

Put the firmware on the Pico (no programming needed)

1. Pick the file that matches your board, from `fencingohmtester/firmware/` :

Board	File
Raspberry Pi Pico (RP2040)	<code>ohm_tester_PICO.uf2</code>
Raspberry Pi Pico 2 (RP2350)	<code>ohm_tester_PIC02.uf2</code>

2. Hold down the white **BOOTSEL** button on the Pico and plug it into your computer with a USB cable. Let go. A drive called **RPI-RP2** (or **RP2350**) appears.
3. Drag the .uf2 file onto that drive. It copies, the drive disappears, and the Pico restarts running the tester.

Do this before wiring anything, and again any time a new version comes out: same three steps, wiring stays.

Wrong file for the board? Nothing breaks: it just does not start. Hold BOOTSEL and drag the other one.

Bring-up checks, in order

Check	Expect	If not
☐ 1. Cell in, switch on	Red, yellow, green flash in turn, one beep, then OHM TESTER	A light stays dark: LED backwards (long leg to the resistor). No beep: piezo wires. Blank screen: OLED VCC/GND or SDA/SCL swapped
☐ 2. Start screen ADC line	"ADC: ADS1115 16-bit" (or "12-bit (ADS1015)", also fine)	"ADC NOT FOUND" and the red light: ADS SDA/SCL, VDD, or ADDR not grounded
☐ 3. Nothing plugged in	READY, all lights off	Something shows: a jack touches its neighbour or a chip pin is bridged
☐ 4. Banana lead from CLIP to PROBE	LAME / MASK, green, 0.0 to 0.05 ohm	Nothing happens: check U1/U2 power, R_LIM, R_REF, the DRV_EN pull-up
☐ 5. A 1 ohm 1 % resistor between CLIP and PROBE (clip leads)	1.00 ohm, +/- 0.02 (the light may sit on the green / yellow line)	Way off: a SENSE wire on the wrong chip, or A2/A3 not on the R_REF legs
☐ 6. Same with a 10 ohm 1 % resistor	10.0 ohm +/- 0.1, red light + buzz	Reads 1 % off on both: R_REF is a 1 % part, which is fine
☐ 7. A cord you know is good, both ends	BODY CORD, chirp, green, all bars short of the line	One wire OPEN: that jack's FORCE wire; nonsense values: its SENSE wire
☐ 8. Same cord, leave it still for 10 s	No clicks, stays green	Clicks while still: loose jack or a bad joint on the lug
☐ 9. Hold the button 1 s, then again	"mute" appears in the header, then a beep and it goes	Nothing: button wire to Pico pin 26, other leg to ground
☐ 10. Switch on, then unplug USB	Keeps running; battery % replaces USB	Goes dark: SW1 off, D1 backwards, or the cell is flat

What is in the firmware

File	What it is
ohm_tester/ohm_tester.ino	Hardware: chips, ADC, screen, lights, piezo, button, the fast break sampler on the Pico's second core
ohm_tester/ohm_logic.h / .cpp	All the measuring arithmetic, the "what is plugged in" detection, pass / fail, which light and which sound. Plain C++, no Arduino needed
test_ohm_logic.cpp	43 automated tests: accuracy model, cords, shorts, crossed, leaky, flex, foil / sabre / epee, lame, and whole scenarios (plugging in, unplugging, swapping weapons, sweeping a lame) checked for the light and sound you would get
firmware/*.uf2	Ready-to-drag builds of the above

The tests model real gear as resistor networks and solve them, so a crushed cord or a dead lame patch reads in the test exactly as the switch chips would see it. Run them on any PC:

```
g++ -std=c++17 -O2 -o test test_ohm_logic.cpp ohm_tester/ohm_logic.cpp && ./test
```

(no g++? `pip install ziglang`, then `python -m ziglang c++` with the same arguments.)

Building it yourself (only if you change the code)

1. Arduino IDE, Boards Manager URL: `https://github.com/earlephilhower/arduino-pico/releases/download/global/package_rp2040_index.json`, install "Raspberry Pi Pico/RP2040/RP2350".
2. Library Manager: install **U8g2**.
3. Open `ohm_tester/ohm_tester.ino`, board "Raspberry Pi Pico" or "Raspberry Pi Pico 2", Upload.

Tested with arduino-pico 6.2.0 and U8g2 2.36. The release .uf2 files were built from exactly the source in this folder.

Prototype. The logic passes its tests and the firmware compiles cleanly for both boards, but the first physical unit has not been built yet. Expect to check step 5 above carefully on the first build.