

Firmware Flashing Guide

How to install firmware on your OSR2+ device

⚠ THE SINGLE MOST IMPORTANT THING IN THIS GUIDE

When installing ESP32 board support (Part 3), you **must** select version **2.0.6** specifically. Newer versions break this firmware. This is covered in detail in Part 3, but it gets its own warning here too because it's that important.

IN THIS GUIDE

Part 1 Install the USB driver

Part 2 Install Arduino IDE

Part 3 Add ESP32 board support (version 2.0.6)

Part 4 Open the firmware and configure

Part 5 Flash the firmware to your device

Part 6 Common issues

Before you start

You'll need:

- Your OSR2 device with a **good** USB cable
- A Windows PC
- About 15–20 minutes the first time you do this
- The firmware `.ino` file you were sent

USE A GOOD USB CABLE AND PORT

This matters more than you'd think. Many USB cables are charge-only and don't carry data properly, and many PC USB ports — **especially front-panel ports on desktops** — are unreliable for flashing. If you've ever had USB devices behave inconsistently on a particular port, don't use that port for this. **Use a rear USB 2.0 port on a desktop, or any port that you know works**

reliably. A bad cable or port means you'll need the BOOT button workaround every time you flash. A good one means flashing is fully automatic.

The first-time setup takes a while because there are programs to install. After that, future re-flashes only take a couple of minutes.

Part 1 Install the USB driver

Your OSR2 talks to your PC through a chip called the CP210x. Windows needs a driver for it.

1. Open your browser and go to **silabs.com** (Silicon Labs).
2. In their search bar at the top, search for **CP210x VCP**.
3. Click the result titled *"CP210x USB to UART Bridge VCP Drivers"*.
4. Scroll down to the **Software Downloads** section. You'll see several CP210x options.
5. Click **"CP210x Windows Drivers with Serial Enumerator"** — this is the version we recommend because it gives your device a stable COM port number that doesn't change between plug-ins. (See the highlighted entry in the screenshot below.)
6. The file downloads as a ZIP (around 10 MB).
7. Right-click the ZIP file and choose **Extract All**.
8. Open the extracted folder.
9. Double-click the file ending in `x64.exe` (this is the version for 64-bit Windows — almost all modern PCs).
10. Click **Next**, accept the agreement, click **Next**, click **Finish**.

SILICON LABS Products ▾ Applications ▾ Software & Tools ▾ Resources ▾ Company ▾ English ▾ AskAI 🔍

// Software and Tools // USB to UART Bridge VCP Drivers

Download and Install VCP Drivers

Downloads for Windows, Macintosh, Linux and Android below.

*Note: The Linux 3.x.x and 4.x.x version of the driver is maintained in the current Linux 3.x.x and 4.x.x tree at www.kernel.org.

Software Downloads

Software (11)

Software	11
CP210x Universal Windows Driver	v11.5.0 12/31/2025
CP210x VCP Mac OS X Driver	v6.0.3 5/30/2025
CP210x VCP Windows	v6.7 9/3/2020
CP210x Windows Drivers	v6.7.6 9/3/2020
CP210x Windows Drivers with Serial Enumerator	v6.7.6 9/3/2020
CP210x_5x_AppNote_Archive	9/3/2020
CP210x_VCP_Win2K	9/3/2020
Linux 2.6.x VCP Revision History	9/3/2020
Linux 3.x.x/4.x.x/5.x.x VCP Driver	v3.x.x/4.x.x/5.x.x 1/29/2021
VCP Driver for WinCE50	v2.1 9/3/2020
VCP Drivers for WinCE50	v2.1 9/3/2020

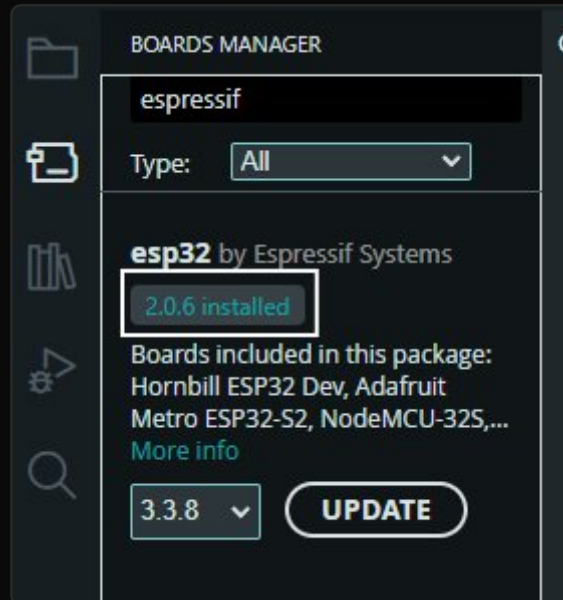
The "CP210x Windows Drivers with Serial Enumerator" download on silabs.com

Driver installed. You only need to do this once per PC.

Part 2 Install Arduino IDE

Arduino IDE is the program that puts firmware onto your OSR2.

1. Go to **arduino.cc** in your browser.
2. Click **Software** at the top of the page.
3. Scroll down to the Arduino IDE section. On the right side you'll see a dropdown and a green **DOWNLOAD** button.
4. The dropdown defaults to **"Windows Win 10 or newer (64-bit)"**. Leave this unless you're on a different OS.
5. Click the green **DOWNLOAD** button.
6. The site will ask for a donation — click **JUST DOWNLOAD** (small grey text) to skip.
7. Once the installer downloads, run it.
8. Click **I Agree** to the licence, then **Next**, then **Install**, then **Finish**.



The Arduino IDE download page — pick your OS and click DOWNLOAD

Arduino IDE will launch. It opens to a blank "sketch" window. Leave it open for now.

Part 3 Add ESP32 board support (version 2.0.6)

⚠ IMPORTANT — VERSION 2.0.6 ONLY

You **must** install version **2.0.6** specifically — not the latest version, not 3.x, not anything else. Newer versions of the ESP32 board package have breaking changes that will prevent this firmware from compiling or cause it to behave incorrectly.

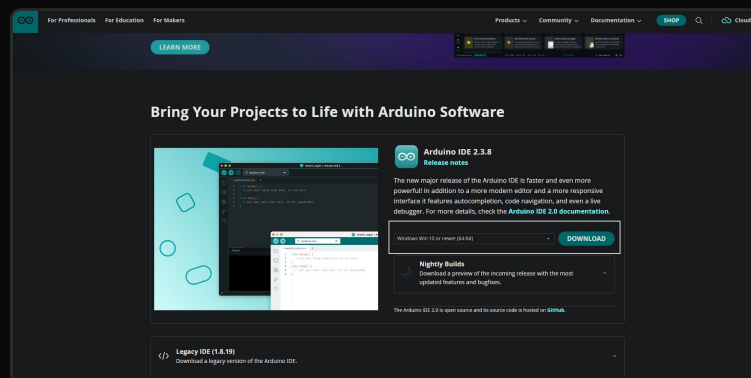
The default in Arduino IDE is the latest version. **You will need to manually pick 2.0.6 from a dropdown.** Do not skip this step.

By default, Arduino IDE can't talk to ESP32 chips. We need to add support — and pick the specific version this firmware was built against.

1. Look at the **left side** of the Arduino IDE window. You'll see a vertical strip of icons going down.
2. The **second icon from the top** looks like a small chip or board. **Click it.** The Boards Manager panel opens on the left.
3. In the search box at the top of the Boards Manager panel, type **esp32** (or just "espressif" — both work).
4. After a moment, results appear. The one you want is:

esp32 by Espressif Systems

5. Below the name, you'll see a **version dropdown**. By default it shows the latest version.
6. **THIS STEP IS CRITICAL:** Click the version dropdown. Scroll through the list and select **2.0.6**. Do not pick any other version.
7. Click the **INSTALL** button.
8. Wait. This download is about 200 MB and takes 2–5 minutes depending on your internet speed. A progress bar appears at the bottom.
9. When it finishes, the entry shows **"2.0.6 installed"** in green, like in the screenshot below.



Boards Manager after successful installation — the green "2.0.6 installed" badge confirms it worked

IF "ESP32 BY ESPRESSIF SYSTEMS" DOESN'T APPEAR IN YOUR SEARCH

On a fresh Arduino IDE install this should work out of the box, but if it doesn't:

1. Click **File → Preferences**
2. Find the field labelled *"Additional boards manager URLs"*
3. Paste this address into it: `https://espressif.github.io/arduino-esp32/package\_esp32\_index.json`
4. Click OK
5. Go back to Boards Manager and search for esp32 again

Part 4 Open the firmware and configure

Now we open the firmware file **first**, then configure the IDE settings. Doing it in this order means the settings are applied to this firmware sketch correctly.

4a — Open the firmware file

1. Find the firmware file you were sent. It ends in `.ino`.
2. **Double-click it.** Arduino IDE should open with the firmware loaded.

3. A popup may appear saying "The file needs to be inside a sketch folder. Create a folder for the file?" — click **OK**.

4b — Plug in your OSR2

Plug your OSR2 into the PC with a **good USB cable**, into a **good USB port** (see the warning at the start of this guide). Wait 5 seconds for Windows to recognise it.

4c — Select the board

1. Click **Tools** at the top of the IDE.
2. Hover over **Board**. A submenu opens.
3. Hover over **esp32**. Another submenu opens (a long list).
4. Scroll down and click **DOIT ESP32 DEVKIT V1**.

4d — Select the COM port

1. Click **Tools** again.
2. Hover over **Port**.
3. You'll see one or more ports listed (like COM3, COM4, etc.). Click the one that matches your OSR2.

NOT SURE WHICH PORT IS YOURS?

Unplug the OSR2 and open **Tools** → **Port** again. Note which port disappears. Plug the OSR2 back in. That port is yours.

4e — Disable "Erase All Flash"

⚠ CRITICAL FOR PROTECTING WIFI CREDENTIALS

If you skip this step, every re-flash will wipe the saved WiFi network and you'll have to redo the setup from scratch. Disable it now.

1. Click **Tools** at the top.
2. Find **Erase All Flash Before Sketch Upload** in the menu.
3. Make sure it is set to **Disabled**.

Part 5 Flash the firmware

5a — Upload

1. Look at the top-left of the Arduino IDE window. You'll see two icons: a **checkmark** and a **right-pointing arrow**.
2. Click the **arrow**. This is the Upload button — it compiles the firmware and writes it to your device.
3. You'll see messages scrolling at the bottom of the IDE: first compilation messages, then `Writing at 0x00010000... (X %)` dots as it uploads.
4. Wait. The whole process takes about 1–2 minutes.
5. When done, you'll see `Leaving...` and `Hard resetting via RTS pin...` in the output.

That's it — the firmware is now on your OSR2.

5b — Optional: verify via Serial Monitor

Most users will just open **MultiFunPlayer** (or whichever software they normally use) and confirm the device responds — that's a perfectly valid check.

If you'd like to verify directly from Arduino IDE instead, here's how:

1. Click the **magnifying glass icon** in the top-right of the IDE. This opens the Serial Monitor.
2. In the bottom-right corner of the Serial Monitor, find the baud rate dropdown.
3. Set it to **115200**.
4. Press the small **EN button** on the ESP32 (the left hole on your device — you may need a thin tool to reach it).
5. Text should scroll past, ending with `Ready!`

If you see the firmware name and `Ready!`, the firmware is running correctly.

Part 6 Common issues

"Failed to connect to ESP32: Timed out waiting for packet header"

The device couldn't enter flash mode automatically. This is almost always caused by a USB cable or port issue — but the BOOT button can also force it into flash mode manually:

1. Hold down the **BOOT button** on the ESP32 (right hole).
2. While holding BOOT, press and release the **EN button** (left hole).
3. Wait a second, then release BOOT.
4. Click the Upload arrow again in Arduino IDE.

5. After flashing succeeds, press EN once to boot normally.

If this happens every single time you flash, your USB cable or port isn't great. Try a different cable, or a rear USB 2.0 port. A good combination means flashing is fully automatic.

No COM port appears under Tools → Port

- The CP210x driver isn't installed correctly. Go back to Part 1 and re-run the installer.
- Your USB cable might be charge-only — try a different cable that you know carries data.
- Try a different USB port (rear ports on desktops are usually most reliable).

Compilation errors mentioning missing functions, libraries, or `ledcSetup`

Almost always caused by the **wrong ESP32 version**. Go back to Part 3, open Boards Manager again, find "esp32 by Espressif Systems", and change the version dropdown to **2.0.6**. Click INSTALL again to switch versions. Then re-flash.

Serial monitor shows garbled characters

The baud rate is wrong. Make sure it's set to **115200** in the dropdown.

WiFi credentials disappeared after re-flashing

"Erase All Flash Before Sketch Upload" was probably enabled. Go back to Part 4e and disable it. You'll need to redo WiFi setup once after this.

Anything else

If you run into a problem this guide doesn't cover, send a screenshot of the error message via your usual support channel. Most issues are quick fixes once we know what the IDE is actually complaining about.