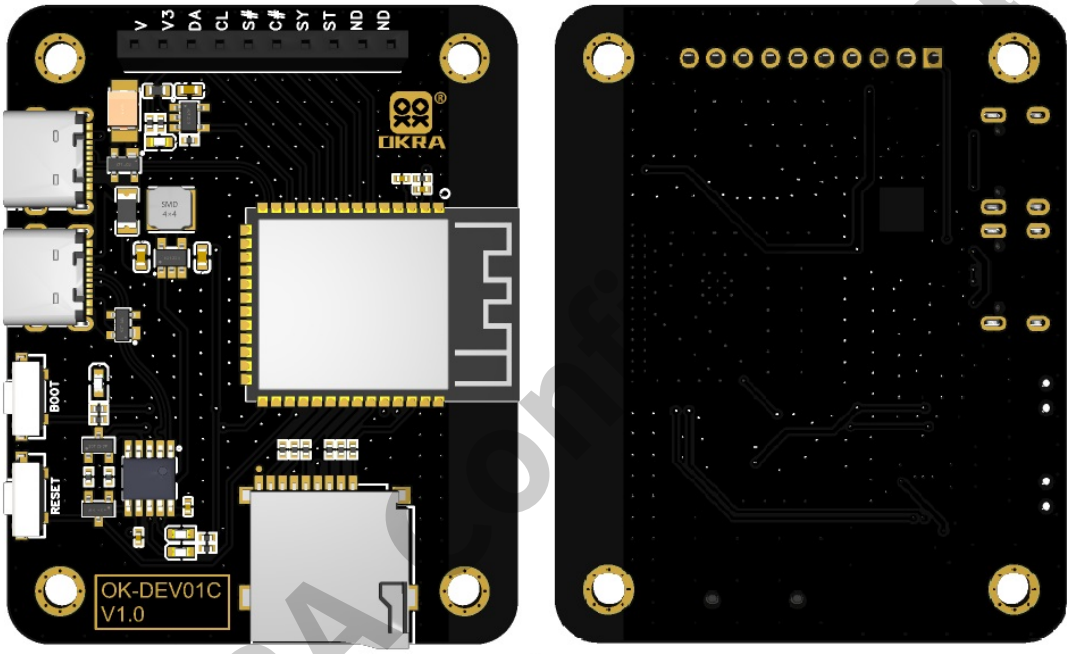
OK-DEV01C Specifications/ User Manual

Foreword

- OK-DEV01C is a development board designed and developed by [Shanghai OKRA Technologies CO., LTD.] (https://www.okratechnologies.com) specifically for small-size SPI interface E-paper displays.



Product Features

- OK-DEV01C is an upgraded version of OK-DEV01A (Bluetooth version EVK), with stronger performance and more convenient development. It is compatible with all the functions of OK-DEV01A.
- Lead out the EPD drive interface, with 5V, 3.3V, compatible with our OK-TR01A, OK--TR02A, OK-TR03A adapter boards.
- Equipped with ESP32S3-WROOM-1-N16R8 module.
- Low-power consumption product design, supports independent control of screen power supply (3.3V only. TR01A needs to be switched to 3.3V power supply).
- External TF card, directly load pictures (jpg format) through TF card for display.
- Support high-precision dithering algorithm.

- Onboard Type-C to serial port debugging interface, allows easy firmware burning and debugging.
- Provides basic reference code based on the Arduino environment to fully implement screen driving.
- It also supports ESPRESSIF's official ESP-IDF and ESPRESSIF-IDE.

ESP32S3-WROOM-1-N16R8

- WIFI+BLE5.0
- Built-in ESP32-S3 series chip, Xtensa dual-core 32-bit LX7 processor, supports single-precision floating-point unit, and clock frequency up to 240MHz.
- Built-in 384KB ROM
- Built-in 512KB SRAM
- Built-in 16KB RTC SRAM
- 8MB PSRAM in package
- External 16MB SPI FLASH
  et the full ESP32S3-WROOM-1-N16R8 datasheet.

Interface and buttons

- OK-DEV01C has a Boot button and a Reset button on board. Press and hold the Boot button, press the Reset button once, and then release the Boot button to manually enter the firmware burning mode. When using Espressif's official burning tool or Arduino IDE, you can automatically burn the firmware without pressing any buttons manually.
- Lead out the TF card interface. ***Not hot-swappable. *** Used to load Slide Show images.
- Battery interface: 1.25 pitch ultra-thin pin header 2P.
- Type-C interface: debug x1, data x1.
- Adapter board interface:

Pinout	IO
5V	Always connected
3.3V	Controllable 3.3V power supply (IO8 control, high level is effective)
SDA/MOSI	IO18

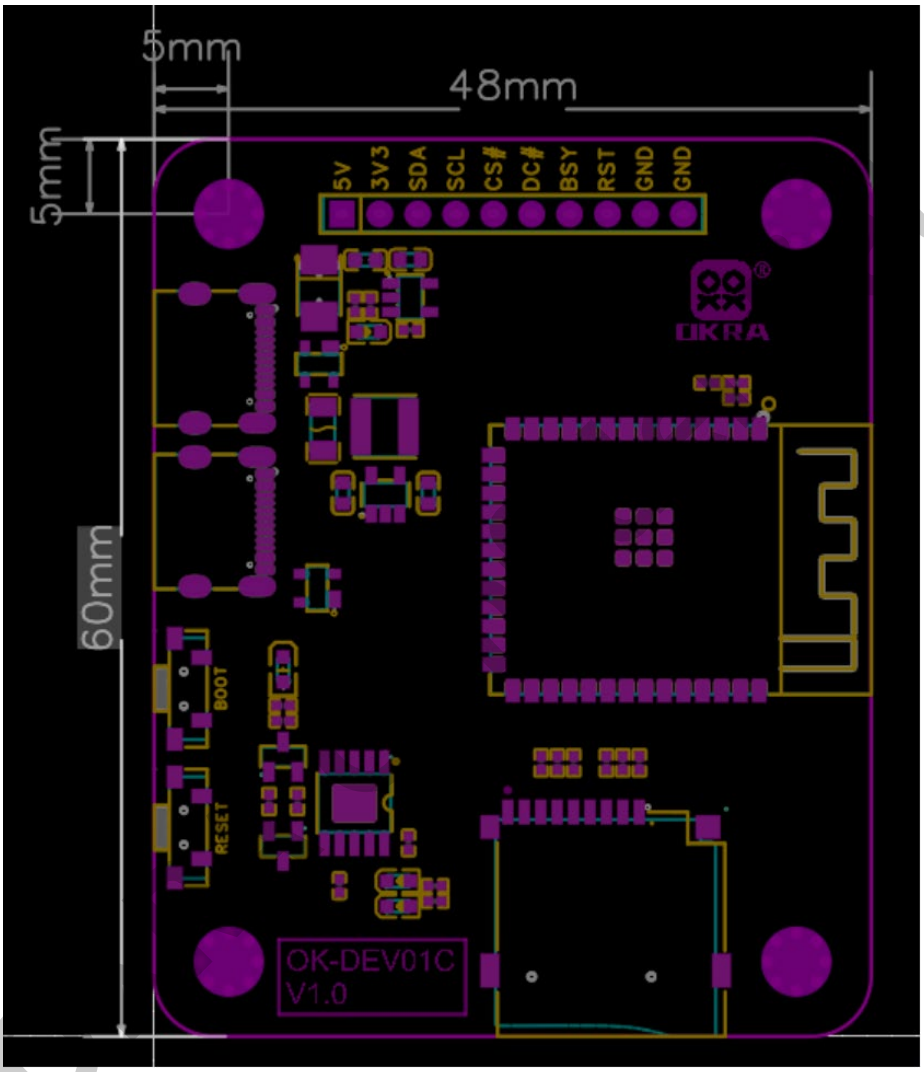
Pinout	IO
SCL/SCLK	IO17
CS#	IO16
DC#	IO15
BUSY#	IO7
RST#	IO6
GND	Ground
GND	Ground

Power supply

- USB-C port for power supply
- USB port input requirement: 5V@500mA.

Outline Dimensions

- The dimensions of OK-DEV01C are 60x48mm, the board thickness is 1.6mm, and the fixing hole specification is M3.



Design and quality

- Plug hole, immersion gold process
- All black PCB

Shipping list

- OK-DEV01C motherboard + OK-TR01A adapter board (with 7.3-inch Spectra 6 firmware burned).
- 8GB TF card (Demo image has been copied)
- USB A-C cable 1

Firmware Support(Need to be used with OK-TR01A V2.0 adapter board)

SlideShow firmware —— Functional Demonstration

- Default factory firmware, optional 7.3-inch or 4-inch Spectra 6 firmware. Use TF card to realize screen carousel function
- **The image file has to be in baseline-encoded jpg format (Windows Paint saves the image in jpg format with baseline encoding. If using Photoshop, you need to manually select baseline encoding)**
- **The image resolution has to correspond to the screen resolution. 7.3-inch Spectra6 EPD is 800x480, 4-inch Spectra 6 EPD is 400x600.**
- The default refresh interval is 30 seconds.
- **TF card is not hot-swappable.**
- Provide binary files ready for flashing.

Basic firmware —— Secondary Development

- Provide reference code: 7.3" Spectra 6, 4" Spectra 6.
- Complete power-on and power-off timing control of the EPD.
- Simple and clear driver architecture, convenient for secondary development.

MQTT firmware(to be complete) —— Project Cooperation

- Bluetooth network configuration.
- WIFI access to the Internet.

- MQTT Broker and username and password are configurable, and the communication protocol is simple and secure.

USB-CDC firmware —— Project Cooperation

- Need to be used with a host computer.
- Supports image transmission and corresponding control functions via USB virtual serial port.
- Image interface, Dither, and data packaging are all completed on the host computer.
- Supports partial refresh.(to be complete)
- This firmware supports direct display of images after transmission, or storage in the built-in Flash of the ESP32S3 module. After unplugging the USB cable, the images stored in the Flash will be automatically rotated.
- *Provide host computer executable files for debugging and testing.
- *Provide communication protocol.

Tips for E Ink Spectra 6 Series E-paper display

- The Spectra 6 series E-paper display continues the color performance of E Ink Gallery and has improved the contrast and saturation. However, if you want the picture to have more realistic color performance on the Spectra 6 EPD, you need to enhance the original image.
- The control of power-on and power-off timing is very important for the EPD. Wrong power-on and power-off timing can easily lead to abnormal image or even display damage. Therefore, when using the source code of our basic firmware for secondary development, please do not modify the Pmic and interface driver code at will unless necessary to prevent abnormal power-on and power-off timing.
- **Please do not disconnect the power supply at will during the screen refresh process** (When the PMIC indicator light is on), otherwise it is very easy to cause damage to the EPD. When the power is off, please make sure that only 2 lights on the board are on (main power indicator light, 3.3V power indicator light).

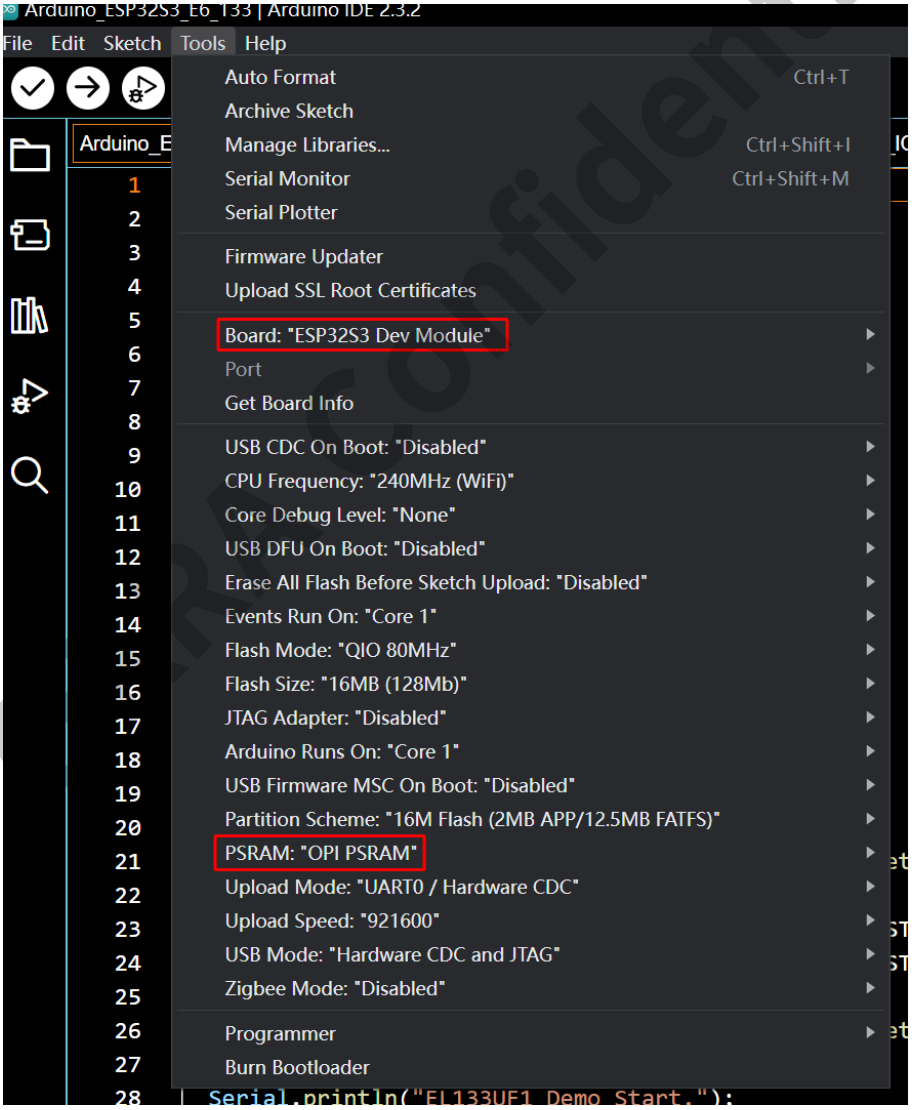
Development environment construction

- Install the latest version of Arduino.
- In the Arduino File->Preferences tab, in the Additional boards manager URLs field, fill in:

  https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

  Click the OK button and wait for the update to complete.

- Download the source code of the basic firmware, unzip it, double-click the Arduino_ESP32S3_E6_315.ino file, and then configure the development board.
- Tools->Boards->ESP32S3 Dev Module
- Tools->PSRAM->"OPI PSRAM"



Project Creation

- Please refer to the Arduino related documentation.

Project compilation

- Please refer to the Arduino related documentation.

Firmware burning

- Please refer to the Arduino related documentation.