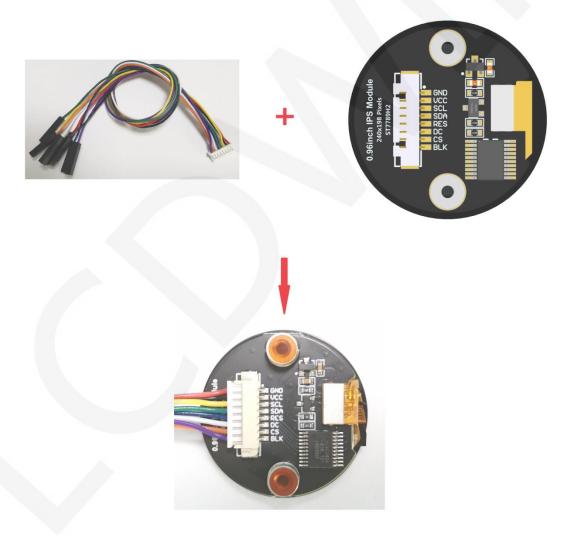
# 1. Introduction to Testing Platform

Development Board : ESP32-WROOM-32E devKit MCU : ESP32-32E module Frequency : 240MHz

# 2. Pin connection instructions

The display module is connected to the microcontroller using a 1.25mm spacing 8P DuPont cable with connectors. The module connection is shown in the following figure:



Picture1. Module wiring

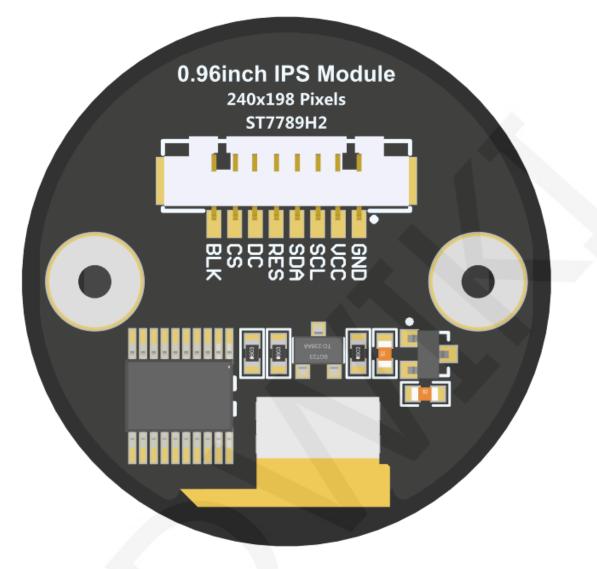


Figure 2 Module Back Pins

ESP32-32E Test Program Pin Direct Insertion Instructions						
NumberModule pinsCorresponding ESP32-32E development board wiring pins		ESP32-32E development	Remarks			
1	GND	GND	LCD Power ground			
2	VCC	5V/3.3V	LCD power positive(It is recommended to connect to 5V. When connected to 3.3V, the backlight brightness will be slightly dim)			
3	SCL	IO14	LCD SPI bus clock signal			
4	SDA	IO13	LCD SPI bus write data signal			

www.lcdwiki.com

5	RES	1027	LCD reset control signal, Low level reset
6	DC	102	LCD command / data selection control signal High level: data, low level: command
7	CS	1015	LCD selection control signal, Low level active
8	BLK	IO21	LCD backlight control signal (If you need control, please connect the pins. If you don't need control, you can skip it)

## 3. Demo Function Description

This sample program uses the ESP32 hardware HSPI bus, which is located in **Demo\_MSP0962\_MSP0963\_ESP32-WROOM-32E\_HSPI** directory, as shown in the following figure:

:件(F) 省	扁辑(E) 查看(V) 工具(T) 帮助(H)			
组织▼	包含到库中 • 共享 • 新建文件夹			
S ~	名称	修改日期	类型	大小
=	Example_01_Simple_test	2023/9/23 13:46	文件夹	
	Lxample_02_colligate_test	2023/9/23 13:46	文件夹	
	Example_03_display_graphics	2023/9/23 14:56	文件夹	
	Example_04_display_scroll	2023/9/23 15:32	文件夹	
	L Example_05_display_clock	2023/9/23 16:30	文件夹	

- A. Example\_01\_Simple\_Test is a screen brushing test program, which does not rely on any software library;
- B. Example\_ 02\_ colligate\_ Test is a comprehensive testing program that displays graphics, lines, and counts program runtime;
- C. Example\_03\_display\_Graphics is a graphic display testing program that displays various graphics;

- D. Example\_ 04\_ display\_ Scroll is a scrolling test program that displays text scrolling;
- E. Example\_05\_ display\_ Scroll is a scrolling test that displays text scrolling;

## 4. Demo Usage Instructions

#### Building Development Environment

For specific methods of building a development environment, please refer to the "Arduino\_development\_environment\_construction\_for-ESP32-EN" document in this directory.

### ♦ Installing software library

After the development environment is set up, the software library used by the sample program needs to be copied to the project library directory so that the sample program can be called. The software library is located in the **Install libraries** directory, as shown in the following figure:



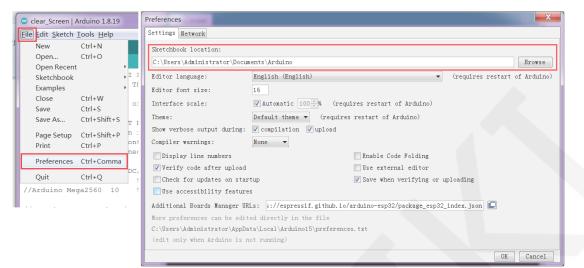
Among them:

TFT\_ eSPI is an Arduino graphics library for TFT-LCD LCD screens, supporting multiple platforms and LCD driver ICs

The software library have been configured and can be directly copied to the project library directory for use. The default path for the engineering library directory is

C:\Users\Administrator\Documents\Arduino\libraries. You can also change the project library directory: open the Arduino IDE software, click File ->Preferences, and

reset the Sketchbook location in the pop-up interface, as shown in the following figure:



If you do not want to use the already configured library, you can download the latest version of the library from Github at the following download address and then configured:

#### https://github.com/Bodmer/TFT eSPI

After the library download is completed, unzip it (for easy differentiation, rename the unzipped library folder, as shown in the Install libraries directory), and then copy it to the engineering library directory. Next, proceed with library configuration. The files that need to be replaced are located in the **Replaced files** directory, as shown in the following figure:

<b>GO</b> -	▶	files 🗸	· ∳ 搜索/
文件(F) 练	扁辑(E) 查看(V) 工具(T) 帮助(H)		
组织▼	包含到库中▼  共享▼  新建文件夹		
	名称	修改日期	类型
8	ST7789_Defines.h	2023/9/23 14:48	H 文件
6	ST7789_Init.h	2023/9/23 14:48	H 文件
	ST7789_Rotation.h	2023/9/23 12:08	H 文件
18 v	User_Setup.h	2023/9/23 15:20	H 文件

#### TFT\_ eSPI library configuration:

First rename the User\_Setup.h file which is in the top-level directory of the

TFT\_eSPI library of the engineering library directory to User\_ Setup\_ bak.h,then copy the User\_Setup.h file which is in the Replaced files directory to the top-level directory of the TFT\_eSPI library, As shown in the following figure:

	COLUMN AND AND	
$\bigcirc$	🗼 « Arduino → libraries → TFT_eSPI →	▼ <b>↓</b> 搜索 T ♀
文件(F) 编	辑(E) 查看(V) 工具(T) 帮助(H)	
组织 ▼	→ 打开 共享 电子邮件 新建文件夹	• • • •
	文档库 TFT_eSPI	排列方式: 文件夹 ▼
	名称	
<b>(3</b> )	TFT_config.h TFT_eSPI.cpp TFT_eSPI.h	
	User_Setup.h	=
	User_Setup_Select.h	

First, set the TFT in the engineering library directory\_ ESPI Library TFT\_ ST7789 in the Drivers directory\_ Init. h, ST7789\_ Rotation. h, ST7789\_ Define. h These three files are renamed as ST7789 respectively\_ Init. h\_ Bak. h, ST7789\_ Rotation\_ Bak. h, ST7789\_ Definitions\_ Bak. h, and then replace ST7789 in the Replaced files directory\_ Init. h, ST7789\_ Rotation. h, ST7789\_ Define. h three copies to TFT in the engineering library directory\_ ESPI Library TFT\_ Drivers directory, as shown in the following figure:

	扁辑(E) 查看(V) 工具(T) 帮助(H)	<u> かど7キュンノルーナー</u>		8==
组织 ▼		新建文件夹		
	文档库			排列方式: 文件夹 ▼
	TFT_Drivers			
	名称	修改日期	类型	大小
	ST7789_2_Init.h	2023/3/18 12:56	H 文件	1 KB
	ST7789_2_Rotation.h	2023/3/18 12:56	H 文件	3 KB
4	📄 ST7789_Defines.h	2023/9/23 14:48	H 文件	6 KB
≪ -	🗎 ST7789_Defines_bak.h	2023/3/18 12:56	H 文件	6 KB
	📄 ST7789_Init.h	2023/9/23 14:48	H文件	6 KB
<b>€</b> ∛	📄 ST7789_Init_bak.h	2023/3/18 12:56	H文件	6 КВ
	ST7789_Rotation.h	2023/9/23 12:08	H文件	3 KB
<b>N</b>	ST7789_Rotation_bak.h	2023/3/18 12:56	H文件	3 KB
۵	ST7796_Defines.h	2023/3/18 12:56	H文件	3 KB
4		2023/3/18 12:56	日文件	4 KB
4	ST7796 Rotation.h	2023/3/18 12:56	H文件	2 KB
22		111	0.20	2 10

### ♦ Compile and Run Programs

After the library installation is completed, the sample program can be compiled and

run as follows:

A. Plug the display module directly into the ESP32 development board, and

connect the development board to a PC to power on;

B. Open Any sample program in the

Demo\_MSP0962\_MSP0963\_ESP32-WROOM-32E\_HSPI directory, as shown

in the following figure (using the colligate test test program as an example):

Correction Control Con							
文件(F) 编辑(E) 查看(V) 工具(T) 帮助(H)							
组织▼ 包含到库中▼ 共享▼ 新建文件夹							
▲ 名称	修改日期	类型	大小				
Colligate_test.ino	2023/9/23 14:55	Arduino file	16 KB				

C. After opening the sample program, select the ESP32 device, as shown in the following figure:

www.lcdwiki.com

 colligate\_test | Arduino 1.8.19
 File Edit Sketch Tools Help Ctrl+T V 🖸 🗈 🖬 Auto Format Archive Sketch colligate\_test Fix Encoding & Reload // IMPORTANT // CONFIGURE Manage Libraries... Serial Monitor Ctrl+Shift+I Ctrl+Shift+M Serial Plotter Ctrl+Shift+L //This progr //of the lib Teensy 4 Security ESP32S3 Dev Module WiFi101 / WiFiNINA Firmware Updater ESP32C3 Dev Module //when using ESP32S2 Dev Module //when using //the SDA pi //if you don //other pins //pin usage Board: "ESP32 Dev Module" Upload Speed: "921600" Boards Manager.. ESP32 Dev Module ESP32-WROOM-DA Module ۲ Arduino AVR Boards CPU Frequency: "240MHz (WiFi/BT)" Flash Frequency: "80MHz" ESP32 Arduino ESP32 Wrover Module Teensyduino ESP32 PICO-D4

#### D. Configure ESP32 Flash, PSRAM, ports, etc. as shown in the following figure:

colligate_test   A		
File Edit Sketch To	ools Help	
	Auto Format Ctrl+T	
	Archive Sketch	
colligate_test	Fix Encoding & Reload	
// IMPORTANT	Manage Libraries Ctrl+Shift+I	
// CONFIGURE	Serial Monitor Ctrl+Shift+N	Λ
//This progr	Serial Plotter Ctrl+Shift+L	
//of the lib	Teensy 4 Security	
	WiFi101 / WiFiNINA Firmware Updater	
//when using		_
//the SDA pi	Board: "ESP32 Dev Module"	
//if you don	Upload Speed: "921600"	
//other pins	CPU Frequency: "240MHz (WiFi/BT)"	
//pin usage	Flash Frequency: "80MHz"	
//	Flash Mode: "QIO"	
//ESP32-WROO	Flash Size: "4MB (32Mb)"	
	Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"	
//Remember t	Core Debug Level: "None"	
/*****	PSRAM: "Disabled"	
* Gattention	Arduino Runs On: "Core 1"	
* Garrention	Events Run On: "Core 1"	
* THE PRESEN	Erase All Flash Before Sketch Upload: "Disabled"	
* WITH CODIN	JTAG Adapter: "Disabled"	
* TIME. AS A	Port	
* DIRECT, IN	Get Board Info	
* FROM THE C		
* CODING INF	Programmer: "Esptool"	

E. Click the upload button to compile and download the program, as shown in the

following figure:

🙁 colligate_test   Arduino 1.8.19					
<u>F</u> ile <u>E</u> dit <u>S</u> ketch <u>T</u> ools <u>H</u> elp					
colligate_test					
// IMPORTANT: LCDWIKI_SPI LIBRARY MUST BE SPECIFICALLY					
// CONFIGURED FOR EITHER THE TFT SHIELD OR THE BREAKOUT BOARD.					
//This program is a demo of how to use most of the functions					
//of the library with a supported display modules.					
//when using the BREAKOUT BOARD only and using these hardware spi lines to the L(					
//the SDA pin and SCK pin is defined by the system and can't be modified.					
//if you don't need to control the LED pin, you can set it to 3.3V and set the pir					
//other pins can be defined by youself, for example					
//pin usage as follow:					
// CS DC/RS RESET SDI/MOSI SCK SDO/MISO LED VCC GN					
//ESF32-WROOM-32E: 15 2 27 13 14 12 21 5V GN					
//Remember to set the pins to suit your display module!					
/*************************************					
* Gattention					

F. If the following prompt appears, it indicates that the program has been compiled

and downloaded successfully, and has already been run:

💿 GetChipID   Arduino 1.8.19
<u>F</u> ile <u>E</u> dit <u>S</u> ketch <u>T</u> ools <u>H</u> elp
GetChipID
/* The true FSD32 chin TD is essentially its MBC address
Done uploading.
<pre>Writing at 0x00024f8b (33 %) Writing at 0x0002a2bd (44 %) Writing at 0x0002f6ab (55 %) Writing at 0x000358af (66 %) Writing at 0x0003fe94 (77 %) Writing at 0x00045d9a (88 %) Writing at 0x0004b2ff (100 %) Wrote 261040 bytes (144184 compressed) at 0x00010000 in 2.5 seconds (e Hash of data verified.</pre>
Leaving Hard resetting via RTS pin
(1801t 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), QIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None, Disabled on COM47

G. If the display module displays content, it indicates that the program has run

successfully.