



1.3inch OLED HAT USER MANUAL

OVERVIEW

This product is 1.3-inch OLED display dedicated for use with Raspberry Pi. It has resolution 128 x 64, internal controller, and can be controlled by I2C or SPI interfaces. It has a joystick and three buttons, which can be convenient at application's extensions.

FEATURES

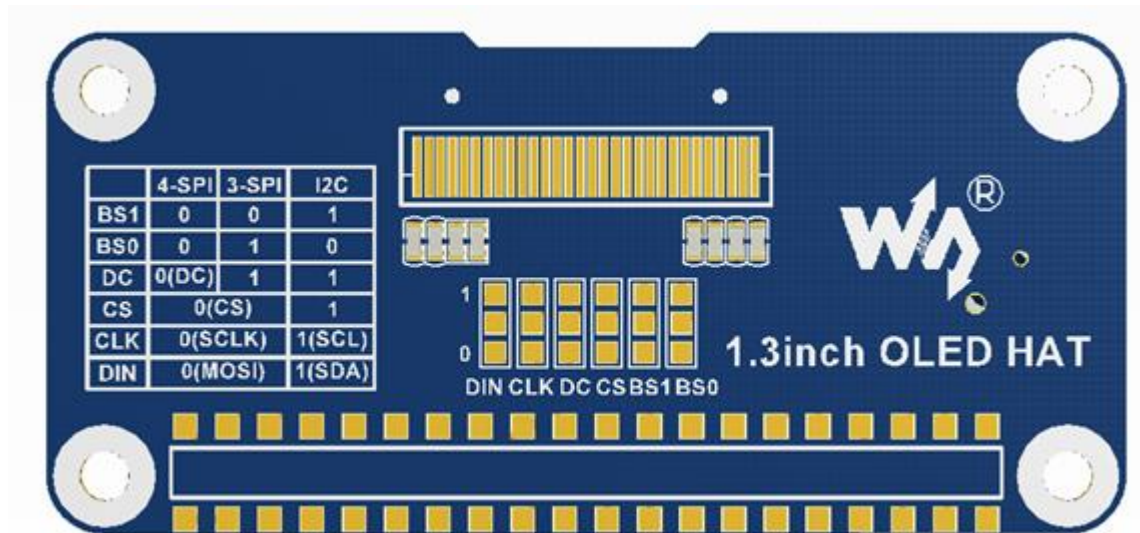
Driver:	SH1106
Interfaces:	3-wire SPI, 4-wire SPI, I2C
Resolution:	128 x 64
Diagonal size:	1.3inch
Dimensions:	65mm x 30mm
Display color:	blue
Viewing angle:	>160°
Operation temperature:	-20°C ~ 70°C
Storage temperature:	-30°C ~ 80°C

PIN CONFIGURATION

Functional pin	Raspberry Pi interface (BCM)	Description
KEY1	P21	Button1 GPIO
KEY2	P20	Button2 GPIO
KEY3	P16	Button3 GPIO
Joystick Up	P6	Joystick Up button
Joystick Down	P19	Joystick Down button
Joystick Left	P5	Joystick Left button
Joystick Right	P26	Joystick Right button
Joystick Press	P13	Joystick Press button
SCLK	P11/SCLK	SPI clock
MOSI	P10/MOSI	SPI data
SCL	P3/SCL1	I2C clock
SDA	P2/SDA1	I2C data
DC	P24	Data/command
CS	P8/CE0	Chip select
RST	P25	Reset

HARDWARE CONFIGURATION

The OLED module supports three drives Interface: 3-wire SPI, 4-wire SPI and I2C interface, and has 6 welding resistors/fuses in the back side for selecting the appropriate communication mode, as shown in picture:



By default, the Module is in the 4-wired SPI communication mode, i.e. the default value of BS0, BS1, DC, CS, CLK, DIN is 0 (Not all the 1 and 0 values stand for level. Some of them are stand for the ways resistors be welded up or down. Showed as hardware-specification table):

Note: the figure above is the map of hardware welding pads; the following table is the actual hardware connection.

	BS1/BS0	CS	D/C	DIN	CLK
3-wire SPI	0/1	CS	1	MOSI	SCLK
4-wire SPI	0/0		D/C		
I2C	1/0	0	1	SDA	SCL

Hardware configuration:

4-wired SPI (factory default): BS0 (0) and BS1 (0) are connected to the ground, CS (0) is connected to CE0 (activation pin) of Raspberry Pi, D/C (0) is connected to P24 of Raspberry Pi (data/command pin), DIN (0) is connected to MOSI of Raspberry Pi, CLK (0) is connected to SCLK of Raspberry Pi;

3-wired SPI: BS0 (1) is connected to VCC, BS1 (0) is connected to the ground, CS (0) is connected to CE0 of Raspberry Pi, D/C (1) is connected to the ground, DIN (0) is connected to MOSI of Raspberry Pi, CLK (0) is connected to SCLK of the Raspberry Pi;

I2C: BS0 (0) is connected to the ground, BS1 (1) is connected to VCC (3.3V), CS (1) is connected to the ground, D/C (1) is connected to the ground, DIN (1) is connected to SDA of Raspberry Pi,

CLK (1) is connected to SCL of Raspberry Pi; Using I2C interface: The level of DC controls the address of slave. Here it is connected to the ground, thus 7-bits of I2C address is: 0x3C.

DEMO CODE

We provide BCM2835, WiringPi and python programs for this module. They has commonly used display functions: drawing dot, line, rectangle, circle, setting sizes and line width; filling arias, and displaying English characters of 5 common fonts and other display's functions.

Following instructions are offered for you convenience

1. Initialize SPI function of the Raspberry Pi

sudo raspi-config

Select: **Advanced Options -> SPI -> yes**

Activate SPI hardware driver

2. Installation of function libraries

Before installation of Raspberry Pi libraries please read it carefully:

https://www.waveshare.com/wiki/Libraries_Installation_for_RPi

It is detailed description of WiringPi, bcm2835 and python installation.

3. Usage

BCM2835 and WiringPi program should be only copied into directory of Raspberry Pi (by samba or directly copy to the SD card). The following code are compiled directly to the user directory of Pi.

- 3.1 Usage of BCM2835

Run **ls** command as you can see below:

```
pi@raspberrypi:~/bcm2835 $ ls
bin  Fonts  Makefile  obj  oled_lin3
```

bin: contains ".o" files. We don't need to change it generally

Fonts: contains 5 commonly used fonts

Pic: contains pictures used for displaying. The resolution of pictures must be 128x128, otherwise they cannot be displayed properly. And the format of pictures must be BMP.

Obj: contains object files, like main.c, OLED_Driver.c, DEV_Config.c, OLED_GUI.c and their header files.

main.c: The mian function. What need to note is that even though there are OLED_ScanDir used to control the direction of scanning, you need not to change it. Because this module is designed for Raspberry Pi, and for compatibility, we don't recommend you to change it.

DEV_Config.c: definitions of Raspberry Pi's pins and modes; according to foregoing hardware configuration the changing of communication mode needs only ".h" file modification in macro-definitions **USE_SPI** and **USE_IIC**;

OLED_Driver.c: Drive code of OLED. Need not change generally.

OLED_GUI.c: graphical primitives: dot, line, bitmap, text; normally you only need to change **GUI_Show()** function, it redefines display function;

Makefile: This file contains compilation rules. If there are some changes in code, please run **make clean** to clean all the dependency file and executable files. Then execute **make** to compile the whole project and generate new executable files.

oled_lin3: this file is generated by command **make**.

3.2 WiringPi

Input **ls** command, now you can see following:

1. WiringPi operates by read/write the device files of Linux OS. and the bcm2835 is library function of Raspberry Pi's CPU, it operates registers directly. Thus, if you have used bcm2835 libraries firstly, the usage of WiringPi code will be failed. In this case, you just need to reboot the system and try again.
2. Due to the first difference, they underlying configuration are different. In DEV_Config.c, use wiringpiPi and the corresponding wiringPiSPI to provide underlay interfaces.

The program executed by command **sudo ./ftlcd_1in8** as well

- a) Python

3.3 Python

In case of python OLED driver library must be installed additionally. There are 2 libraries:

1. Adafruit_Python_SSD1306 driver supports SSD1306 only
2. Luma.oled driver supports SSD1306/SSD1322/SSD1325/SSD1331/SH1106

Thus we install luma.oled driver:

```
sudo apt-get install python-dev python-pip libfreetype6-dev libjpeg-dev
sudo -H pip install --upgrade pip
sudo apt-get purge python-pip
sudo -H pip install --upgrade luma.oled
```

Note: if at the installation of the Luma.oled driver you get red colored words alarming about an error, continue the installation please. It happens due to a problem of downloading from internet a driver library called Pillow.

Run **sudo python demo.py** to execute the program

3.4 Auto-run

Initialize autorun in Raspberry Pi by configuring code of /etc/rc.local file:

```
sudo vim /etc/rc.local
```

Before exit0 add:

sudo python /home/pi/python/demo.py &

Important: to place the program /home/pi/python/demo.py at the same director, you can input command **pwd** to get the path. And & character is necessary at the end of command line, otherwise probable need to reinstall the system (impossible terminate the process by pressing ctrl+c, impossible to login with pi user permission).