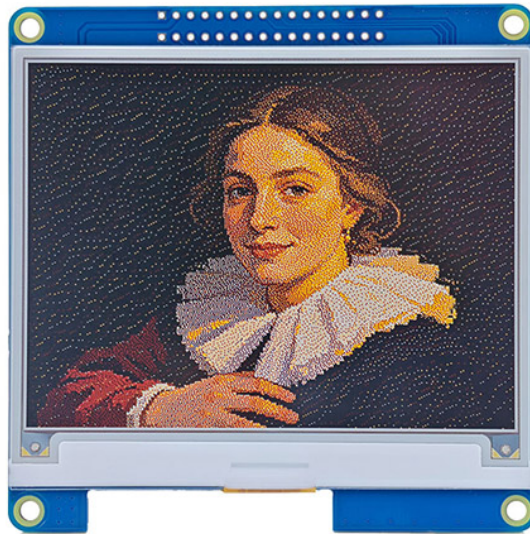


Manual

4.2 inch Photo Board(4C)



Open-source development board · Supports secondary development

Document Version: **v1.0** August 2026



Table of Contents

1. Overview
 2. Key Features
 3. Onboard Resources
 4. Dimensions
 5. Arduino Development Environment Setup
 6. Compile and Flash
 7. Quick Start
 8. Web UI Description
 9. Playback Mode & Album Management
 10. TF Card Mirror Backup
 11. Frequently Asked Questions
 12. Notes & Version Info
-

1 Overview

The **4.2 inch Photo Board(4C)** is an open-source 4-color E-Paper development board based on the **ESP32-WROOM-32E** MCU. It carries an onboard 4.2-inch black / white / red / yellow E-Paper display at 400×300 resolution, suitable for desktop photo frames, dynamic labels, low-power information displays, and similar applications.

Designed for makers, educators, and secondary developers, the board integrates a **USB-C port**, **CH340C USB-to-serial bridge**, **TF card slot**, **three physical buttons**, dual I/O expansion headers, plus the **BOOT** and **RST** buttons — flashing and debugging require no extra tools.

The companion sketch is built on the Arduino framework and provides a full photo-album manager plus a web configuration UI: upload an image through the browser and the board displays it. Auto-switching follows a configurable timer, with TF card mirror backup and one-click clear-all.

2 Key Features

- **MCU**: ESP32-WROOM-32E, 240 MHz dual-core, integrated Wi-Fi 4 + BLE.
- **Display**: 4.2-inch 4-color E-Paper (black / white / red / yellow), 400×300 resolution, FPC-24P interface, no backlight, easy on the eyes.

- **Storage:** Internal LittleFS (~1 MB available) as primary album storage; TF card can serve as a mirror backup.
- **Buttons:** Previous / Mode / Next three-button control for manual page change and mode switch.
- **Indicators:** Red PWR power LED + blue USR user LED.
- **Interfaces:** USB-C (power + data / flashing), TF card slot, P3/P5 and P1 dual-row I/O expansion pins.
- **Wireless:** Defaults to AP mode hotspot (SSID: `ePaperDisplay`); can be switched to STA mode to join a home router.
- **Open source:** Schematics, firmware source code and Web UI are all open; secondary development is supported.

3 Onboard Resources

The figure below labels every connector and functional block on the board. Use it together with the physical unit for reference.

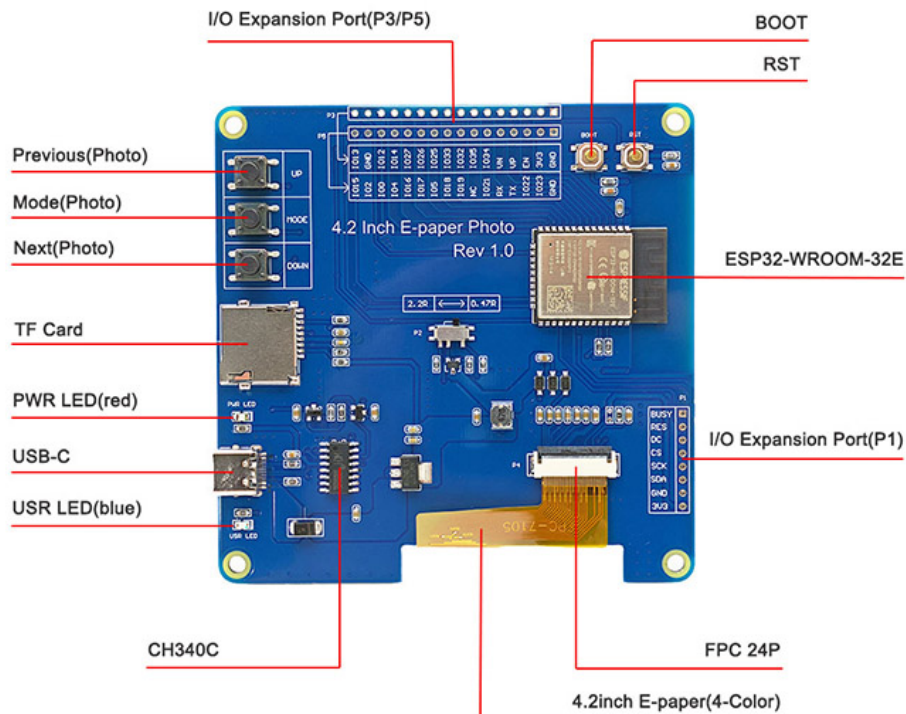


Fig. 2 4.2 inch Photo Board(4C) onboard resource layout



3.1 Power & Flashing Area

- **USB-C:** For 5 V supply, serial communication and firmware flashing. Use a data-capable Type-C cable.
- **CH340C:** USB-to-UART bridge. No external USB-TTL converter is required for ESP32 flashing.
- **PWR LED (red):** Power indicator; stays lit when power is on.
- **USR LED (blue):** User-programmable indicator.

3.2 Buttons

- **Previous (Photo):** Previous photo.
- **Mode (Photo):** Switch playback mode (Loop / Single / Random).
- **Next (Photo):** Next photo.
- **BOOT:** Enters ESP32 download mode (usually not needed; if flashing fails, hold BOOT and press RST to recover).
- **RST:** Reset / reboot the board.

3.3 Storage & Display

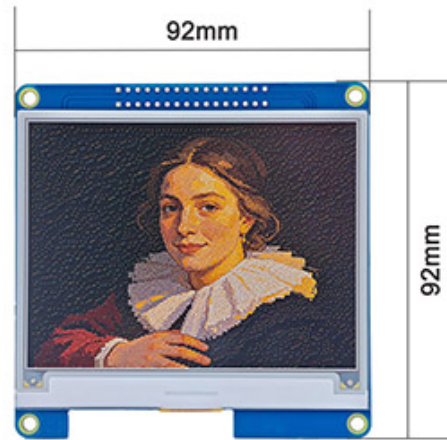
- **TF Card:** Standard Micro SD slot, FAT32 format. Once inserted it automatically becomes the album's mirror backup; removal does not affect the album.
- **FPC 24P:** 24-pin FFC connector to the 4.2-inch 4-color E-Paper panel. Do not hot-plug the cable.

3.4 Expansion Interfaces

- **I/O Expansion Port (P3 / P5):** Dual-row pins exposing ESP32 GPIO / 3V3 / GND for secondary-development peripherals.
- **I/O Expansion Port (P1):** E-paper expansion port with 3V3 / GND and several GPIOs.

4 Dimensions

The board follows the standard 4.2-inch E-Paper footprint; the PCB outline is square for easy integration into enclosures or photo frames.



Fi

g. 3 4.2 inch Photo Board(4C) dimensions (unit: mm)

Mechanical Dimensions	
Board outline	92 mm × 92 mm (PCB perimeter)
Active display area	~84.8 mm × 63.6 mm (400 × 300 pixels)
Mounting holes	Ø3 mm at four corners, M3 compatible
Board thickness	~1.6 mm (PCB) + component height

5 Arduino Development Environment Setup

Prerequisites Recommended: Arduino IDE 2.x (faster, more stable) or Arduino IDE 1.8.19. This manual uses **Arduino IDE 2.x**.

5.1 Install Arduino IDE

1. Visit the Arduino website <https://www.arduino.cc/en/software> and download the installer for your operating system.
2. On first launch, open **File** → **Preferences** and (optionally) set the editor language to your preference.



3. Keep the IDE connected to the Internet so it can download the ESP32 board package and dependencies.

5.2 Add ESP32 Board Support

1. Open **File** → **Preferences** and locate **Additional boards manager URLs**.
2. Paste the following URL (you can keep existing entries; multiple URLs go on separate lines):
`https://espressif.github.io/arduino-esp32/package\_esp32\_index.json`
3. Open **Tools** → **Board** → **Boards Manager**.
4. Search for **esp32**, find **esp32 by Espressif Systems**, choose version **2.0.14** or newer, and click **Install**. Close the manager when done.

5.3 Install USB Driver (CH340C)

The board uses a **CH340C** USB-to-serial chip. Make sure the driver is installed on first use under Windows:

- Windows 10 / 11 typically ships with the CH340 driver. Once the board is plugged in, **Device Manager** → **Ports (COM & LPT)** will show **USB-SERIAL CH340 (COMx)**.
- If not recognized, download the CH340 Windows driver and run it as administrator. macOS / Linux usually work without an extra driver.

5.4 Required Libraries

This sketch is based on the Arduino framework and uses only the libraries listed below (all bundled with the **esp32** board package; no extra installation needed):

- **WiFi.h** — bundled with ESP32 package
- **WebServer.h** — bundled with ESP32 package
- **LittleFS.h** — bundled with ESP32 package
- **SD.h** — bundled with ESP32 package
- **DNSServer.h** — bundled with ESP32 package

6 Compile and Flash

Firmware Preparation Extract the project archive. The folder should contain

`photo_4_2.ino`, `webui.h`, `Display_EPD_W21.cpp` / `.h` and `Display_EPD_W21_spi.cpp` / `.h` (6 files in total). Open `photo_4_2.ino` directly with the Arduino IDE; `webui.h` will be opened as an additional tab in the same editor window.



6.1 Board and Port Settings

1. **Tools** → **Board**: select **ESP32 Dev Module** (or **ESP32 Wrover Module**).
2. **Tools** → **Port**: select the recognized **USB-SERIAL CH340 (COMx)**.
3. Other common settings can stay at their defaults: Flash Size **4MB (32Mb)**, Partition Scheme **Default 4MB with spiffs**, Upload Speed **921600**.

6.2 Compile

Click the **✓ Verify** button (or **Sketch** → **Verify/Compile**). The first compile takes a while (the ESP32 toolchain needs heavy header preprocessing); on success the bottom console shows **Done compiling**, along with the firmware size.

6.3 Flash

1. Click the → **Upload** button (or **Sketch** → **Upload**).
2. The IDE invokes **esptool** automatically. The serial console shows **Connecting...** → **Writing at 0x... → Hash of data verified**.
3. After flashing the board auto-resets and the screen shows the 4-color hint image.

Flash failed? If the console stays on **Connecting...** too long: hold **BOOT**, click **Upload** in the IDE, and release **BOOT** once you see **Writing....**

7 Quick Start

7.1 Connect Power

Use a standard USB-C data cable to connect the board to a computer or charger. On power-up the screen shows the 4-color hint image.

7.2 Connect to the Board

The board starts Wi-Fi in **AP mode** by default:

- **SSID**: **ePaperDisplay**
- **Password**: **88888888**

After connecting a phone or computer to this hotspot, open the browser at <http://192.168.4.1> to enter the configuration page.



7.3 Switch to STA Mode (Join Home Wi-Fi)

To make the board join your home router directly, fill in the SSID and password under the **Settings** card of the Web UI and save. The board reboots and joins the router in STA mode. Open the browser at the IP it obtained from the router to access it on your local network (recommended for advanced users).

7.4 Upload Your First Photo

1. Open <http://192.168.4.1> (AP mode) or the board's LAN IP.
2. In the **Upload** card click **Choose File** and pick a JPG / PNG image from your computer (recommended aspect ratio $\approx 4:3$).
3. The browser quantizes the image locally into a 400×300 4-color image (black / white / red / yellow) and shows a live preview.
4. Click **Upload**; once upload finishes, the screen refreshes within ~21 seconds.

8 Web UI Description

The Web UI is divided into the following cards:

Card	Function
Status	Shows TF card state, photo count, internal free space, and whether a sync is in progress.
Current Screen	Live render of what the E-Paper is currently displaying, for remote preview.
Upload	Pick a local image, quantize it, and upload to the album.
Playback mode	Set playback mode (Loop / Single / Random) and the auto-switch interval.
Photo Library	Lists all uploaded photos; click Show to display a photo immediately, or Delete to remove it.
Delete all	Clears the internal album and the TF mirror at the same time (irreversible; double confirmation required).



9 Playback Mode & Album Management

9.1 Playback Modes

Choose a mode under the **Playback mode** card:

- **Loop**: cycles through photos in album order, looping back at the end.
- **Single photo**: stays on the same photo; no auto-switching.
- **Random**: switches to a random photo every configured interval.

Switch interval: time between two photos, in seconds. Default is 60 s.

9.2 Physical Buttons

The onboard buttons let you operate the board without a network:

- **Prev / Next**: previous / next photo.
- **Mode**: cycle through playback modes.

9.3 Delete & Clear

- In **Photo Library**, click **Delete** next to a photo to remove just that photo (the TF mirror is removed too).
- Click **Delete all** in the top-right to clear the internal album **and** the TF mirror; this is irreversible and requires double confirmation.

10 TF Card Mirror Backup

The board uses internal LittleFS as the **primary album storage** and the TF card as the **mirror backup**:

10.1 Two-way Sync

- **Internal** → **TF**: ~2 s after insertion, sync starts automatically, copying one photo per pass (~30 KB). The web UI stays responsive.
- **TF** → **Internal**: any legal `img_*.bin` files on the TF that are missing internally are also restored (provided there is enough free space).

10.2 Incremental Updates

While the TF is mounted:



- Uploading a new photo → automatically double-writes to TF.
- Deleting a photo → automatically double-deletes on TF.
- No manual intervention required.

10.3 Removal Is Safe

The TF card can be pulled at any time — the album keeps working from internal storage. Re-insert to rebuild the mirror automatically.

11 Frequently Asked Questions

Q1 Screen shows nothing after flashing?

Check that the FPC 24P cable is firmly seated and oriented correctly (typically metal side up); make sure the supply is stable (recommend a 5 V / 1 A or better USB-C source).

Q2 Cannot find the USB-SERIAL CH340 port?

Try a different data cable (some cheap cables only charge); install the CH340 driver manually under Windows; hold BOOT while powering on to force the ESP32 into download mode.

Q3 Web page won't open (192.168.4.1 no response)?

Confirm the phone / computer has joined the board's AP hotspot (SSID: `ePaperDisplay`). Some phones conflict with mobile data — temporarily disable mobile data, or wait a few seconds after connecting.

Q4 Screen doesn't update after upload?

A full E-Paper refresh takes ~21 s — that's normal. If nothing changes after 30 s, press **RST** to reboot.

Q5 How many photos can the album hold?

Internal flash has ~1 MB available; each photo is ~30 KB, so the theoretical maximum is ~30 photos. Use a TF card for near-unlimited expansion.



12 Notes & Version Info

- Use a reputable Micro SD card with capacity ≤ 32 GB, formatted as **FAT32**.
- Do not hot-plug the FPC cable — this can damage the E-Paper panel.
- Avoid prolonged direct sunlight and high temperature (> 60 °C) environments.
- This board is open-source hardware; the documentation and source code are free for personal learning, teaching, and secondary development. For commercial use, please contact the original manufacturer.

Document Version	
Document Version	v1.0
Applicable Firmware	photo_4_2.ino v1.0 and above
Update Date	August 2026
Maintainer	OLIGHKER