

# ESPARGOS One

**Digital Phased Array for Channel Measurement & Research**

# ESPARGOS One Quick Start Manual

**Revision:** 1.0

**Date:** June 2026

TTI GmbH / TGU ESPARGOS

Latest version online

**<https://espargos.net>**

## Contents

<b>1</b>	<b>Overview</b>	<b>5</b>
1.1	What's in the Box . . . . .	5
1.2	What You Need . . . . .	5
1.3	Key Features . . . . .	6
<b>2</b>	<b>Intended Applications</b>	<b>6</b>
<b>3</b>	<b>Technical Specifications</b>	<b>7</b>
<b>4</b>	<b>Hardware Overview</b>	<b>8</b>
4.1	Connectors . . . . .	8
4.2	Buttons & LEDs . . . . .	9
4.3	Camera Hole . . . . .	9
4.4	M4 Holes on Case Side and Breaking Point Cutouts . . . . .	9
4.5	Mounting Options . . . . .	10
<b>5</b>	<b>How It Works</b>	<b>11</b>
5.1	Sensors and Controller . . . . .	11
5.2	Synchronization Principle . . . . .	11
5.3	Data Capture . . . . .	12
<b>6</b>	<b>Quick Start</b>	<b>13</b>
Step 1	Power up <b>ESPARGOS One</b> . . . . .	13
Step 2	Connect to ESPARGOS . . . . .	13
Step 3	Access the Web Interface . . . . .	14
<b>7</b>	<b>Firmware Updates</b>	<b>16</b>
7.1	Web Update . . . . .	16
7.2	Firmware Recovery . . . . .	17
<b>8</b>	<b>Python Library: pyespargos</b>	<b>18</b>
<b>9</b>	<b>Advanced Topics</b>	<b>19</b>
9.1	Single- and Multi-Board Setups with Unsynchronized Arrays . . . . .	19
9.2	Multi-Board Phase-Coherent Systems (Research Only) . . . . .	20
9.3	Configuring the Reference Signal . . . . .	21
9.4	Transmitting With ESPARGOS / Radar . . . . .	23
<b>10</b>	<b>Cleaning &amp; Maintenance</b>	<b>24</b>
<b>11</b>	<b>Troubleshooting</b>	<b>25</b>

---

<b>12</b>	<b>Safety, Regulatory &amp; Environmental Information</b>	<b>28</b>
12.1	Electrical Safety . . . . .	28
12.2	Handling . . . . .	28
12.3	Regulatory: WiFi Country Code . . . . .	28
12.4	ESD & Servicing . . . . .	28
12.5	Ground Loops . . . . .	28
12.6	Operating & Storage Conditions . . . . .	29
12.7	Intended for Use Within the European Union . . . . .	29
12.8	Lawful Use: Privacy, Interception & Data Protection . . . . .	29
12.9	Security & Vulnerability Reporting . . . . .	29
12.10	REACH: Substances of Very High Concern (Art. 33) . . . . .	30
12.11	Cabling . . . . .	30
12.12	CE Conformity & Electromagnetic Compatibility (EMC) . . . . .	30
12.13	Open-Source Software & License Notices . . . . .	30

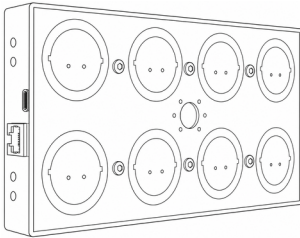
**Note** Jump ahead to page 13 for a quick start guide to get **ESPARGOS One** up and running in minutes, or visit <https://espargos.net/setup> for the most up-to-date instructions.

## Overview

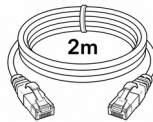
**ESPARGOS One** is a **phase-coherent, real-time WiFi measurement array**. It passively receives the ambient WiFi signals around it and, for each received packet, measures the **Channel State Information (CSI)**: the complex (amplitude and phase) channel response at every antenna, resolved across the OFDM subcarriers. Its 8 dual-polarized patch antennas form a  $2 \times 4$  grid at half-wavelength ( $\lambda/2$ ) spacing, and all eight receivers are **phase-synchronized** to a common clock and phase reference, so the CSI is captured coherently across the whole array. This phase coherence exposes the spatial structure of the WiFi channel: the **angle of arrival** of incoming signals, multipath propagation, and polarization. The CSI is streamed to your computer in real time.

### What's in the Box

The package contains the following items:



1 × **ESPARGOS One** antenna array



1 × Network cable (RJ45)



1 × Quick Start Manual (this booklet)

*Please check the delivery note for any potential additional items.*

### What You Need

You need these additional (not included) items to set up and use **ESPARGOS One**:

- IEEE 802.3at Type 1 (Class 3) PoE injector / switch or 5 V/2 A USB-C power supply with suitable cable (Note: Non-standard “passive” PoE injectors are not supported.)
- WiFi transmitter (e.g., smartphone, laptop, access point) in the same 2.4 GHz band
- Computer with web browser for accessing the web interface and streaming CSI

## Key Features

- Phase-synchronous CSI from all 8 elements, up to 40 MHz bandwidth
- Passive promiscuous mode: Zero interference with other network devices
- Dual polarization per element:  $\pm 45^\circ$  slant-polarized feeds, switchable
- Single-cable deployment: PoE Ethernet or USB-C for power *and* data
- Built-in web interface, no additional software needed to configure
- `pyespargos` Python library for real-time CSI streaming
- *For research*: Multiple units combine into larger phase-coherent systems

## Intended Applications

**ESPARGOS One** is a **test-and-measurement instrument**: it measures and records WiFi channel data for offline or real-time analysis by the operator. It is intended for professional, research, and educational users. Typical applications include:

- **WiFi channel measurement and sounding**: characterizing the 2.4 GHz propagation channel and recording phase-coherent CSI, including multipath and angle-of-arrival information, for research and engineering
- **Research and teaching platform**: MIMO and array-processing experimentation, algorithm development, and reproducible data collection for wireless research and education
- **Measurement front-end for sensing research**: providing raw CSI measurements that researchers may use to investigate passive WiFi sensing techniques

In its default configuration, **ESPARGOS One** only *receives* and measures pre-existing, ambient WiFi signals, and is **not supplied as a localization, tracking, or radiodetermination product**. The hardware is, however, capable of transmitting (for example for active WiFi radar): transmission is **disabled by default** and may only be enabled by an operator who is authorized to transmit (see *Transmitting With ESPARGOS*, page 23). Any localization, sensing, or similar application built on the measurement data, as well as any use of the transmit capability, is implemented and operated solely by the user, who is responsible for its lawful use (see *Lawful Use*, page 29).

**ESPARGOS One** is a measurement instrument and is **not designed or qualified for safety-critical or life-supporting functions**. Do not rely on its measurements for decisions where a failure, error, or interruption could lead to injury, loss of life, or significant property or environmental damage.

## Technical Specifications

Software-related specifications are subject to change with firmware updates. For the latest details, please visit <https://espargos.net>.

### Antenna Array

<b>Arrangement</b>	2 × 4 (2 rows, 4 columns = 8 elements)
<b>Antenna type</b>	Dual slant-polarized patch ( $\pm 45^\circ$ feeds, with some intentional cross-polarization sensitivity)
<b>Polarization</b>	Per-element RF switch; selects one polarization at a time (or reference / isolation)
<b>Element spacing</b>	60 mm ( $\lambda/2$ at 2.4 GHz)

### Radio & CSI

<b>Frequency band</b>	2.4 GHz ISM band (2.412–2.472 GHz center frequency)
<b>Standards</b>	IEEE 802.11b/g/n/ax (passive / promiscuous receive)
<b>CSI bandwidth</b>	20 MHz (802.11g/n/ax); 40 MHz only via 802.11n HT40 channel bonding
<b>CSI formats</b>	L-LTF (802.11g/n/ax, 20 MHz), HT-LTF (802.11n, 20 MHz or 40 MHz), HE-LTF (802.11ax, 20 MHz)
<b>CSI output</b>	Per-subcarrier complex CSI for every received packet
<b>CSI resolution</b>	L-LTF: 12-bit; HT-LTF / HE-LTF: 8-bit (I and Q per subcarrier)
<b>Reference</b>	Shared 40 MHz clock and 2.4 GHz phase reference (distributed internally)
<b>Gain control</b>	Automatic per-antenna AGC, or manual global / per-sensor gain
<b>Operation</b>	Receive-only (passive) by default; an optional WiFi radar transmit mode is disabled by default (see page 23)

### Interfaces

<b>Ethernet</b>	100BASE-TX (RJ45)
<b>PoE</b>	IEEE 802.3at Type 1, Class 3
<b>USB</b>	USB-C: 5 V/2 A power supply; UART/serial access
<b>Sync I/O</b>	2 × SMA: 3.3V DC bias + 40 MHz reference + 2.4 GHz phase

### Software & Connectivity

<b>CSI streaming</b>	Real-time over Ethernet (WebSocket / UDP) or USB-C
<b>Control</b>	Built-in web interface and JSON HTTP/RPC API
<b>Python library</b>	<code>pypespargos</code> for configuration and CSI streaming
<b>Networking</b>	IPv4 / IPv6; mDNS service discovery
<b>Updates</b>	Online firmware and web-interface updates over HTTPS

### Power, Mechanical & Environment

<b>Power consumption</b>	~3.5 W standby power consumption
<b>Dimensions</b>	240 mm × 120 mm × 26.5 mm
<b>Mounting</b>	VESA 100 mm (M4); $\frac{1}{4}$ "-20 tripod; side M4 holes
<b>Environment</b>	Indoor use only; 0 °C to +50 °C; 20–80% RH (non-condensing)

## Hardware Overview

### Connectors



Figure 1. The side panel of ESPARGOS One with the RJ45 Ethernet and USB-C connectors.

<b>RJ45 Ethernet</b>	Data and PoE power input
<b>USB-C</b>	Alternative 5 V power (min. 2 A); UART/serial access
<b>SYNC OUT</b>	3.3 V DC bias + 40 MHz clock + 2.4 GHz phase ref (Master)
<b>SYNC IN</b>	External reference clock and phase signal input (Slave/Master)



Figure 2. The two SMA reference-signal ports, SYNC IN and SYNC OUT.

**SYNC Ports: Advanced Use Only** The **SYNC IN** and **SYNC OUT** SMA ports are for **advanced users only**. They serve solely to combine several **ESPARGOS One** units into one larger phase-coherent system and are not used in normal single-array operation. See the phase-coherent multi-board setup (page 20) before connecting anything to them. **Do not connect antennas or unterminated cables to these ports.** They are a reference signal interface, not an antenna connection.

## Buttons & LEDs

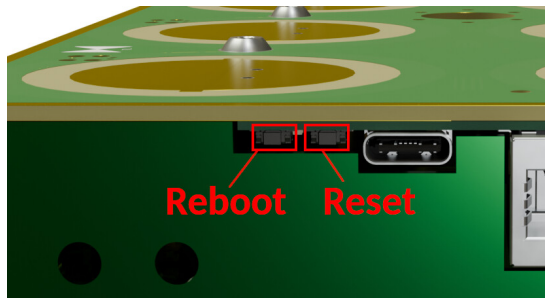


Figure 3. The Reboot and Factory Reset buttons.

<b>Reboot button</b>	Short press: reboot, all settings preserved
<b>Factory Reset button</b>	Hold 5–10 s until reboot: all settings erased

The LEDs are *inside* the case, but faintly visible from outside. An **orange** LED indicates PoE power supply. A **blue** LED will be always-on while the device is powered on. A **red** LED will blink briefly after boot (and if the “identify” feature on the web interface is used). A **green** and an **orange** LED indicate network activity.

**Factory Reset:** A factory reset erases all settings and reverts to the default IP address (192.168.1.2 / 255.255.0.0). The firmware version is not affected.

### Camera Hole

The PCB has a large, circular hole in the middle, with additional mounting holes next to it. This is for a future hardware upgrade with an integrated camera, and not intended for users.

### M4 Holes on Case Side and Breaking Point Cutouts

The side walls carry additional **M4-sized holes**, and the housing has marked **breaking-point cutouts** that can be broken out to route cabling into the enclosure.

**Case Modifications: Research Use Only, at Your Own Risk** Using the side M4 holes or breaking out a cutout **modifies the enclosure**. This is **not officially supported** and is intended for **research use only**: it may void the product’s regulatory compliance (e.g., EMC, electrical safety, and radio approvals) as well as the warranty, and a cutout cannot be restored once broken out. **You alone are responsible** for any such modification and its consequences.

## Mounting Options

- **VESA 100 mm:** four standard M4 holes on the back panel, in a 100 mm × 100 mm pattern (Figure 4).
- **1/4"-20 tripod thread:** on the bottom of the housing, for camera tripods and compatible mounts (Figure 5). Make sure the 1/4" screw is sufficiently long to properly hold the device.

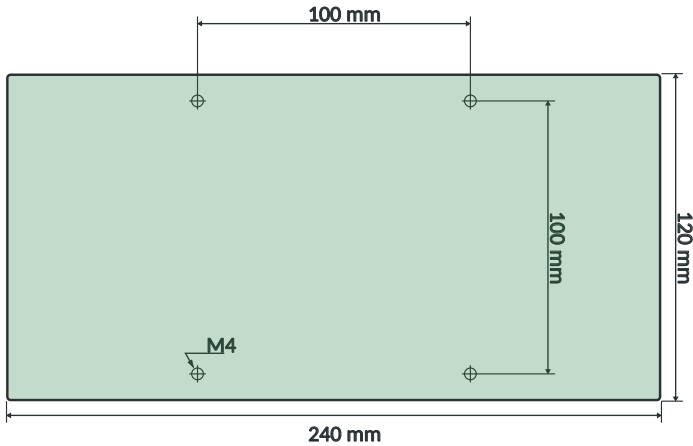


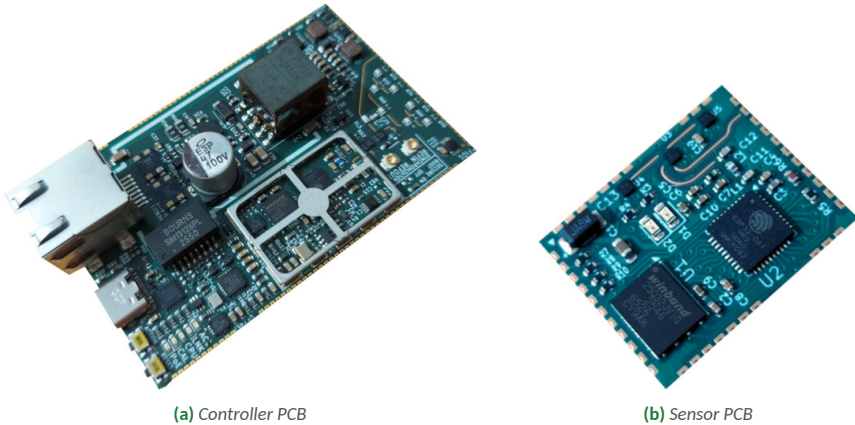
Figure 4. VESA 100 mm mounting-hole pattern (M4) on the back panel.



Figure 5. Underside of ESPARGOS One with the 1/4"-20 tripod mount highlighted.

## How It Works

### Sensors and Controller



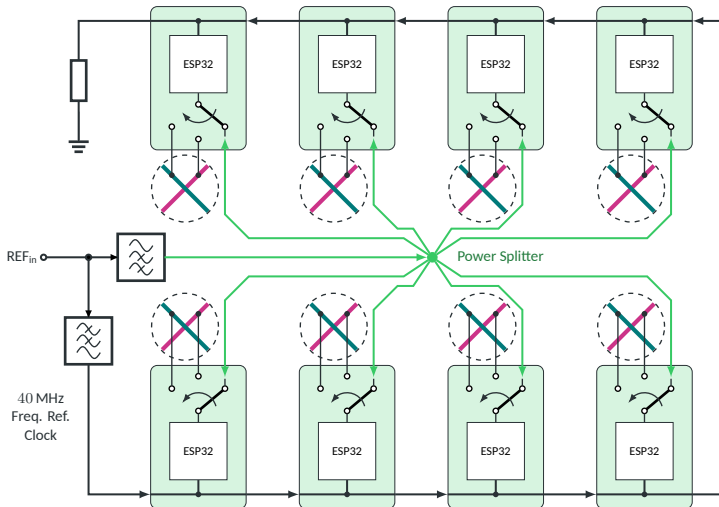
**Figure 6.** The modules in **ESPARGOS One**: the central controller and one of the eight sensor nodes.

**ESPARGOS One** is built from two types of modules: The eight **sensor** nodes each pair an ESP32 WiFi receiver with an RF switch, and are each connected to a dual-polarized patch antenna; they perform the actual CSI capture. The single, central **controller** is a separate ESP32 that configures and orchestrates all eight sensors, aggregates their CSI, and provides the outward-facing interfaces (Ethernet and USB-C). Throughout this manual, *sensor* always refers to one of the eight receive nodes, while *controller* refers to the central management ESP32. Both module types are shown in Figure 6.

### Synchronization Principle

All 8 sensor ESP32 chips share a common **40 MHz reference clock**. The remaining per-sensor LO phase ambiguity is corrected via the **2.4 GHz phase reference** distributed over PCB traces of known lengths. `pyespargos` estimates these offsets during a calibration step.

Each antenna element has two switchable feeds ( $\pm 45^\circ$  slant), enabling polarimetric analysis. While it is not possible to measure the channel on both feeds at the same time, the sensors can randomly switch between the feeds. Polarization can then be determined by exploiting short-term stationarity of the environment. The internal architecture of **ESPARGOS One** is shown in Figure 7.



**Figure 7.** Block diagram of the **ESPARGOS One** array. Each of the 8 sensor nodes contains an ESP32 WiFi receiver, an RF switch, and a dual slant-polarized patch antenna ( $+45^\circ$  and  $-45^\circ$  feeds). Green lines distribute the 2.4 GHz phase reference from the central power splitter to all nodes. The  $REF_{in}$  input carries both the 40 MHz frequency reference and the 2.4 GHz phase reference on one coaxial line, separated by a high-pass / low-pass filter pair.

## Data Capture

Each sensor streams its captured CSI to the controller over the **Serial Peripheral Interface (SPI)**. The eight sensors are split across two SPI buses, one per array row. The controller validates the per-sensor data and repacks it into frames that are then forwarded to the host. From the controller, CSI is forwarded to the host, over **Ethernet**, either over **UDP** or **WebSocket** streams.

Alternatively, the web interface and CSI can be tunneled over the **USB-C** serial port. A host-side bridge exposes the controller's HTTP and CSI interfaces locally, so the same tools work without a network connection. This path is bandwidth-limited (the serial link runs at a few Mbit/s, versus 100 Mbit/s over Ethernet): it is intended for initial setup and diagnostics on a machine without Ethernet access, not for sustained full-rate CSI streaming from all antennas.

Configuration and control use a **JSON RPC API**, served over HTTP on the Ethernet interface (port 80) and also tunneled over the USB-C serial port when no network connection is available.

**Good to Know** In practice, you will not need to worry about these low-level details: the `pyespargos` Python library (see page 18) abstracts all of this away and provides a simple API for configuration and real-time CSI streaming.

## Quick Start

### 1 Power up ESPARGOS One

You have two ways to provide power to **ESPARGOS One**:

- **Power over Ethernet (PoE)**: Connect an IEEE 802.3at Type 1 (Class 3) PoE injector or switch to the RJ45 port.
- **USB-C**: Connect a USB-C supply (or simply your computer) able to provide at least **5 V / 2 A**.

If both are connected, **ESPARGOS One** gives priority to PoE. Power and data are independent: you can power the device with PoE and stream data over USB-C, or power over USB-C and stream data over Ethernet.

### 2 Connect to ESPARGOS

Choose one of the following connection methods before opening the web interface.

#### ETHERNET Over Ethernet (recommended)

The default IPv4 configuration of **ESPARGOS One** is **192.168.1.2/16**.

**Operate on a Trusted Network Only** **ESPARGOS One** has **no access control**: its web interface, CSI streams, and configuration API are **fully open and require no authentication**, and the firmware can be updated over the network. Connect **ESPARGOS One** only to a **trusted, private network** (or directly to your computer), and never expose it to the open internet.

1. Connect your computer to the **ESPARGOS One** RJ45 port, directly or via a switch. Make sure no other device on the network uses **192.168.1.2**.
2. Set your computer's IP address to **192.168.1.100**, subnet mask **255.255.0.0**. Any address in the **192.168.1.0/16** subnet works.

**ESPARGOS One** also supports mDNS, so the configured hostname works too, as well as IPv6 (link-local and global addresses are shown in the web interface). The serial UART debug interface will also display the device's IP addresses after boot.

**Note** If you cannot connect, check that **192.168.1.2** is not already taken by another device. If it is still unreachable, perform a factory reset (hold the reset button 5–10 s) to restore the default network settings.

### USB-C Over USB (limited throughput)

The USB-C port is connected through a USB-to-serial converter, so its data throughput is much lower than Ethernet; use the Ethernet interface for demanding real-time applications. Tunnelling the web interface over USB uses the UART router script from `pyespargos` (if it is not yet installed, see page 18 first).

1. Connect **ESPARGOS One** to your computer via USB-C. It appears as a serial (UART) device, which by default only shows debug output.
2. Run the UART router, replacing `/dev/ttyUSB0` with the serial port that **ESPARGOS One** uses on your computer (e.g., `/dev/ttyACM0`, or COM3 on Windows):

```
python tools/espargos-uart-router.py uart:/dev/ttyUSB0
```

Then open the address shown by the router, e.g., <http://127.0.0.1:8400>.

## 3 Access the Web Interface

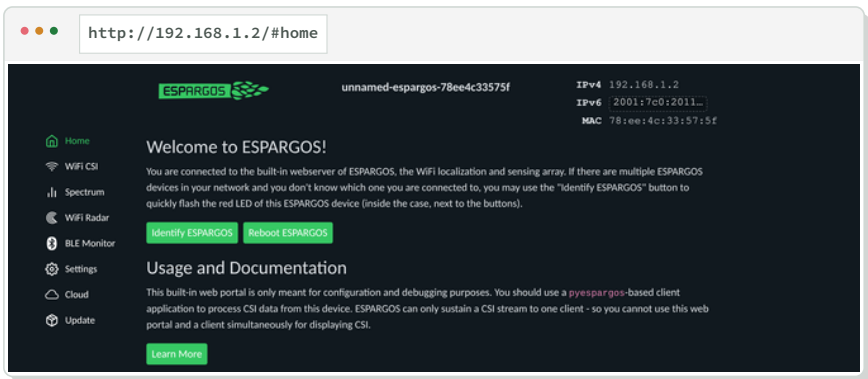


Figure 8. Home page of the built-in web interface.

Open the address that matches the connection method you chose (these are the defaults; they differ if you configured another IP address or port):

- **Ethernet:** <http://192.168.1.2>
- **USB UART router:** <http://127.0.0.1:8400>

You should be greeted by the home page of the web interface (Figure 8).

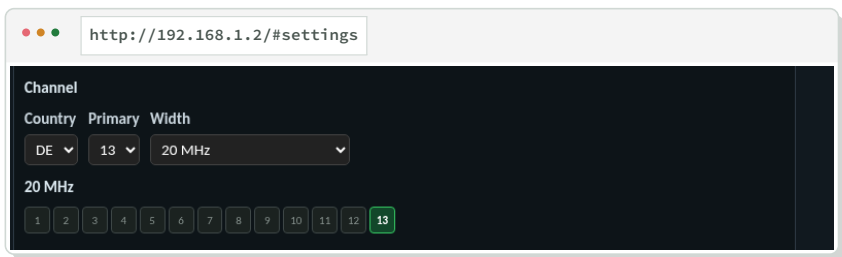
The web interface lets you view live CSI and change settings such as:

- WiFi channel and **country code**

- Network settings: IP address, hostname, mDNS
- Calibration and reference signal behavior
- Ephemeral configuration, such as receiver gain, MAC filters and similar
- Firmware updates (Ethernet only; see page 16)

**Regulatory Requirement** Set the correct **country code** in the web interface **before first use**. The country code determines which channels are permitted in your location (see also *Regulatory: WiFi Country Code*, page 28).

The WiFi country code and channel are set as shown in Figure 9.



**Figure 9.** Country code and WiFi channel selection in the web interface.

## Firmware Updates

### Web Update

**ESPARGOS One** can update itself over the internet (HTTPS), with no need to download or flash files by hand. On the **Update** page of the web interface, the device fetches the latest firmware directly from the ESPARGOS update server and installs every update (controller firmware, web interface, and all eight sensor nodes) in the correct order.

1. Make sure that your ESPARGOS network configuration provides internet access. This usually means configuring a suitable IP address / subnet mask / DNS server or using DHCP / automatic DNS.
2. Open the **Update** page and select an update channel (see below; **stable** is recommended for most users).
3. Click **Check for Updates**. The device contacts the server and proposes an update plan (Figure 10). This can take a few seconds.
4. Click **Start Web Update**. The selected components are downloaded and installed automatically; a progress page shows each step, and the device reboots on its own when finished.

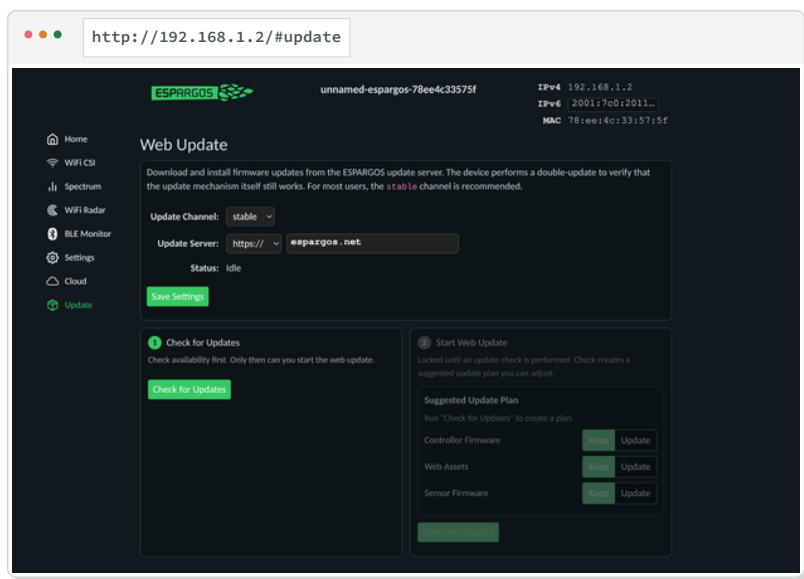


Figure 10. Automated web-update page with the proposed update plan.

A **double-update** mechanism verifies that the new firmware boots correctly before it is made permanent, which makes the update process resilient to most interruptions:

if an update is interrupted (e.g., by power loss), you can usually just repeat it. This is not an absolute guarantee, however; if the controller does become unresponsive, the manual recovery procedure (page 17) is required.

### Update Channels

Three release channels are offered on the update server; choose one with the **Update Channel** selector:

- **stable**: thoroughly tested releases. Recommended for normal operation.
- **testing**: release candidates for early validation, before they are promoted to **stable**.
- **dev**: the latest development builds; may be unstable and are intended for development only.

**Note** The default update server is <https://espargos.net>. In isolated networks you can point **ESPARGOS One** at your own mirror by changing the **Update Server** field, or use **Local Update** to upload firmware files manually.

### Firmware Recovery

If a firmware update fails, recovery is **always** possible.

1. Obtain the recovery firmware images and read the latest recovery instructions: <https://espargos.net/firmware#recovery>
2. **Controller**: Use `esptool` to flash the controller recovery image.
3. **Sensors**: Upload the sensor recovery partition via the web interface, then click *Provision All Sensors* (will take up to 10 minutes).

## Python Library: pyespargos

`pyespargos` is the official Python library for **ESPARGOS One**. It handles connection management, CSI de-serialization, phase calibration, clustering, and buffering, and can combine multiple arrays into larger apertures. The built-in web interface is intended for configuration and debugging only. To actually process CSI you should always use a `pyespargos`-based client.

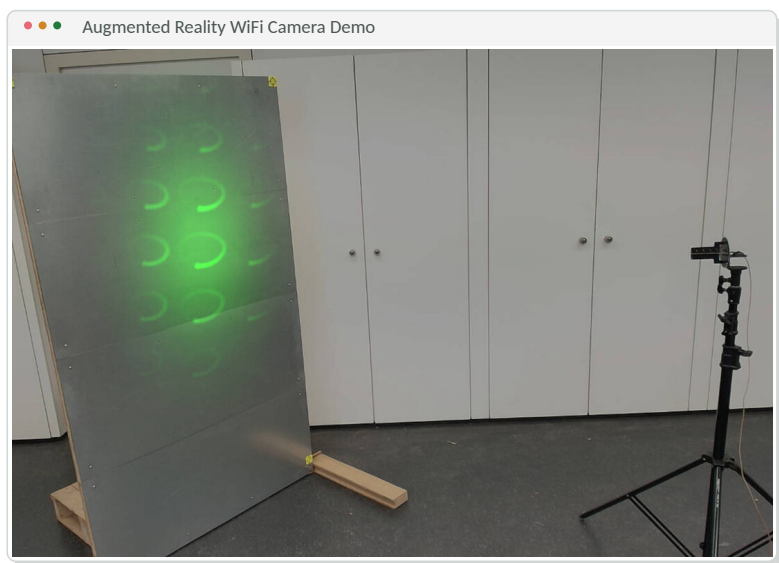


`pyespargos` also ships the UART router script (`tools/espargos-uart-router.py`) (see also page 13), which tunnels the web interface and the CSI streams over the USB-C serial port when no Ethernet connection is available.

Installation instructions, full API documentation, and ready-to-run demo applications (such as the augmented reality polarization visualizer shown in Figure 11) are available at:

<https://espargos.net/setup>

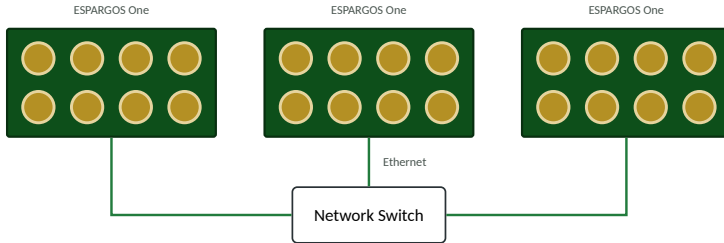
<https://github.com/ESPARGOS/pyespargos>



**Figure 11.** A `pyespargos` demo application visualizing the received signal's polarization in augmented reality. Here a circularly polarized transmitter is observed via a reflection off a metallic wall; the rotating dot patterns encode the polarization, whose handedness flips (LHCP  $\rightarrow$  RHCP) on reflection.

## Advanced Topics

### Single- and Multi-Board Setups with Unsynchronized Arrays



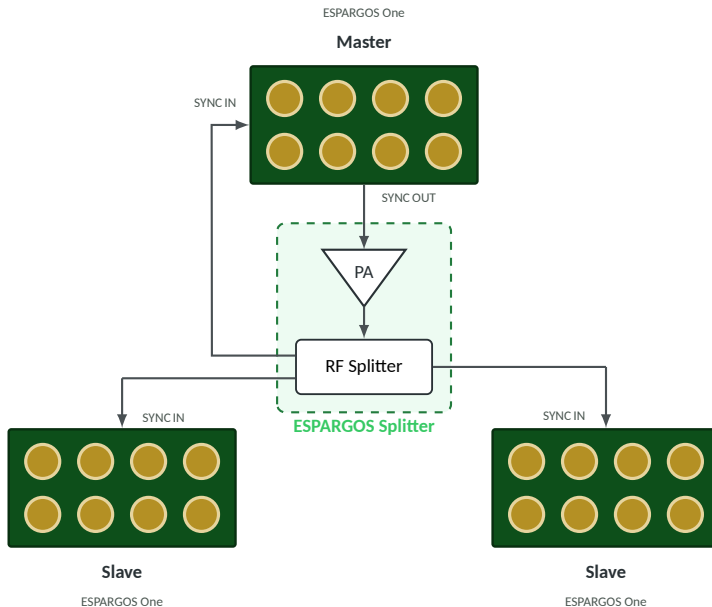
**Figure 12.** Three independent single-board **ESPARGOS One** arrays connected only by Ethernet cables to a common network switch. No synchronization cabling runs between them, so each board uses its own internal reference and the boards are not mutually phase-coherent. *pyespargos* still receives the CSI streams from all of them over the network.

**Individual, unsynchronized ESPARGOS One boards is the normal configuration, and the only one supported for the intended uses.** A standalone unit generates its own 40 MHz clock and 2.4 GHz phase reference internally and needs no external synchronization cabling: the **SYNC IN** and **SYNC OUT** ports stay unused, and the reference settings are left at their factory defaults (**Reference Signal Source** = *Internal*, **Generate Phase Reference** = *During Calibration*). These settings are shown in the **Reference Signal and Channel** card in web interface's settings (Figure 14) and summarized for every role in the table on page 21.

The trade-off is that each board derives its references from its own internal oscillator, so **separate boards are not mutually phase-coherent**: each carries an independent frequency and phase offset, and the CSI of several standalone boards (Figure 12) therefore cannot be combined into one larger coherent aperture. For many indoor localization applications, this is not a limitation.

This is because **angle-of-arrival (AoA)**-based localization relies only on the phase coherence *within* a single board and needs no synchronization between boards: several independent boards can each estimate AoA on their own. **Time-of-arrival (ToA)** localization, by contrast, compares signal timing across boards and therefore *does* require inter-board coherence, i.e., the multi-board setup described below.

## Multi-Board Phase-Coherent Systems (Research Only)



**Figure 13.** Distributed star-topology setup: the master's SYNC OUT drives a PA whose output feeds an RF splitter. One port returns to the master's own SYNC IN; the rest feed the slaves. *pyespargos* combines the CSI streams from all boards in software. For clarity, only the synchronization (SYNC) cabling is shown; the Ethernet cabling (as in Figure 12) is omitted.

Combining several boards into one larger aperture is an advanced feature intended for research use only.

To overcome the per-board offset, several **ESPARGOS One** units can share a common reference signal so that they become phase-coherent and their CSI can be combined coherently. One unit acts as the **master** and generates the reference; all others are **slaves** that lock onto it. The role is selected per unit on the same **Reference Signal and Channel** card (Figure 14); the settings and wiring for each role are summarized in the table on page 21.

The coherent boards can be arranged either as **one large-aperture array** (boards mounted close together to synthesize a single, larger aperture with finer angular resolution) or as **multiple distributed small arrays** (boards spread across the environment to observe it from several vantage points). In both cases the shared reference keeps all boards mutually phase-coherent, so their CSI can be combined.

- The master generates the shared reference (40 MHz clock + 2.4 GHz phase, with DC bias) and outputs it on **SYNC OUT**. It also receives the reference back on its

own **SYNC IN**, locking onto the same distributed signal as its slaves.

- Slaves use **SYNC IN** only. Because the master must also receive the reference back on its own SYNC IN, the SYNC OUT signal is *always* routed through an **RF power splitter** in a **star topology**, with one output per node *including the master* (Figure 13). For larger setups, insert a **power amplifier (PA)** between SYNC OUT and the splitter to offset its insertion loss.
- **All cables from the splitter outputs to the nodes must be identical in length and type.** Any difference in electrical length introduces a fixed phase offset that would require additional calibration steps, which are currently not supported in software.

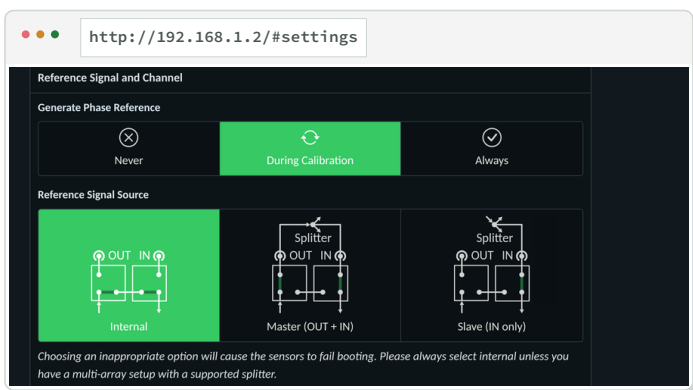
The RF Splitter must be sufficiently wideband to split both the 40 MHz clock signal and the 2.4 GHz phase reference signal cleanly. Most resistive splitters will be sufficiently wideband.

**Note** For one master and one slave unit, the PA is not necessary; a simple low-cost resistive RF splitter provides sufficient calibration signal amplitude.

**Note** For up to 4 devices (one master, three slaves), an **ESPARGOS Splitter**, a single unit combining the PA and RF splitter, is available from the manufacturer upon request.

## Configuring the Reference Signal

Once the units are wired, each one's role is selected on the **Settings** page of the web interface, in the **Reference Signal and Channel** card (Figure 14). Two settings govern the multi-board behavior:



**Figure 14.** The **Reference Signal and Channel** settings card: per-unit selection of the reference signal source (Internal / Master / Slave) and when the phase reference is generated.

- **Reference Signal Source** selects how the unit obtains the 2.4 GHz phase reference:
  - **Internal:** standalone single array (default). The unit uses its own internal generator; the SYNC ports are unused.
  - **Master (OUT + IN):** the unit generates the shared reference and drives **SYNC OUT**, and also locks onto the distributed signal via its own **SYNC IN**.
  - **Slave (IN only):** the sensors lock onto an external reference received on **SYNC IN**.
- **Generate Phase Reference** sets when the reference packets are emitted:
  - **During Calibration:** emitted only while the sensors measure the reference channel; use this for *Internal* and *Master* units.
  - **Never:** the unit never emits the reference; use this for *Slave* units, which receive it from the master.
  - **Always:** continuous emission, intended for debugging only.

The table below summarizes the settings and wiring for each role:

Setting	Single-board (default)	Multi-board master	Multi-board slave
<b>Reference Signal Source</b>	Internal	Master (OUT + IN)	Slave (IN only)
<b>Generate Phase Reference</b>	During Calibration	During Calibration	Never
<b>SYNC OUT</b>	unused	drives reference into splitter	unused
<b>SYNC IN</b>	unused	reference returned from splitter	reference from splitter
<b>Intended use</b>	normal, supported	research only	research only

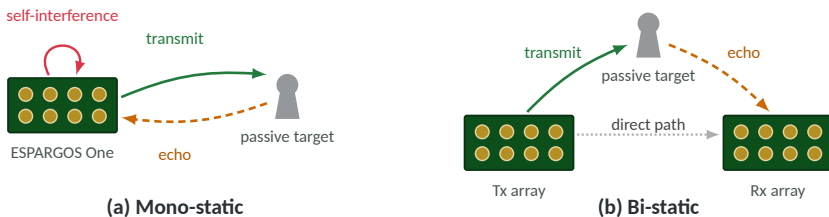
Selecting a source that does not match the physical wiring (e.g., *Slave* with no signal present on SYNC IN) will cause the sensors to **fail to boot**, so leave the setting at **Internal** unless a supported splitter is connected.

**Avoid Unintended Emissions on SYNCOUT** A unit configured as master **actively drives** SYNC OUT with strong **40 MHz** and **2.4 GHz** signals. An open, unterminated, or poorly shielded SYNC OUT can **radiate** the 2.4 GHz signal like an antenna, an unintended transmission over multiple bands that may interfere with other equipment or violate local regulations. Never run a master with SYNC OUT left open: always route it through shielded SMA cabling into a splitter and slave(s), or terminate the port. The same operator responsibility applies as for deliberate transmission (page 23).

## Transmitting With ESPARGOS / Radar

**ESPARGOS One** is designed as a **passive receiver**: in normal operation it only listens to ambient WiFi traffic and never transmits. Some advanced research applications, however, such as active WiFi radar, require the array to *transmit* radio signals, and the **ESPARGOS One** hardware is technically capable of doing so.

For regulatory reasons, **transmission is disabled by default**. It can be enabled only deliberately, in the web interface, where the operator must assert that they are legally authorized to transmit on the selected channel and power (e.g. as a licensed radio amateur). Until then, the device does not transmit.



**Figure 15.** Active WiFi radar. **(a) Mono-static:** one array transmits and receives the echo from a passive target; possible, but limited by self-interference (the transmitting sensor leaks into the receiving sensors). **(b) Bi-static:** a separate array receives the echo; the residual direct Tx-to-Rx path is much weaker and geometrically separable, so this works better.

The array transmits from one sensor at a time, so it does not form a phase-coherent transmit beam in real time; thanks to the array's phase coherence, however, an equivalent beam can be *synthesized* in post-processing (**MIMO radar** / virtual-array beam-forming). Neither the firmware nor **pyespargos** provides **radar signal processing**; this research-only mode requires you to implement the processing yourself.

**Transmission: Operator Responsibility** Enabling transmission is done **entirely at your own risk and sole responsibility**. As with a software-defined radio, the hardware is capable of emitting radio signals, but **you alone are responsible** for ensuring that any transmission complies with all applicable laws, regulations, frequency allocations, licensing requirements, and power limits in your jurisdiction.

Enabling the transmission setting in the web interface does **not** certify regulatory compliance and grants no authorization of any kind; it merely records that you have asserted that you are authorized to transmit. The manufacturer and seller make **no representation** that transmission is lawful in your location.

The manufacturer and seller accept **no liability whatsoever** for any use of the transmission capability, including any interference, damage, regulatory penalties, or legal consequences arising from it. Operating a radio transmitter without the required authorization may be unlawful and subject to prosecution. If you are in any doubt as to whether you are permitted to transmit, **leave transmission disabled**.

## Cleaning & Maintenance

Routine exterior cleaning is the *only* maintenance **ESPARGOS One** requires; there are no user-serviceable parts inside (see *ESD & Servicing*, page 28). The front face of the device is the exposed antenna PCB, so fingerprints, dust, or grease on the patch antennas are the most common reason to clean it.

### To clean the antenna face and enclosure:

1. **Disconnect power** (and any cables) first.
2. Wipe gently with a **soft, dry, lint-free cloth** (e.g. microfibre) to remove dust and most fingerprints.
3. For stubborn grease or fingerprints, lightly **dampen the cloth with isopropyl alcohol** (IPA) and wipe. Apply the liquid to the cloth, *never* directly to the device.
4. Let **ESPARGOS One** **dry completely** before reconnecting power.

A few further points keep the device in good order:

- Keep the protective caps on the unused **SYNC IN/SYNC OUT** SMA ports so dust and debris stay out of the connectors.
- Periodically check that the **mounting screws are tight**, especially for overhead or wall installations.
- Keep the enclosure free of heavy dust and do not cover it during operation, so it can dissipate heat.
- Keep the firmware up to date for the latest fixes and features (see *Firmware Updates*, page 16).

**Do Not Wash or Soak** **ESPARGOS One** is **not water-resistant** and is for indoor use only. Do **not** immerse, rinse, or hold it under running water, and do **not** spray cleaning liquid onto it; always dampen the cloth instead, so that no liquid enters the seams, connectors, camera hole, or cutouts. Avoid **acetone**, aggressive solvents, and abrasive cleaners or pads, which can craze the enclosure or scratch the antenna PCB.

## Troubleshooting

Common problems are listed below as, followed by the checks and solutions that usually resolve them. When in doubt, a reboot or, as a last resort, a factory reset (hold the reset button 5–10 s) clears most transient issues (it will *not* reset the firmware to factory state, however, only the configuration). Contact support at [info@espargos.net](mailto:info@espargos.net) if you cannot resolve your problem by yourself, make sure to include details of your setup and the troubleshooting steps you have taken.

### 1. No power at all: all LEDs remain off

- The status LEDs are **very faint inside the case**: look through the gaps next to the network and USB ports before assuming the device is dead.
- Check the power source: the PoE injector or switch is switched on and compliant with **IEEE 802.3at Type 1 (Class 3)**; “passive” PoE injectors are **not supported**. If using USB-C, the supply must be rated for **5 V / 2 A** and properly connected.

**Disconnect if it gets warm** If the device does not power on but becomes warm, **disconnect power immediately** and contact support.

### 2. Powered on but unreachable: no ping response / web interface inaccessible

- Check that your computer’s IP address is in the **same subnet** (e.g., 192.168.1.x, mask 255.255.0.0). You should then be able to ping the device.
- Check the physical link: the Ethernet port is **100BASE-TX**. Confirm the link LED comes up, and try another cable or switch port if it does not.
- If you do not know the device’s IP address, try its **mDNS hostname**; **ESPARGOS One** advertises itself on the local network via mDNS.
- Alternatively, read the device’s IP addresses from the **UART debug output** on the USB-C serial port (**115200 baud**); they are printed shortly after boot.
- Use **http://**, not **https://**; the web interface is served over plain HTTP.
- If several **ESPARGOS One** units are present, check for **IP conflicts** and clear your computer’s **ARP cache**.
- No Ethernet? Connect over **USB-C** and use the serial bridge to reach the web interface locally (see *Over USB*, page 14).
- Still unreachable? Perform a **factory reset** (hold the reset button 5–10 s until reboot) to restore the default network settings (see *Buttons & LEDs*, page 9).
- Still no response? Use a serial terminal to watch the **UART output over USB-C** for status and error messages during boot. Anything suspicious? Perform a firmware recovery (page 17).

### 3. Reachable but no WiFi CSI is captured

- First, check whether the **sensor list** on the web interface's home page is empty. If it is, no sensors are being detected; see symptom 9.
- Ensure that no other client is currently streaming CSI, as only one client can access the stream at a time. If in doubt, reboot **ESPARGOS One** to clear any existing connections.
- Ensure the correct **WiFi country code** is set and the **channel** matches one that is allowed in your environment (see page 14, Step 3), with a WiFi transmitter present on that 2.4 GHz channel.
- Set the per-element **RF switch to "REF" mode** to confirm that the reference signal arrives. If the reference is missing,
  - ensure that the **receiver gain** is set to **Auto** and
  - make sure "Generate Phase Reference" is set to "During Calibration" and "Reference Signal Source" to "Internal" (unless you have a multi-board setup, in which case see page 21 for the correct settings).
- If the reference signal is present but over-the-air CSI is still missing,
  - ensure that the transmitter channel and the **ESPARGOS One** channel are the same,
  - ensure that the **receiver gain** is set to **Auto**,
  - ensure that CFO correction is enabled and
  - ensure that the transmitter uses an OFDM modulation scheme supported for CSI reception (802.11g/n/ax). Note that 802.11b will be received and displayed, but CSI will not be available for it.
- If nothing helps, try a firmware update or recovery (page 16).

### 4. CSI is captured but appears very noisy or unstable

- **RF switch:** make sure the RF switch is in either "+45°" or "-45°" mode, not Isolated, REF, or random.
- **Polarization:** experiment with the polarization setting of the RF switch. Some software applications can incorporate both polarizations (randomized RF switch setting), which leads to better results.
- Weak signals can cause noisy CSI. Check whether a stronger signal gives better results.
- The transmitter may use beamforming itself: access points in particular can appear to have unstable CSI when they actively steer their beam toward or away from the array.

### 5. Device is slow to react: lag between movement and measurement

- The device may buffer CSI measurements: if the link to the host computer, or the host computer itself, is not fast enough, packets arrive late.
- Switch to Ethernet if possible: it has much higher throughput than USB.
- Use `pyespargos` (see page 18) instead of the web interface. `pyespargos` uses the faster UDP transport mode, which supports hundreds of packets per second.

### 6. Flashing the controller recovery image fails

- Make sure that your computer can provide enough power over USB-C.
- Use an up-to-date `esptool`: support for hardware flow control-enabled USB-UART adapters was added only in May 2026.

### 7. The module LEDs are behaving unexpectedly

- The expected sensor module LED behavior is:
  - Right polarization: Top row is **red**, bottom row is **green**
  - Left polarization: Top row is **green**, bottom row is **red**
  - Calibration: Both rows show both **green** and **red**
  - Isolation, web update or recovery partition flashing: Both rows are off
  - Other patterns appear in special states, e.g., fast firmware updates or transmission
- If the sensor module LEDs blink uncontrollably or do not respond to settings, please ensure that the clock source setting is chosen appropriately, see page 21. “Internal” should always work.

### 8. After an update, the web interface shows “Web assets unavailable.”

- Power cycle the device and plug it into the same network that it was plugged into for the software update. It will try to download the web assets and recover automatically. This can take up to 5 minutes.
- If that does not help, perform controller firmware recovery (page 16).

### 9. The list of sensors on the web interface’s home page is empty.

- If you recently updated any firmware, please perform a complete web firmware update again. When asked to configure the update plan, choose “Update” (not “Keep”) on all items (controller firmware, web assets and sensor firmware).
- For a single-array setup, set the **clock source** to “Internal”; without a valid clock the sensors are not detected (see page 21).
- Through the gaps in the case, try to spot the LEDs of the sensors to better understand what is going on (see symptom 7).

## Safety, Regulatory & Environmental Information

### Electrical Safety

**ESPARGOS One** may receive up to **57 V DC** via IEEE 802.3at PoE. Use only IEEE 802.3at-compliant PoE equipment. Do not handle cables or connectors in wet conditions, and do not short-circuit or pierce any connector.

**Disconnect Power Immediately** If the enclosure shows physical damage, smoke, or unusual heat, **disconnect power immediately**.

### Handling

The enclosure and the exposed edge of the antenna PCB can have **sharp edges**. Handle **ESPARGOS One** with care, particularly when mounting or removing it, to avoid minor cuts or abrasions.

### Regulatory: WiFi Country Code

Operating on channels or at power levels not permitted in your location is illegal and may cause interference with other radio systems. Set the correct country code before first use; see page 14 (Quick Start, Step 3).

### ESD & Servicing

The PCB contains ESD-sensitive RF components. If the enclosure must be opened for authorized purposes, use an anti-static wrist strap and work on a grounded mat. **ESPARGOS One** contains **no user-serviceable parts**; unauthorized repair voids the warranty and may create safety hazards. Contact [info@espargos.net](mailto:info@espargos.net) for service.

### Ground Loops

The shield of the **RJ45 Ethernet** connector is tied to the device ground, as is the ground of the **USB-C** port. A **ground loop** can therefore form if **ESPARGOS One** is connected over both interfaces at the same time and *both* of them provide a path to earth ground, for example a USB-C link to a mains-powered, earthed computer while Ethernet runs to earthed network equipment. Ground-loop currents can introduce noise that degrades measurement quality and, in extreme cases, stress the connectors.

In practice this is rarely a problem: most standalone USB power supplies are **galvanically isolated**, so powering **ESPARGOS One** from a USB charger while using Ethernet does *not* create a loop. If you do observe interference, power and connect **ESPARGOS One** through a **single interface** (PoE/Ethernet or USB-C), or insert a USB isolator or an isolated supply to break the second path to ground.

## Operating & Storage Conditions

<b>Operating temperature</b>	0 °C to +50 °C
<b>Storage temperature</b>	-20 °C to +70 °C
<b>Relative humidity</b>	20 % to 80 % (non-condensing)

**ESPARGOS One** is intended for **indoor use only** and must not be exposed to rain, moisture, or outdoor weather conditions.

### Intended for Use Within the European Union

**ESPARGOS One** is designed, configured, and supplied for sale and use **within the European Union (EU) only**. It has not been prepared, tested, or authorized for use in other regions, where different radio, electrical, safety, or other regulatory requirements may apply.

By purchasing **ESPARGOS One**, the buyer agrees **not to export, re-export, ship, carry, or otherwise transfer the device, or permit it to be transferred, outside the European Union**, whether directly or indirectly, and to impose this same restriction on any subsequent recipient. The buyer alone is responsible for compliance with all rules that may apply to moving the device across borders. If use outside the EU is required, please contact the manufacturer before any such transfer.

### Lawful Use: Privacy, Interception & Data Protection

**Lawful Use Is the Operator's Responsibility** **ESPARGOS One** is a passive WiFi sensing array: in normal operation it receives and processes ambient WiFi signals that are **not necessarily addressed to the operator**, and it can be used to detect, localize, or characterize the presence and movement of people and devices. Depending on how and where it is deployed, this may be subject to laws on the **interception of communications and telecommunications secrecy** (e.g., anti-wiretapping statutes) and on **data protection and privacy** (e.g., the EU GDPR), because captured data, such as device identifiers or patterns of presence, can constitute personal data.

**You, the operator, are solely responsible** for ensuring that your use of **ESPARGOS One** complies with all applicable laws, including obtaining any required legal basis, authorization, notice, or consent, and for acting as the responsible data controller where applicable. The manufacturer and seller make **no representation** that any particular use is lawful and accept **no liability whatsoever** for unlawful or non-compliant use. If in doubt, obtain legal advice before deploying.

### Security & Vulnerability Reporting

**ESPARGOS One** exposes a network interface and an automatic firmware-update mechanism. It is designed to be operated on a **private, secured network** under your control and must not be exposed directly to the public internet. Operate it only on trusted networks and keep the firmware up to date (see also the trusted-network warning on page 13). Future firmware revisions are planned to add **authenticated and en-**

**rypted interfaces** and **secure boot** to further harden the device. To report a security vulnerability, please contact [security@espargos.net](mailto:security@espargos.net).

### REACH: Substances of Very High Concern (Art. 33)

Under Article 33 of the EU REACH Regulation (EC) No 1907/2006, suppliers must inform recipients if an article contains a Substance of Very High Concern (SVHC) from the ECHA Candidate List in a concentration above 0.1 % by weight. To the best of the manufacturer's knowledge, **ESPARGOS One** does not contain such substances above this threshold, beyond any entries inherent to standard electronic assemblies.

### Cabling

Connect **ESPARGOS One** using **shielded** cables. For the network/PoE link, use a shielded Ethernet cable (for example Cat 6 S/FTP or F/UTP) with metal-shielded RJ45 plugs; for USB-C, use a fully shielded USB-C cable. The Ethernet cable supplied with **ESPARGOS One** is shielded and may be used.

### CE Conformity & Electromagnetic Compatibility (EMC)

**ESPARGOS One** bears the **CE** mark and conforms with the applicable EU harmonization legislation, in particular the **EMC Directive 2014/30/EU** and the **RoHS Directive 2011/65/EU**. Conformity with the EMC Directive has been demonstrated by emission testing to **DIN EN 55032** and immunity testing to **DIN EN 55035** (using the EN 61000-4-2, -4-3, -4-4 and -4-6 basic standards); RoHS technical documentation follows **DIN EN IEC 63000**. The full text of the EU Declaration of Conformity is available at [espargos.net/ce](http://espargos.net/ce).

**Class A Equipment (DIN EN 55032)** **ESPARGOS One** is a **Class A** product. In a residential environment it may cause radio interference, in which case the user may be required to take adequate measures. **ESPARGOS One** is intended for **professional and research use only**.

### Open-Source Software & License Notices

The **ESPARGOS One** firmware is built on **Espressif ESP-IDF**, licensed under the **Apache License 2.0**, which itself incorporates third-party open-source components, including **FreeRTOS** (MIT License), **lwIP** (BSD 3-Clause License), and **Mbed TLS** (Apache 2.0). In addition, the firmware uses **cJSON** (MIT License) and several Espressif libraries and drivers: the **mdNS** component, the **LAN87xx** Ethernet PHY driver, and **esp-serial-flasher**, all under the **Apache License 2.0**.

These components remain under their respective licenses; the corresponding copyright notices and full license texts are available from the respective upstream projects and, on request, from the manufacturer. All other firmware and software are

© TTI GmbH / TGU ESPARGOS

unless stated otherwise.





## Manufacturer

TTI GmbH an der Universität Stuttgart  
TGU ESPARGOS  
Nobelstraße 15, 70569 Stuttgart, Germany

**Web:** <https://espargos.net>


**Email:** [info@espargos.net](mailto:info@espargos.net)

**Model:** ESPARGOS One

**Batch:** 2025-12

**Country of Manufacture:**  
Germany

## Compliance

**CE Marking**  DIN EN 55032, DIN EN 55035, DIN EN IEC 63000

The CE symbol confirms that this product conforms with the above mentioned standards and relevant union harmonization legislation. The full text of the declaration of EU conformity is available at <https://espargos.net/ce>. See *CE Conformity & Electromagnetic Compatibility* (page 30) for details.

**WEEE**



Do not dispose with ordinary household waste. Return to an authorized WEEE collection point. (Directive 2012/19/EU)

## Warranty

12 months from delivery date; covers manufacturing and material defects present at time of delivery. Seller's remedy at sole discretion: repair, replacement, or pro-rated credit. Excludes damage caused by misuse, unauthorized modification, or operation outside specified conditions.

## Inspection & Notice of Defects (§ 377 HGB)

This product is sold exclusively business-to-business. Buyer must inspect goods on receipt. Apparent defects must be notified in writing within 5 business days of delivery; hidden defects within 5 business days of discovery, no later than 13 months after delivery. Claims not reported within these periods are waived.

## Limitation of Liability

Aggregate liability shall not exceed the net invoice price. Liability for indirect, consequential, or incidental loss — including loss of data, revenue, or profit — is excluded, save where caused by wilful misconduct or gross negligence. Statutory product liability and liability for personal injury remain unaffected.