



USER MANUAL

2.9-inch Black & White E-Paper Electronic Shelf Label

WiFi AP · Web Editor · Barcode · TF Card Backup

User Manual v1.0

OLIGHKER · August 2026

Compatible firmware: A32_Label_2_9_V30

Hardware version: Rev 2.0



Contents

01 Product Overview

1.1 Introduction · 1.2 Specifications · 1.3 Applications

02 Hardware Description

2.1 Ports & Buttons · 2.2 Electrical Specs · 2.3 Pin Assignment

03 Compiling & Flashing with the Arduino IDE

3.1 Preparation · 3.2 IDE Install · 3.3 GxEPD2 · 3.4 Project · 3.5 Config · 3.6 Verify · 3.7 Upload

04 User Guide

4.1 Connect · 4.2 Editor · 4.3 Standard Tag · 4.4 Album Photo · 4.5 TF Card · 4.6 Language · 4.7 Browser Library

05 FAQ

Q1 – Q6

06 Technical Support

Product resources

1. Product Overview

1.1 Product Introduction

The A32 is a 2.9-inch black & white E-Ink electronic shelf label (ESL) based on the ESP32-WROOM-32E. It integrates a WiFi access point and a web-based editor, so you can design product price tags, generate EAN-13 barcodes, and send photos from your phone's album straight to the e-paper — without installing any app. Labels can optionally be archived to a TF card for redundant storage.

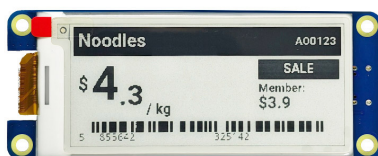


Fig. 1-1 A32 2.9-inch e-paper shelf label (front view)

The e-paper display retains its image even when power is removed. Energy is consumed only while refreshing; a static image can persist for years. This makes the A32 ideal for supermarkets, convenience stores and warehouses where content stays on display for long periods but changes only occasionally.

1.2 Key Specifications

Feature	Description
Display size	2.9 inch, 296 × 128 pixels
Display color	Black & white (1-bit)
MCU	ESP32-WROOM-32E (WiFi + BLE dual-mode)
Connectivity	Built-in WiFi access point (no external router required)
Editor	Built-in web editor — no app installation
Label templates	4 presets (Standard / Large Price / Promo / Minimal) + custom images
Barcode	Built-in EAN-13 generator with automatic check digit
Storage	Optional TF (Micro SD) card for redundant label storage
UI languages	English / Russian, switchable



Refresh	Full-screen refresh in approx. 2–3 s
Power supply	USB-C, 5 V

1.3 Applications

- Price tags in supermarkets, convenience stores and fresh-food shops
- Warehouse rack labels (SKU, batch, bin location)
- Meeting-room door signs and temporary notice boards
- Workstation cards and equipment inspection cards in factories
- Any e-paper application needing wireless refresh and no cabling

2. Hardware Description

2.1 Ports and Buttons

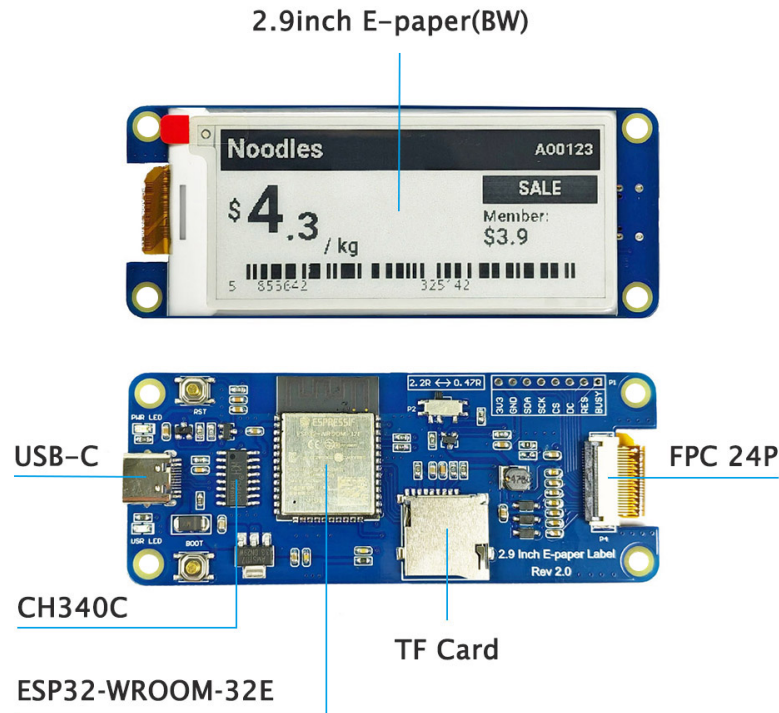


Fig. 2-1 A32 main board — ports and buttons

No.	Silkscreen	Type	Description
①	2.9 inch E-paper (BW)	Display module	GDEM029T94, 128×296 px, SSD1680 driver, connected to the main board via a 24-pin FPC ribbon cable
②	USB-C	Power / Serial	Provides 5 V power and USB-to-serial download via the CH340C; used for firmware flashing and serial logs
③	ESP32-WROOM-32E	MCU module	Dual-core 240 MHz, WiFi 802.11 b/g/n and Bluetooth; runs all application logic
④	CH340C	USB-to-serial chip	Converts the USB-C signals to the ESP32's UART0 (TX/RX); no external serial adapter



			needed
⑤	TF Card	Storage slot	Optional Micro SD card (FAT) for redundant storage of label bitmaps and form metadata
⑥	FPC 24P	Ribbon connector	24-pin FPC connector of the e-paper module, factory-soldered to the main board
⑦	PWR LED	Power indicator	Steady on when 5 V is present on USB-C
⑧	USR LED	User indicator	Controlled by GPIO2
⑨	RST button	Reset key	Short press resets the ESP32
⑩	BOOT button	Download-mode key	Hold BOOT + tap RST to enter the ESP32 serial download mode

2.2 Electrical Specifications

Parameter	Value
Input voltage	DC 5 V (via USB-C)
Operating current	≈ 80 mA average, ≈ 250 mA peak during refresh
Static power draw	< 0.01 mW (the e-paper keeps its image with no power applied)
Operating temperature	0 °C – 50 °C
Storage temperature	–20 °C – 70 °C

2.3 Pin Assignment (ESP32)

Function	ESP32 GPIO
E-paper CS	GPIO27
E-paper DC	GPIO25
E-paper RST	GPIO17
E-paper BUSY	GPIO26
E-paper SPI	Default VSPI (SCK=18 / MOSI=23)
TF card CS	GPIO5
TF card SPI	Shares VSPI with the e-paper, independent CS



3. Compiling and Flashing with the Arduino IDE

This chapter is required reading for anyone who wants to modify the firmware or re-flash the board. If you only intend to use the factory firmware, you may skip ahead to Chapter 4.

3.1 Preparation

3.1.1 Hardware Checklist

- 2.9-inch e-paper label board × 1
- USB-C data cable × 1 (must be a data cable — charge-only cables cannot communicate)
- A Windows / macOS / Linux computer × 1

3.1.2 Software Checklist

- Arduino IDE 2.x (2.3 or newer recommended — <https://www.arduino.cc/en/software>)
- ESP32 core package (esp32 by Espressif Systems, v2.0.14 or newer)
- CH340 serial driver (manual install on Windows; usually built into macOS / Linux)
- Project libraries:
 - GxEPD2 (e-paper library maintained by ZinggJM)
 - Built-in WiFi, WebServer, SPI, SD

3.2 Installing the Arduino IDE and the ESP32 Core

Step 1 — Install the Arduino IDE

Download the Arduino IDE 2.x installer for your operating system from the official website and follow the wizard.

Step 2 — Add the ESP32 Board Manager URL

In the Arduino IDE, open the menu:

File → Preferences (macOS: Arduino → Settings)

Paste the following URL into the "Additional boards manager URLs" field:

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

Tip If other URLs are already listed, separate entries with commas or line breaks — multiple URLs can coexist.

Click OK to save.

Step 3 — Install the ESP32 core

Navigate to:

Tools → Board → Boards Manager...



Type `esp32` in the search box, locate "esp32 by Espressif Systems", pick version 2.0.14 or newer, and click Install.

Note Depending on your network, the download may take 3–10 minutes. Please be patient.

Step 4 — Install the CH340 driver (Windows only)

Download the CH340 driver from the WCH official website or any dev-board vendor's download page, install it, and reboot the computer. On macOS, if a "System extension blocked" prompt appears, go to System Settings → Privacy & Security and allow it.

After installation, connect the board via USB-C. A new USB-SERIAL CH340 (COMx) entry under Device Manager → Ports (COM & LPT) confirms the driver works.

3.3 Installing the GxEPD2 Library

Navigate to:

Sketch → Include Library → Manage Libraries...

Type GxEPD2 in the search box, find "GxEPD2" by Jean-Marc Zingg, and click Install.

3.4 Opening the Project

Copy the entire `A32_Label_2_9_bw_V30/` directory to any local path (avoid non-ASCII characters in the path), then in the Arduino IDE:

File → Open... → select `A32_Label_2_9_bw.ino`

The IDE automatically picks up the other `.h` files in the project folder. The expected layout is:

```
A32_Label_2_9_bw/
├─ A32_Label_2_9_bw.ino           # main program
├─ GxEPD2_display_selection_new_style.h  # display selection
├─ GxEPD2_selection_check.h          # compatibility check
├─ GxEPD2_wiring_examples.h          # wiring reference
└─ myfont.h                         # custom fonts
```

3.5 Configuring the Board

Open the Tools menu and configure the following options:

Option	Recommended setting
Board	ESP32 Dev Module (or ESP32 Wrover Module — both work)
Upload Speed	921600 (drop to 115200 if flashing fails)
CPU Frequency	240MHz (Workbench)
Flash Frequency	80MHz



Flash Mode	QIO
Flash Size	4MB (32Mb)
Partition Scheme	Default 4MB with spiffs
Port	The COM port that appeared in Step 4 (e.g. COM5, /dev/ttyUSB0, /dev/cu.usbserial-*)
Programmer	Esptool

3.6 Compiling (Verify)

Click the ✓ (Verify) button in the toolbar, or press Ctrl + R (macOS: Cmd + R).

Tip The first build compiles the whole ESP32 toolchain and GxEPD2 and may take 5–15 minutes; subsequent incremental builds usually finish within tens of seconds.

"Done compiling." together with "Sketch uses XXXXX bytes (XX%) of program storage space" in the console means the build succeeded.

3.7 Flashing (Upload)

Step 1 — Enter download mode

The A32 board can usually be flashed directly. If you see "A fatal error occurred: Failed to connect to ESP32...", enter download mode manually:

1. Hold BOOT

Press and hold the BOOT button on the board.

2. Tap RST

Short-press the RST button once, then release it.

3. Release BOOT

Let go of the BOOT button.

The ESP32 is now in serial download mode.

Step 2 — Upload the firmware

Click the → (Upload) button in the toolbar, or press Ctrl + U (macOS: Cmd + U). The console shows the upload progress. Wait until the following output appears:

```
Writing at 0x000XXXXX... (100 %)
Wrote XXXXX bytes (XXXXX compressed) at 0x000XXXXX in X.X seconds.
Hash of data verified.
Leaving...
Hard resetting via RTS pin...
```



Flashing succeeded and the board reboots automatically.

Step 3 — Check the serial log

After flashing, open Tools → Serial Monitor (Ctrl + Shift + M) and set the baud rate to 115200. On a successful boot you should see a log similar to:

```
=====
AP start: OK
AP SSID: ePaperDisplay
AP Pwd : 88888888
AP Ch  : 6
AP IP   : 192.168.4.1
TX Pwr  : 19.5dBm (max)
Shelf label mode. Connect phone to AP, open the IP in a browser.
=====
HTTP server started.
```

The e-paper shows a centered on-screen hint containing the WiFi name, the password and the management IP address.



4. User Guide

4.1 Connecting to the Device

Step 1 — Join the WiFi access point

WiFi name (SSID)	ePaperDisplay
Password	88888888

Search for WiFi with your phone or laptop, find ePaperDisplay and connect.

Note On first connection, your device may warn that "This network has no Internet access". Choose Keep using this network / Stay connected — this is expected, as the ESP32 does not provide Internet access.

Step 2 — Open the web console

Once connected, enter the following address in your browser (Chrome / Safari / Edge strongly recommended):

```
http://192.168.4.1
```

The web editor home page opens.

4.2 Editor Layout

The web editor is organized into the following cards, top to bottom:

1. Preview & refresh — shows what the label currently looks like on the e-paper, with the Send to display button.
2. Photo album — pick a photo from your phone's album; it is automatically binarized and sent to the display.
3. TF label library (optional) — browse / edit / rename / delete labels stored on the TF card.
4. Template selection — switch between the 4 label templates.
5. Form area — product name, price, unit, promo tag, member price, original price, barcode, SKU.
6. Browser library — save frequently used labels in the browser via IndexedDB.

4.3 Creating a Standard Price Tag

Follow these steps to create the most common "Standard template" price tag:

1. Pick a template

In the Template card, click Standard.

2. Fill in the product information



-
- Product name: Instant Noodles
 - Currency: \$
 - Price: 4.30
 - Unit: bag
 - Spec / origin: 500g / China

3. Choose a promo tag (optional)

Select SALE / NEW / MEMBER / LIMIT from the drop-down, or leave it on none.

4. Fill in the member price (optional)

Member price: 3.90

5. Fill in the original price (optional)

Original price: 4.80 (a strikethrough is applied automatically)

6. Enter the barcode

Barcode: 692015240002 (enter 12 digits; the 13th check digit is calculated automatically)

7. Enter the SKU

SKU: A00123

8. Preview and send

The canvas at the top previews the final layout in real time. Click Send to display — the e-paper refreshes with your tag in 2–3 seconds.

4.4 Sending a Photo from the Album

To put a product photo or a store logo directly onto the e-paper:

1. Locate the card

Scroll to the Image from phone album card.

2. Upload the photo

Click Upload from album and pick a photo (resolution close to 296×128 is best; other sizes are scaled proportionally).

3. Pick a binarization mode

- Threshold — best for text, logos and line art; crisp edges.
- Dithered — best for photos and gradients; smoother gray tones.

4. Send to the display



Wait until "Photo loaded. Press \"Send to display\"." appears at the top, then click Send to display. The e-paper shows the photo in 2–3 seconds.

Tip The e-paper is black & white only, so photos are automatically binarized before being sent. If the result is not to your liking, switch between Threshold and Dithered and try again.

4.5 Using a TF Card (Optional)

The TF card is an optional accessory — the label works fully without one. When a card is inserted, historical labels are archived and can be restored with one click after a power cycle.

4.5.1 First Use

1. Prepare the card

A FAT32-formatted Micro SD card (≤ 32 GB recommended).

2. Insert the card

With the board powered off, slide the TF card into the slot (contacts facing down).

3. Save the first label

Power on, edit a label and click Send to display. The firmware creates the `/labels/` directory automatically and stores the label bitmap and form metadata.

4. Verify the status

Below the editor you should see: TF: ready · saved label available · usedMB/totalMB MB.

4.5.2 Browsing / Redisplaying Saved Labels

Click TF label library to expand a gallery of all saved labels with thumbnails, file names and timestamps. Click Show to push the selected label to the e-paper immediately; click Edit to load its form data back into the editor.

4.5.3 Formatting the TF Card

Click Format TF card. After confirming twice, everything under `/labels/` is erased (other files on the card are not affected).

4.6 Switching the UI Language

The drop-down box in the top-right corner supports English / Русский. The choice is stored in the browser's `localStorage` and survives page reloads.



4.7 Saving to the Browser Library

Click Save to Browser to store the current label's form data in the browser's IndexedDB. On your next visit, click an entry in the list to refill the form with one click.

Note The browser library and the TF card library are independent. Clearing browser data also wipes the browser library.



5. FAQ

Q1: Cannot connect to the ePaperDisplay WiFi?

1. Make sure the board is powered on (PWR LED steady on).
2. Make sure your computer/phone does not filter unknown hotspots via random-MAC policies.
3. Move away from 2.4 GHz interference sources (microwave ovens, Bluetooth speakers, other WiFi routers).
4. If it still fails, press the RST button to reboot the board.

Q2: WiFi connects, but 192.168.4.1 won't open?

1. Check that the address is exactly `http://192.168.4.1` (no https, no extra port number).
2. Some phones show a "no Internet" interstitial when joining an offline hotspot — tap Keep using this network.
3. Try another browser (Chrome / Safari / Edge recommended).

Q3: The screen looks pale after a refresh, with ghosting?

A small amount of ghosting after the first refresh is normal — it is inherent to how E-Ink works. Refresh again or wait 1–2 minutes and it disappears. If ghosting persists, leave the board unpowered overnight, then power it on again.

Q4: The TF card shows "TF: unavailable"?

1. Make sure the card is FAT32 (not exFAT or NTFS).
2. Test the card in a computer; a damaged card will show as unavailable.
3. Cards larger than 32 GB are not recommended — they can exhaust ESP32 resources.

Q5: How do I update 100 labels in bulk?

Use the HTTP API programmatically:

Endpoint	Description
POST /upload	Uploads a 4736-byte 1-bit bitmap + form metadata; refreshes the display and saves to TF
GET /tf-labels	Lists all saved labels
POST /tf-label-show?file=xxx	Pushes a saved label to the display
POST /tf-label-rename?file=xxx&newname=yyy	Renames a label
POST /tf-label-delete?file=xxx	Deletes a label

Write your own script (Python, Node.js, ...) and call these endpoints over WiFi in bulk.



Q6: Can it display Chinese?

Yes. Two changes are needed:

1. Add a bitmap font containing the Chinese glyphs to myfont.h (or use a Chinese-capable library such as u8g2).
2. Switch the canvas rendering font in the web editor to a font with Chinese support.

Since the panel is only 296×128 px, Chinese fonts of 16 px or smaller are recommended for readability.



6. Technical Support

- Product resources: <https://olighker.com>
-

© 2026 OLIGHKER. All rights reserved.

2.9-inch B&W E-Paper Electronic Shelf Label · User Manual v1.0