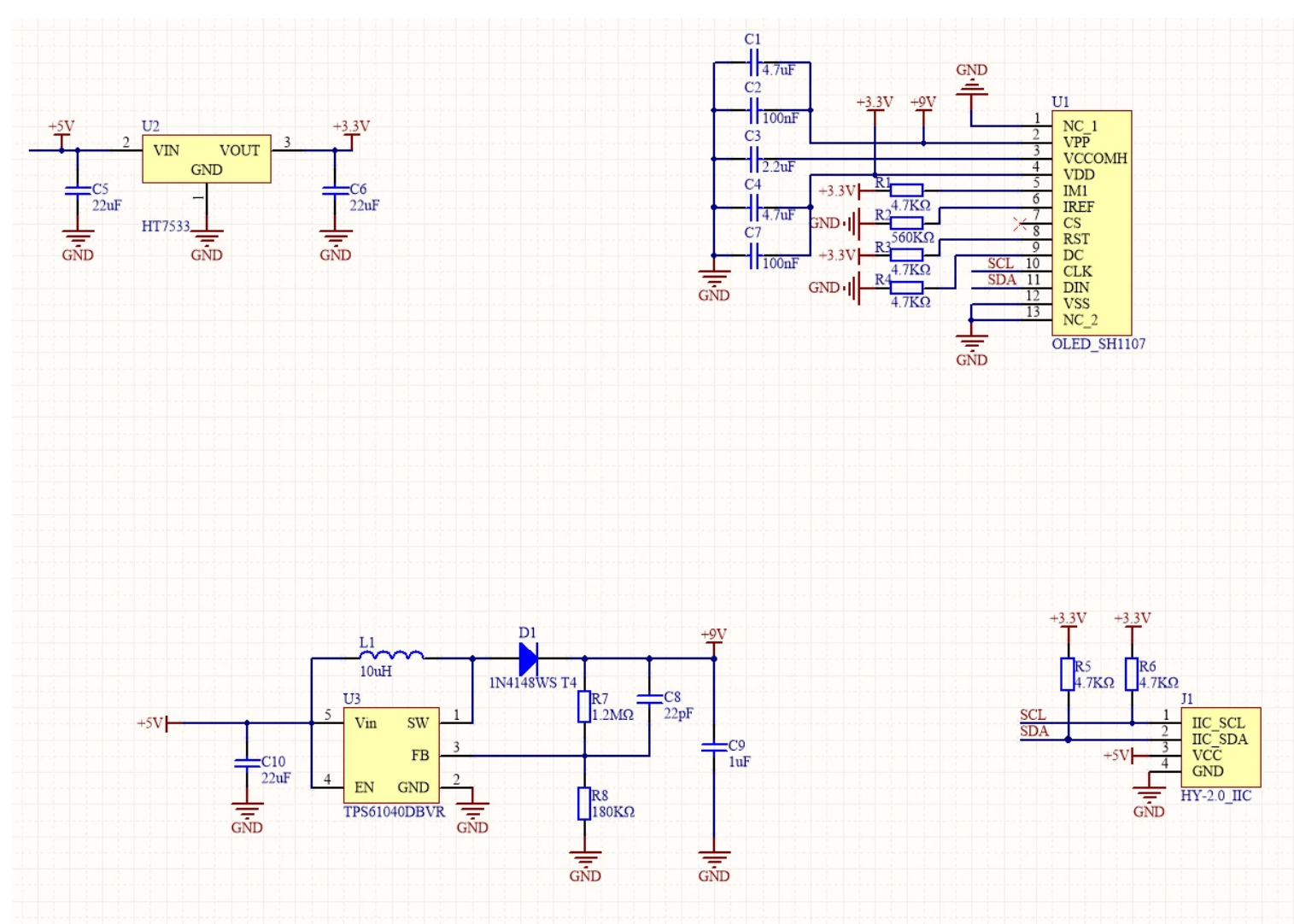
## Specification

Specification	Parameter
Working current	58mA
Communication protocol	I2C address: 0x3C
Display size	14.7 × 29.42 (mm)
Pixel pitch	0.17×0.17 (mm)
Pixel size	0.15×0.15 (mm)
Viewable	Full view
Operating temperature	0°C to 60°C
Net weight	9.3g
Gross weight	20g
Product Size	24*56*8mm
Package Size	67*53*12mm
Case material	Plastic ( PC )

## PinMap

M5Core(PORT A)	GPIO22	GPIO21	5V	GND
Unit OLED	SCL	SDA	5V	GND

## Schematic



## Related Link

- **Datasheet**

- [SH1107](#)

- **Library**

- [M5GFX Library](#)



# Example

---

◦ Arduino

```
#include <M5UnitOLED.h>

M5UnitOLED display;

M5Canvas canvas(&display);

static constexpr char text[] = "Hello world ! こんにちは世界! this is long long string sample. 寿限無、寿限無、五劫の擦り切れ、海砂利水魚の、水行末・雲来末・風来末、喰う寝る処に住む処、藪ら柑子の藪柑子、パイポ・パイポ・パイポのシューリンガン、シューリンガンのゲーリンダイ、ゲーリンダイのポンポコピーのポンポコナの、長久命の長助";
static constexpr size_t textlen = sizeof(text) / sizeof(text[0]);
int textpos = 0;
int scrollstep = 2;

void setup(void)
{
    display.init();
    display.setRotation(2);
    canvas.setColorDepth(1); // mono color
    canvas.setFont(&font::lgfxJapanMinchoP_32);
    canvas.setTextWrap(false);
    canvas.setTextSize(2);
    canvas.createSprite(display.width() + 64, 72);
}

void loop(void)
{
    int32_t cursor_x = canvas.getCursorX() - scrollstep;
    if (cursor_x <= 0)
    {
        textpos = 0;
        cursor_x = display.width();
    }

    canvas.setCursor(cursor_x, 0);
    canvas.scroll(-scrollstep, 0);
    while (textpos < textlen && cursor_x <= display.width())
    {
        canvas.print(text[textpos++]);
        cursor_x = canvas.getCursorX();
    }
    display.waitDisplay();
    canvas.pushSprite(&display, 0, (display.height() - canvas.height()) >> 1);
}
```

# Video

---