1. Introduction to Testing Platform

Development Board : ESP32-WROOM-32E devKit

MCU : ESP32-32E module

Frequency : 240MHz

2. Pin connection instructions

The module can be directly plugged into the ESP32-32E development board, as

shown in the following figure:



Figure 1: Module Inline ESP 32-32E Development Board

www.lcdwiki.com



Figure 2 Module Back Pins

ESP32-32E Test Program Pin Direct Insertion Instructions					
NumberModule pinsCorresponding ESP32-32E development board wiring pins		Corresponding ESP32-32E development board wiring pins	Remarks		
1	VCC	5V	LCD power positive		
2	GND	GND	LCD Power ground		
3	LCD_CS	IO15	LCD selection control signal, Low level active		
4	LCD_RST	1027	LCD reset control signal, Low level reset		
5	LCD_RS	102	LCD command / data selection control signal High level: data, low level: command		
6	SDI(MOSI)	1013	SPI bus write data signal(SD card and LCD screen used together)		
7	SCK	IO14	SPI bus clock signal(SD card and LCD screen used together)		
8	LED	IO21	LCD backlight control signal (If you need control, please connect the pins. If you don't need control, you can skip it)		
9	SDO(MISO)	1012	SPI bus read data signal (SD card and LCD screen used together)		

			Capacitive touch screen IIC bus clock	
10	CTP_SCL	1025	signal (modules without touch screens	
			do not need to be connected)	
			Capacitor touch screen reset control	
11		1022	signal, low-level reset (modules without	
11		1055	touch screens do not need to be	
			connected)	
			Capacitive touch screen IIC bus data	
12	CTP_SDA	IO32	signal (modules without touch screens	
			do not need to be connected)	
			Capacitor touch screen IIC bus touch	
			interrupt signal, when generating touch,	
13	CTP_INT	1039	input low level to the main control	
			(modules without touch screens do not	
			need to be connected)	
			SD card selection control signal, low	
14	SD_CS	1022	level active (without SD card function	
			can be disconnected)	

3. Demo Function Description

This sample program uses the ESP32 hardware HSPI bus, which is located in **Demo_MSP4030_MSP4031_ESP32-WROOM-32E_HSPI** directory, as shown in the following figure:

	z 🍌 ≪ Demo_ESP32 ▸ Demo_MSP4030_M	SP4031_ESP32-WROO	M-32E_HSPI 🕨	- € γ
文件(F) 约	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			
1. *	名称	修改日期	类型	大小
8	Example_01_Simple_test	2023/11/22 10:28	文件夹	
	Example_02_colligate_test	2023/11/22 10:28	文件夹	
	Example_03_display_graphics	2023/11/22 10:28	文件夹	
	Example_04_display_scroll	2023/11/22 10:28	文件夹	
	👢 Example_05_show_SD_bmp_picture	2023/11/22 10:28	文件夹	
	Example_06_show_SD_jpg_picture	2023/11/22 10:28	文件夹	
	Example_07_display_phonecall	2023/11/22 10:28	文件夹	
~	👢 Example_08_touch_pen	2023/11/22 10:28	文件夹	
	Example_09_LVGL_Demos	2023/11/22 10:28	文件夹	

♦ Description of sample program content

- 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_ show_ SD_ bmp_ Picture is a BMP image display program that displays BMP format images within SD;
- F、Example_06_ show_ SD_ jpg_ Picture is a JPG image display program that displays images in jpg format within SD;
- G. Example_07_display_Phonecall is a touch testing program for telephone dialing, which simulates the dialing function through touch;
- H. Example_08_ touch_ Pen is a touch stroke test program that draws on the LCD screen through touch;
- K. Example_ 09_ LVGL_ Demos is an LVGL example display program that allows you to experience the powerful UI design features of LVGL. The bin file for this example has been extracted and can be directly burned using the corresponding tool.

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:

FT6336 arduino is the driver of FT6336 capacitive touch IC

LVgl is LVGL GUI graphics software library

TFT_ ESPI is an Arduino graphics library for TFT-LCD LCD screens,

supporting multiple platforms and LCD driver ICs

TJpg_ Decoder is a JPG format image decoding library for the Arduino platform These 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:

LCDWIKI

💿 clear_Screen Arduino 1.8.19	Preferences
<u>File</u> dit <u>S</u> ketch <u>T</u> ools <u>H</u> elp	Settings Network
New Ctrl+N Open Ctrl+O Open Recent	Sketchbook location: C: Users \Administrator\Documents \Arduino Editor language Evelight (Evelight)
Examples Ti Close Ctrl+W Save Ctrl+S Save As Ctrl+Shift+S m, t	Editor font size: 16 Interface scale: Image and the scale of the
Page Setup Ctrl+Shift+P n : Print Ctrl+P ont net	Show verbose output during: 📝 compilation 🖓 upload Compiler warnings: None 💌 Discolar line number:
Preferences Ctrl+Comma	Verify code after unload
Quit Ctrl+Q (//Arduino Mega2560 10 (Check for updates on startup Use accessibility features
I	Additional Boards Wanager URLs: 5://espressif.github.io/arduino-esp32/package_esp32_index.json More preferences can be edited directly in the file C:\Users\Administrator\AppData\Local\Arduinol5\preferences.txt (edit only when Arduino is not running)

If you do not want to use the already configured library, you can download the latest version of the library (excluding FT6336 arduino) from Github at the following download address and then configured:

lvgl: https://github.com/lvgl/lvgl/tree/release/v8.3 (Only V8. x version can be used,

V9. x version cannot be used)

TFT_eSPI: https://github.com/Bodmer/TFT_eSPI

TJpg_Decoder: <u>https://github.com/Bodmer/TJpg_Decoder</u>

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:

Water -			x
← ← ▲ « Demo_ESP32 ► Replaced files	▼ 🍫 搜索 Repla	aced files	Q
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			0
▲ 名称 ▲	修改日期	类型	
🔹 📄 lv_conf.h	2023/4/4 9:34	H 文件	
User_Setup.h	2023/6/19 11:36	H文件	
₩ + +			Þ

www.lcdwiki.com

LVGL library configuration:

Copy the **lv_conf.h** file which is in the **Replace files** directory to the top-level directory of the lvgl library in the engineering library directory,As shown in the following figure:



Open the **lv_conf_internal.h** file which is in the Lvgl library **src** directory under the engineering library directory,As shown in the following figure:

0-	• 📙 « 文档 → Arduino → libraries → lvgl → src →	▼ ⁴↑ ∄	叟索 src 👂
文件(F) 练	編辑(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	• == v	0
	<mark>文档库</mark> src	排列方式: 文	件夹 ▼
	名称		*
2	👢 core		
	🗎 lvgl.h		
	lv_conf_kconfig.h		
	lv_conf_internal.h		=
	📄 lv_api_map.h		
	< III		•
	lv_conf_internal.h 修改日期: 2023/4/4 9:35 H文件 大小: 73.7 KB		

completed.

After opening the file, modify the content of line 41 as shown in the following figure

(from "../../lv_conf. h" to "../lv_conf. h"), and save after the modifications are

```
/*If lv_conf.h is not skipped include it*/
#ifndef LV_CONF_SKIP
    #ifdef LV CONF PATH
                                                    /*If there is a path defined for lv conf.h u
        #define __LV_TO_STR_AUX(x) #x
        #define __LV_T0_STR(x) __LV_T0_STR_AUX(x)
       #include __LV_TO_STR(LV_CONF_PATH)
#undef __LV_TO_STR_AUX
#undef __LV_TO_STR
    #elif defined(LV CONF INCLUDE SIMPLE)
                                                 /*Or simply include lv_conf.h is enabled*/
        #include "lv conf.h"
    #else
        #include "../lv conf.h"
                                                 /*Else assume lv conf.h is next to the lvgl for
    #endit
    #if !defined(LV_CONF_H) && !defined(LV_CONF_SUPPRESS_DEFINE_CHECK)
        /* #include will sometimes silently fail when __has_include is used */
        /* https://gcc.gnu.org/bugzilla/show_bug.cgi?id=80753 */
        #pragma message("Possible failure to include lv_conf.h, please read the comment in th:
    #endif
#endif
```

Copy the examples and demos directories under the engineering library directory to the src directory under the lvgl library. These two directories are shown in the

		_ 🗆	X
	▼ 49	搜索	lvgl 🔎
文件(F) 编辑(E) 查看(V) 工具(T) 帮助(H)			
组织▼ 共享▼ 新建文件夹			0
文档库 Ivgl	排列方式:	文件夹	•
and E 名称 ■ L src			*
env support			
docs			Ŧ
27 个对象			4

following figure in the lvgl library:

The directory status after copying:

\bigcirc	📕 « 文档 🕨 Arduino 🕨 librari	ies ▶ lvgl ▶ src ▶
文件(F) 练	扁辑(E) 查看(V) 工具(T) 帮助(H)	
组织▼	共享▼ 新建文件夹	III 🔹 🗖 🔞
	<mark>文档库</mark> src	排列方式: 文件夹 ▼
	名称 人 extra	
	damos	E
1	Core	
	< < Ⅲ	
	13 个对象	

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:



♦ 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_MSP4030_MSP4031_ESP32-WROOM-32E_HSPI directory, as shown

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

Correction of the second secon	Room-32e_HSPI 🕨 e	xample_02_colligate	e_test	• •
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	2023/3/31 11:03	Arduino file	16 KB	

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

following figure:

🙁 colligate_test Ar	duino 1.8.19		
File Edit Sketch To	pols Help		
colligate_test	Auto Format Archive Sketch Fix Encoding & Reload	Ctrl+T	
// IMPORTANT // CONFIGURE	Manage Libraries Serial Monitor Serial Plotter	Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	
//of the lib	Teensy 4 Security WiFi101 / WiFiNINA Firmware Updater	ESP32S3 Dev Module ESP32S3 Dev Module	
//the SDA pi //if you don //other pins //pin usage	Board: "ESP32 Dev Module" Upload Speed: "921600" CPU Frequency: "240MHz (WiFi/BT)" Flash Frequency: "80MHz"	Boards Manager ESP3252 Dev Module Arduino AVR Boards ESP32 Dev Module ESP32 Arduino ESP32 Wrover Module Teensyduino ESP32 PICO-D4	dule

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

	\sim	\ A /		Z I -
		 \/\/	1.14	< I .
- L	_0	v v		<u> </u>

colligate_test Are	duino 1.8.19	
File Edit Sketch To	ols Help	
colligate_test	Auto Format Archive Sketch Fix Encoding & Reload	Ctrl+T
// IMPORTANT // CONFIGURE	Manage Libraries Serial Monitor	Ctrl+Shift+I Ctrl+Shift+M
//This progr //of the lib	Teensy 4 Security WiFi101 / WiFiNINA Firmware Updater	Ctri+Shitt+L
//when using	Board: "ESD32 Dev Module"	L
//the SDA pi	Unload Speed: "921600"	
//other pins	CPLL Erequency: "240MHz (WiEi/BT)"	· · · · ·
//pin_usage	Elech Frequency: "80MHz"	
//	Elash Moder "OIO"	, G
//ESP32-WROO	Flash Size: "4MB (32Mb)"	, G
//Remember t	Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)" Core Debug Level: "None")
/*******	PSRAM: "Disabled"	• • • •
* @attention	Arduino Runs On: "Core 1"	•
*	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"	•
	Draw Deetleeden	

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

following figure:

👓 colligate_test Arduino 1.8.19					
<u>File Edit Sketch Tools 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					
<pre>//of the library with a supported display modules.</pre>					
//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					
//ESP32-WROOM-32E: 15 2 27 13 14 12 21 5V GP					
//Remember to set the pins to suit your display module!					
/************************************					
* @attention					

www.lcdwiki.com

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

and downloaded successfully, and has already been run:



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

♦ LVGL example bin file burning

Due to the long compilation time of the LVGL sample program, the compiled bin

file has been extracted and can be directly burned using the flash download tool.

Bin file located in

Demo_ESP32\Flash_Download_LVGL_Demos\bin directory, as shown in the following figure:

					. 🗆 🗙		
Correction of the second secon					搜索 bin 👂		
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 副社 ▼ □ 2							
م	名称	修改日期	类型		大小		
8	😻 lv_demo_benchmark.bin	2023/5/19 17:12	BIN 文件		643 KB		
32	💖 lv_demo_keypad_encoder.bin	2023/5/19 17:15	BIN 文件		553 KB		
~	💖 lv_demo_music.bin	2023/5/19 17:17	BIN 文件		942 KB		
	💖 lv_demo_stress.bin	2023/5/19 17:19	BIN 文件		557 KB		
	😻 lv_demo_widgets.bin	2023/5/19 17:09	BIN 文件		746 KB		
▼ ∢		111			•		
	5 个对象						

Using the flash_download_tool can burn in the

Demo_ESP32\Flash_Download_LVGL_Demos directory, as shown in the following

figure:

			X
← ← ↓ ≪ Demo_ESP32 ► Flash_Download_LV0	GL_Demos 🕨	▼ 4 <i>投索 F</i>	l 🔎
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		· · · · ·	0
名称 名称	修改日期	类型	A
😑 🗼 bin	2023/5/3 17:38	文件夹	=
liash_download_tool_3.9.4	2023/5/6 10:59	文件夹	-