Commissioning an openRemoteTRX station: operator guide

This guide takes you, step by step, from unpacking to the first link from outside. It applies to any transceiver known to Hamlib; what is specific to one model (USB driver, menus, ports) is in its **transceiver sheet**:

- Icom IC-7300MK2: **validated** on a real transceiver;
- Yaesu FTDX10: prepared, **validation in progress**.

For another transceiver: its manual, the “Depending on your brand” table of the quick start, and the tests of part 3, which serve as validation: do them all, on a dummy load, and report the result (part 8, “Asking for help”). Status of each transceiver tested: https://bob56fr.github.io/openRemoteTRX/postes/

> **In a hurry?** The quick start takes you from the connected transceiver to the web page in a quarter of an hour, transceiver TOT included. Come back to the parts below at the first problem.

The path

Part	What you do	Approximate time
Quick start	the essentials for the impatient, safety tests included	15 min
0. Before you start	hardware, regulations, choosing Debian, Raspberry Pi or Windows	15 min of reading
1. Prepare the transceiver	cabling, USB driver, transceiver menus	30 min
2. Install the station	software on the PC or the Raspberry Pi	20 min (Pi: 1 h with the SD card)
3. Integrate the TRX and test it	wizard, receive then transmit tests, safety tests	1 h 30
4. Use on the local network	web page, phone, Android app	30 min
5. Use remotely	VPN or exposed port, test over 4G	1 h
6. Unattended station	alerts, restart after a power cut, protections	30 min
7. Day-to-day operation	start, monitor, update, back up	to read
8. Troubleshooting	error codes, common problems, asking for help	when needed
9. Optional features	visitors, DX Cluster, FreeDV, SELCALL, SWR…	as you wish
Glossary	the project's words: who is what	when needed
10. Checklist	know where you stand	

Transceiver sheets: Icom IC-7300MK2, Yaesu FTDX10. Diagram of the station's building blocks: part 0, “How it is organised”.

How to read this guide

- The **openRemoteTRX Console menus** are given by their number: “menu 2.1” = theme 2 (Transceiver and hardware), menu 1. On Windows, the shortcut Start menu > **openRemoteTRX** opens the same console (part 2). The console opens in **full screen**: themes on the left with their icon (arrows and Enter), help for the selected menu on the right; you can also type the number in the text console (`trx-console --texte`).
- A **“Safety”** box marks a point that concerns transmission: do not skip it.
- What changes with the system is arranged in **tabs** Debian / Raspberry Pi / Windows: choose yours once, it is remembered for the whole page. On GitHub, the tabs are shown one below the other.
- This guide also exists as a web page, with a checklist you tick online, and as a **PDF** to print (link “Download as PDF” at the top of the page): https://bob56fr.github.io/openRemoteTRX/guide/en/

Quick start

For the impatient: your transceiver in the browser in a quarter of an hour. The rest of the guide explains each step in detail; come back to it at the first problem.

You need

- a transceiver known to Hamlib, connected to the PC: a USB cable for most recent transceivers (control and audio), otherwise an interface such as Digirig or SignaLink;
- a PC running **Debian 12 or 13** or **Windows 10 / 11** 64-bit, or a **Raspberry Pi 4 or 5** (Raspberry Pi OS 64-bit, official power supply), always on near the transceiver, connected to the internet box;
- a **dummy load** for the first transmission;
- a phone or a PC with a recent browser.

0. First, the transceiver TOT (2 min)

> **Safety.** The **TOT** (*Time Out Timer*) is the timer that cuts the transmission after a maximum time. Set the transceiver's one **before anything else**.

Why? Remotely, you cannot see your transceiver. If the station PC freezes, if the USB cable comes loose or if a program crashes during a transmission, nobody can put the transceiver back to receive any more: it would keep transmitting a carrier, block the frequency, heat up, and you would remain responsible.

What it is for. The station has its own TOT (90 s by default) and a supervisor that watches it. The transceiver TOT is the **last barrier**: it is inside the transceiver itself, it cuts even when the PC, the cable or the station has failed.

How. In the transceiver menu, look for “Time-Out Timer” (“Time-Out Timer (CI-V)” on Icom) or “TX TIME OUT TIMER” and choose **3 minutes**: longer than the station TOT, it never cuts a normal transmission.

The menu path for your brand is in the table below. On the IC-7300MK2, the station reads this setting back at each connection and warns you if it is OFF.

Depending on your brand

What changes from one transceiver to another, in one line; the details are in the transceiver sheet when there is one.

	Icom	Yaesu	Kenwood	Xiegu
USB driver (Windows)	“Icom USB Driver”, version **1.80 or later**	Silicon Labs **CP210x**; “Enhanced” port for control	manufacturer's driver	manufacturer's driver
Transceiver TOT	MENU > SET > Function > Time-Out Timer (CI-V)	OPERATION SETTING > GENERAL > TX TIME OUT TIMER	transceiver manual	transceiver manual
Voice over USB	MENU > SET > Connectors > DATA MOD = **USB** (transmit in USB-D / LSB-D)	MOD SOURCE = **REAR** and REAR SELECT = **USB**, in each mode	transceiver manual	transceiver manual
Status	**validated** (IC-7300MK2)	**in progress** (FTDX10)	not tested yet	not tested yet
Sheet	IC-7300MK2	FTDX10	—	—

On Linux (Debian, Raspberry Pi), no driver to install. For a transceiver not tested yet: install the USB driver from its manufacturer's website, and do the three tests of step 5 with particular care.

1. Install (5 min)

Download the package of the latest version: https://github.com/Bob56fr/openRemoteTRX/releases

Debian

Extract `trx-remote-debian-X.Y.Z.zip`, open a terminal in the folder obtained, then

```
sudo bash install.sh
```

Log out and log back in at the end.

Raspberry Pi

Card prepared with Raspberry Pi Imager (Raspberry Pi OS 64-bit, SSH enabled) and system up to date: part 2, Raspberry Pi tab. Then the **same package as Debian**: with a desktop, exactly as on Debian; without a desktop, from your PC:

```
scp -o User=<account> trx-remote-debian-X.Y.Z.zip <pi-address>:
ssh -t -l <account> <pi-address>
```

and on the Pi: `unzip -q trx-remote-debian-X.Y.Z.zip && cd trx-remote-debian-X.Y.Z && sudo bash install.sh`. Log back in at the end.

Windows

1. First install **Hamlib 64-bit** 4.7 or later (`hamlib-w64-X.Y.Z.exe`): https://github.com/Hamlib/Hamlib/releases
2. Right-click `trx-remote-windows-X.Y.Z.zip` > Properties > **Unblock**, then Extract All and double-click **`installer.cmd`**.
3. Home network set to the **Private** profile, **sleep disabled**.

2. Configure (5 min)

Open **openRemoteTRX Console** (Debian: applications menu or `trx-console`; Raspberry Pi without a desktop: `ssh -t -l <account> <pi-address> trx-console`; Windows: Start menu > **openRemoteTRX**), then **menu 2.1** (the wizard). The transceiver is only read: no transmission. What matters:

Question	Answer
Model	type the transceiver name (e.g. `ic-7300`, `ftdx10`), choose from the list
Port and speed	the transceiver port; the speed set in the transceiver's CAT menu
PTT	**CAT** if the wizard offers it
Sound cards	the transceiver's one, for receive and for transmit
Allowed bands	the bands of **your licence**, nothing else. Empty list = receive only
Maximum duration of a transmission (station TOT)	**90** seconds (offered, 120 at most)
Network access	“Local network or VPN, over HTTPS” (offered)
Account	a username; Enter on the password generates a strong one: **write it down**

For everything else, Enter keeps the offered value.

3. Start and check (2 min)

1. **Menu 1.1**: the station starts, transceiver in receive.
2. **Menu 6.1** (diagnostic): fix each line in **failure** before going further; the line says what to do.

4. Connect (1 min)

1. On the phone or a PC at home: `https://<station-address>:8443` (address shown at the top of openRemoteTRX Console).
2. Certificate warning: “Advanced” then “Proceed”. This is normal (certificate made by your station); to stop seeing it: menu 4.4.
3. Log in with the wizard's account. No sound: tap the sound button, the browser had blocked it.

Reserve the PC's address in your internet box (DHCP reservation): if it changes, the certificate is no longer valid.

5. Before the first transmission: three tests (3 min)

> **Safety.** Dummy load connected, reduced power, a band of your licence. The **PTT is a toggle**: one press to transmit, one press to go back to receive. After each test, look at the **transceiver's display**, not only the page.

Test	Expected
During a transmission, **STOP** button	immediate receive
During a transmission, close the tab (or turn off the phone's Wi-Fi)	receive in about 2 s
Frequency outside your bands, then PTT	transmission **refused**, message in the page

A single failed test: **do not transmit**, stop the station (menu 1.2) and read part 8. The nine complete tests are in part 3.5: do them before transmitting on an antenna.

Next

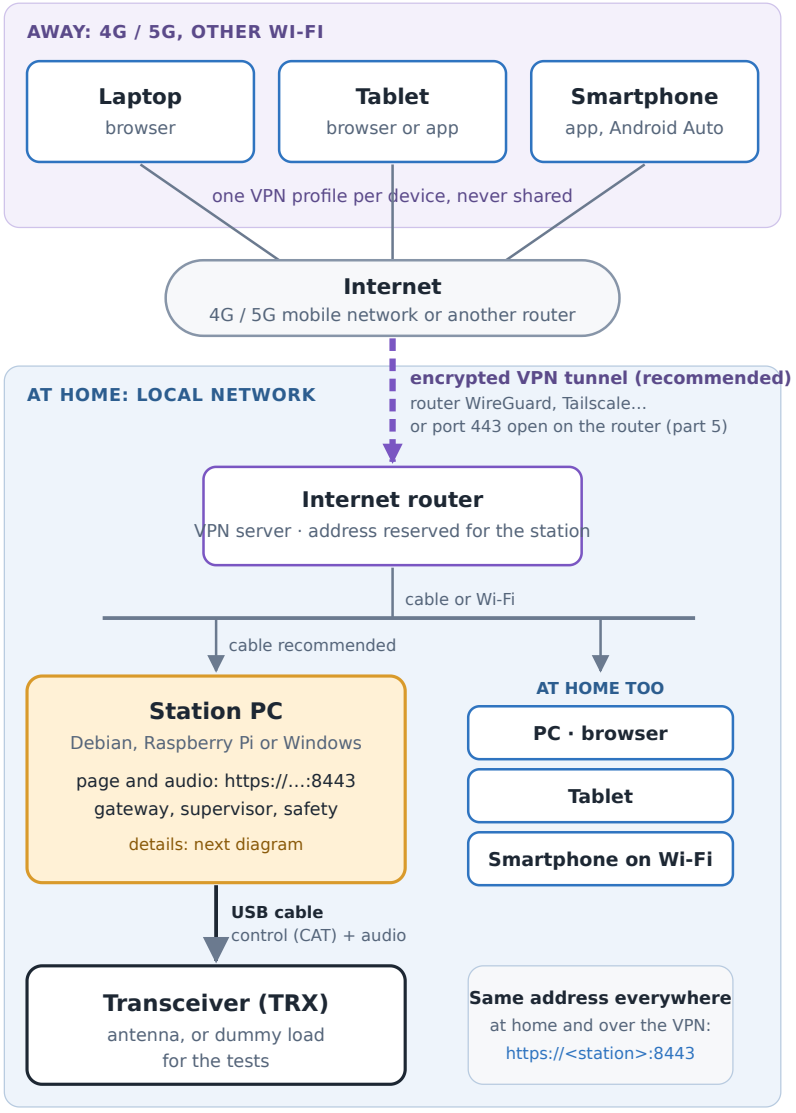
- **Android app**: menu 4.1, QR code to scan (app in invitation-only testing, part 0).
- **From outside** (4G, another network): VPN or open port, part 5.
- **Alerts on the phone**: menu 3.5, with a test alert.
- **A problem**: menu 6.1, then the report from menu 6.3 to attach to a request for help (part 8).
- **All the console menus**: https://bob56fr.github.io/openRemoteTRX/console/en/

0. Before you start

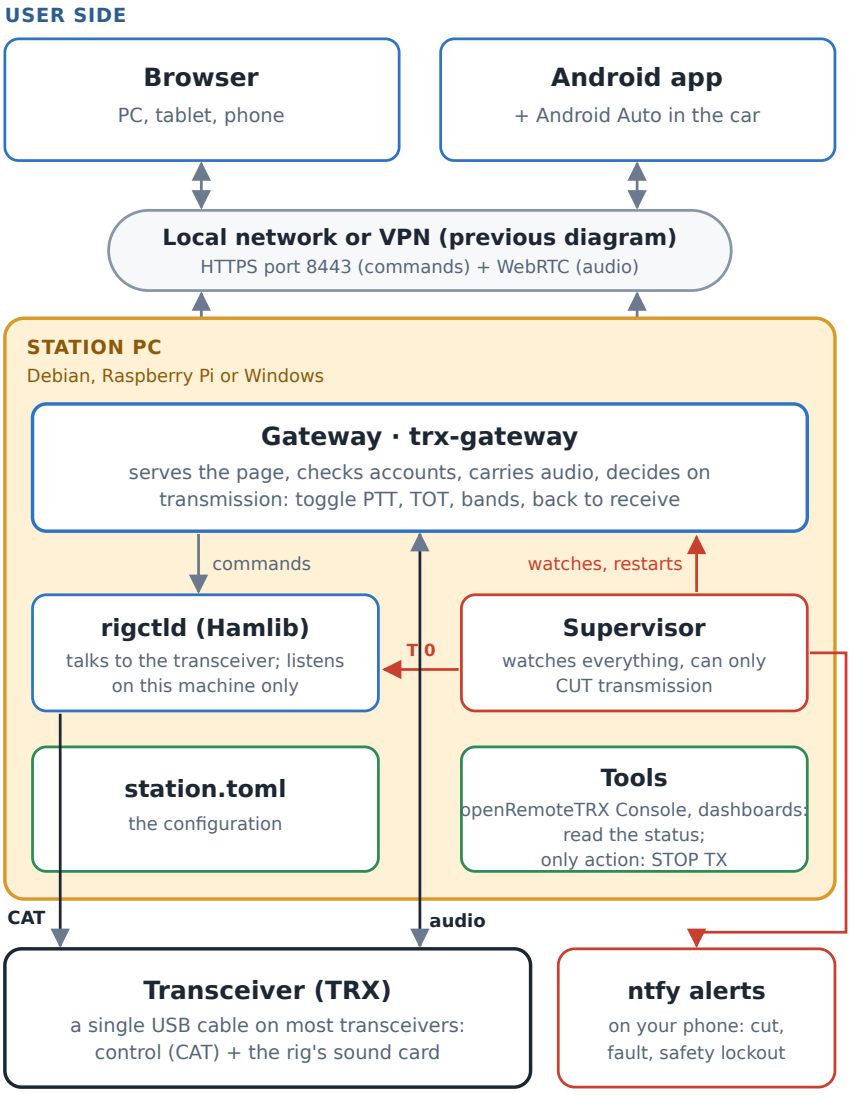
What you will get

- Your transceiver controlled from a **browser** (PC, tablet, phone) or the **Android app**: frequency, mode, filters, S-meter, received audio, and **voice transmission** with the phone's or the PC's microphone.
- A station that **protects itself**: it always starts in receive, cuts the transmission after a maximum duration (90 s by default), if the phone stops answering or if the link drops, and refuses to transmit outside the bands you have declared.
- An independent **supervisor** that watches the station and can send you alerts on your phone.

How it is organised



What is where: devices, network, station ·



The building blocks of an openRemoteTRX station ·

- You use a **browser** or the **Android app**. They talk to the **gateway**, the heart of the station, which runs on the **station PC** next to the transceiver.
- The gateway controls the transceiver through **rigctld** (the Hamlib program) and exchanges audio with the **transceiver's sound card**. It is the gateway that decides on transmission and applies the safety rules.
- The **supervisor** watches everything independently: it can only cut a transmission, restart a service, and send you **alerts**.
- **openRemoteTRX Console**, the dashboard and the indicator are your tools on the PC: they read the station's status.
- The configuration fits in a single file, `station.toml`.

Each word is explained in the glossary.

The hardware

Item	What you need
The transceiver	a transceiver known to Hamlib, its power supply, an antenna. Transceivers already tested have their sheet: Icom IC-7300MK2 (validated), Yaesu FTDX10 (validation in progress)
The PC - transceiver link	on most recent transceivers, **one USB cable** that carries both control (CAT) and audio; otherwise an external interface (Digirig, SignaLink…)
The station PC	PC running **Debian 12 or 13** with a desktop (recommended), **Raspberry Pi 4 or 5** running Raspberry Pi OS 64-bit, or a **Windows 10/11** 64-bit PC. It stays on all the time near the transceiver
The network	the PC connected to the internet box, preferably **by cable**
For the tests	a **dummy load** (50 Ω) that handles the test power: **mandatory** for the whole of part 3

Item	What you need
User side	an Android 10 or later phone, or any recent browser

Debian, Raspberry Pi or Windows?

	Debian (recommended)	Raspberry Pi	Windows
Hardware	a PC, even an old one (2 cores, 4 GB)	Pi 4 (**4 GB** advised) or Pi 5, official power supply, cooling, preferably an SSD	a 64-bit PC
System	Debian 12 or 13	Raspberry Pi OS 64-bit (Debian 12 or 13), with or without a desktop	Windows 10 or 11
Package	`trx-remote-debian-X.Y.Z.zip`	the **same** as Debian	`trx-remote-windows-X.Y.Z.zip`
Start	when the PC boots, without logging in	when the Pi boots, without logging in	when **you log in** (the PC must stay with your session open, locked if you wish)
Updates	over the Internet from openRemoteTRX Console, signature checked	as Debian	by hand, by running the installer of a new version
Tools	openRemoteTRX Console (menus 1 to 8)	the same, on the Pi's desktop or over SSH from another PC	openRemoteTRX Console too, through the single shortcut Start menu > openRemoteTRX (except updates, menus 7.x)
Status	validated (IC-7300MK2)	in service with a beta tester; load (SDR, FreeDV, several visitors) not measured yet	validated (IC-7300MK2)

The Raspberry Pi 3 and older models are too slow for live audio: the installation reports it.

Regulations

> **Safety and responsibility.** You remain responsible for every transmission of your station, **including remotely**: compliance with your licence, the allowed bands and powers, identification. The conditions for remote operation depend on the country: check with your administration (in France: the ANFR) before using it. Some band buttons of the page (85, 45, 11 m) are for **listening** only: they are not amateur radio bands.

Where to find the software

The installation packages are on the project's releases page:
https://github.com/Bob56fr/openRemoteTRX/releases

- Debian and Raspberry Pi: `trx-remote-debian-X.Y.Z.zip` (and its signature file `.sig`).
- Windows: `trx-remote-windows-X.Y.Z.zip`.

The Android app is in **invitation-only testing** (Google Play): ask for access by opening a ticket (part 8). Without it, the web page does everything from the phone's browser.

1. Prepare the transceiver

Station **switched off**, nothing connected: now is the time to set up the transceiver. The details of your transceiver's menus are in its sheet (Icom IC-7300MK2, Yaesu FTDX10); for another transceiver, see its manual and the “Depending on your brand” table of the quick start. This part tells you in which order to do things.

1.1 Install the USB driver (Windows only)

On Windows, the transceiver's USB driver (the one from its manufacturer: “Icom USB Driver”, Silicon Labs CP210x for Yaesu…) is installed **before** you plug in the USB cable; otherwise Windows may install the wrong driver. The name, the minimum version and the procedure are in the transceiver sheet, “USB driver”, or else on the manufacturer's website.

On Debian and on a Raspberry Pi, there is nothing to install: the kernel recognises the transceiver by itself.

1.2 Connect

1. Antenna disconnected, **dummy load** on the antenna socket in use.
2. USB cable between the transceiver and the PC.
3. Switch the transceiver on (long press on ON/OFF).

The PC now sees one or two **serial ports** and a **sound card**. The transceiver sheet tells you which one is used for control (IC-7300MK2: the first one, the second one is used for the HF spectrum; FTDX10: the “Enhanced COM Port”, the “Standard” one is not used).

Debian

Nothing to note: the wizard finds the right port by itself. To see it: `ls -l /dev/serial/by-id/` in a terminal (one stable name per port, which does not change from one USB socket to another).

Raspberry Pi

As on Debian. Plug the transceiver cable **directly** into a USB port of the Pi (the blue USB 3 ports work too); use a hub only if it has its own power supply: the Pi does not supply enough current to several devices.

Windows

Device Manager > Ports (COM & LPT): note the COM number of the control port.

1.3 Set the transceiver menus

Set the menus listed in your transceiver's sheet, section “Menus to set” (otherwise, its manual). They are essential:

- **CAT speed** identical to the station's (chosen in the wizard);
- **voice over USB**: otherwise the transceiver modulates with its own microphone, that is, the noise of the room. On an Icom (IC-7300MK2): MENU > SET > Connectors > MOD Input > **DATA MOD = USB** (the station only transmits in data mode); the transceiver accepts the USB input only in DATA MOD **or** DATA OFF MOD. With the second port set up, the station checks it at each connection and warns you if it is not on USB;
- **no RTS/DTR line** for PTT or CW: otherwise the transceiver may go into transmit when the PC starts;
- **VOX off**: a VOX that is on makes the transceiver transmit on a noise in the room, without any command. The station checks it each time an operator connects and **turns it off** if it is on (message in the page log, code ORT-5034). On some transceivers it stays on in one mode while it is off in another (seen on the IC-7300MK2: off in data mode, still on in voice);
- **built-in antenna tuner** (if the antenna goes through it): tuning must start by itself when transmission is keyed on a new frequency. On the IC-7300MK2: MENU > SET > Function > Tuner, **[TUNER] Switch = Auto** (tuner state stored per band) and **PTT Start = ON**. Answer “yes” to the wizard's tuner question: the station will check these settings at each connection and warn you if they are wrong (it never changes them). Outside the amateur bands, the transceiver does not guarantee tuning on PTT. During a transmission, if the SWR exceeds 2 for more than one second, the page shows a red banner and beeps (it does not cut);
- **transceiver transmit timer** (TX TIME OUT TIMER): a second safety, independent of the station. Set it (3 min recommended): if the station, the network or the PC fails during a transmission, this is what cuts. For a

transceiver whose reading is qualified (IC-7300MK2), the station reads it at each connection and warns you if it is OFF; for the others, it cannot check it.

> **Safety.** An extra safety costs nothing: also set a reasonable **maximum power** in the transceiver (HF MAX POWER). The station cannot exceed it, whatever happens.

1.4 PC sound

The transceiver's sound card must **never** be the PC's default sound device, otherwise desktop sounds go to the transceiver (and the desktop occupies the card):

Debian

Settings > *Sound*: the PC speakers as output, its microphone as input.

Raspberry Pi

Without a desktop (Raspberry Pi OS Lite): nothing to do, no program takes the transceiver's card. With a desktop: as on Debian, in the sound settings of the top bar. The Pi's own sound card (“Headphones”, “HDMI”) is not used by the station.

Windows

Settings > *System* > *Sound*: the PC speakers as output, its microphone as input; in “More sound settings”, the transceiver's card is neither “Default Device” nor “Default Communication Device”; sound scheme “No Sounds”. The details are in the Windows notice included in the package.

Transceiver sheet: Yaesu FTDX10

> **Prepared, not yet validated on a real transceiver.** Settings taken from the FTDX10 *CAT Operation Reference Manual* (Yaesu, edition 2308-F), from the USB driver instructions (2205-E) and from Hamlib's capabilities. The first operator who runs the tests of part 3 validates this sheet: thank you for reporting the result.

Identity for the station

Hamlib number	**1042** (“FTDX10”)
Link	USB, “**Enhanced COM Port**”, **38400** baud
PTT	by **CAT** (the station sends the transmit command through the USB port)
Sound card	“USB Audio CODEC” (receive and transmit)
Read by the station	frequency, mode, filters, S-meter, power, ALC, SWR, voltage, VFO A / B, split, clarifier
Not available	transceiver HF spectrum (Yaesu does not send the screen data over CAT: only the audio spectrum is shown); transceiver temperature (the station's temperature threshold has no effect)

USB driver (Windows)

1. **USB cable unplugged.** Download the “CP210x Universal Windows Driver” (link on the Yaesu website, or directly from Silicon Labs) and unzip it.
2. Right-click the `silabser.inf` file > (Windows 11: “Show more options”) > **Install**.
3. Plug in the USB cable, switch on the transceiver: Windows finishes the installation (sometimes asks for a restart). The sound card driver is the Windows one, nothing to do.
4. Device Manager > Ports (COM & LPT): two lines “Silicon Labs Dual CP210x USB to UART Bridge”: **Enhanced COM Port (COMx)** for the station, Standard COM Port (COMy) unused. A “!” or a cross on a line: uninstall then reinstall the driver.

On Debian: no driver. The stable port names are in `/dev/serial/by-id/` (`…CP2105…-if00-port0` = Enhanced, `…-if01-port0` = Standard); the wizard offers the right one by default.

Menus to set

Paths as they appear in the transceiver's settings menu (see the FTDX10 operating manual to open it). The “MODE” settings are made **for each mode** you want to use in phone.

CAT link: OPERATION SETTING > GENERAL

Menu	Value	Why
CAT RATE	**38400 bps**	same speed as in the station (offered by the wizard)
CAT TIME OUT TIMER	**1000 msec**	value offered by the wizard
CAT RTS	**ENABLE**	Hamlib uses RTS/CTS flow control on this transceiver
TX TIME OUT TIMER	**3** (minutes)	second safety of the transceiver itself, beyond the station's maximum time (90 s)

(`232C RATE` and `232C TIME OUT TIMER` concern the RS-232C socket: no effect over USB.)

Voice through USB: RADIO SETTING > MODE SSB, MODE AM, MODE FM, MODE PSK/DATA

Menu (in each mode)	Value	Why
SSB / AM / FM / DATA **MOD SOURCE**	**REAR**	the modulation comes from the rear of the transceiver, not from the front microphone
REAR SELECT	**USB**	the rear = USB (and not the DATA socket)

Menu (in each mode)	Value	Why
RPTT SELECT	**DAKY**	the “rear” PTT stays on the DATA socket, where nothing is plugged in; **never RTS or DTR**
RPORT GAIN	to adjust (part 3)	level of the voice arriving through USB: to set with the ALC
SSB / AM / FM / DATA OUT LEVEL	factory value	level of the rear RTTY/DATA socket only: **no effect on the USB audio** (test of 2026-11-19). Received audio too low: gain of menu **2.9** of openRemoteTRX Console

Without “MOD SOURCE = REAR” in SSB, the wizard limits transmission to data modes (DATA-U, DATA-L): the page will not be able to transmit in USB / LSB.

> **Safety: RPTT SELECT and PC KEYING.** Yaesu warns that, when the PTT goes through the USB port lines, *“the transceiver may go into transmission when the computer starts”*. The station does not use these lines: leave **RPTT SELECT = DAKY** in each mode and **CW SETTING > MODE CW > PC KEYING = OFF**.

Other recommended settings

Menu	Value	Why
Front panel **VOX** key	**off**	with the voice through USB, a stray sound from the PC could make the transceiver transmit
OPERATION SETTING > TX GENERAL > HF MAX POWER (and 50M, 70M, AM)	the power you allow yourself	ceiling of the transceiver itself, which the station cannot exceed
OPERATION SETTING > TX GENERAL > EMERGENCY FREQ TX	**DISABLE**	no transmission on the emergency frequency

Mode mapping

Web page / Hamlib	FTDX10 screen
USB / LSB	USB / LSB
PKTUSB / PKTLSB	DATA-U / DATA-L
FM / PKTFM	FM / DATA-FM
AM	AM
CW / CWR	CW-U / CW-L

Points to check during the tests (validation of this sheet)

To note in the report of part 3:

1. The diagnostic (menu 6.1) finds the transceiver; frequency and mode follow when you turn the VFO.
2. **VFO A / VFO B** indicator in the page when you press A/B on the transceiver.
3. Transmission on a dummy load: the page sees power, **ALC** and SWR; the ALC level at full voice (the ALC alert threshold has not yet been measured on a Yaesu).
4. Split enabled on the transceiver: the page shows it, the transmission goes out on the right frequency.
5. Transmission with the transceiver microphone: the page shows “local operator” and blocks the remote PTT.
6. No transmission when the PC starts or when the cable is plugged in.

Transceiver sheet: Icom IC-7300MK2

> **Validated on a real transceiver** (public status “validated”, ★ GO and level 5 “complete” on the transceivers page, 2026-11-19): control, transmission, audio, web page, HF spectrum through the second port and failure tests, on Windows and on Debian 13. What remains to be tested is listed at the end of the sheet. The IC-7300 (first model) is close, but has not been tested.

Identity for the station

Hamlib number	**3094** (“IC-7300MK2”); requires Hamlib **4.7 or newer** (Debian 13: installed by `install.sh`)
Link	**a single USB cable**: control (CI-V), audio and second port
Serial ports	two ports: the **first** for control (Debian: `…-if00` in `/dev/serial/by-id/`; Windows: one COM), the **second** for the HF spectrum (Debian: `…-if02`; Windows: another COM)
PTT	by **CAT** (CI-V)
Sound card	“**USB Audio Device**” (the transceiver's C-Media chip; on Windows its name starts with a digit that changes with the USB socket)
Read by the station	frequency, mode, filters, S-meter, power, ALC, SWR, voltage, split, RIT / ΔTX, VOX, tuner settings, **transceiver TOT**
Not available	active VFO (A, B, memory); transceiver temperature (the station's temperature threshold has no effect)

USB driver (Windows)

Install the **“Icom USB Driver” version 1.80 or newer** from the Icom website, cable unplugged, then plug in the transceiver. Version 1.70 freezes Chrome and Edge when a page reads the serial port (Autotest). On Debian: no driver.

Menus to set

Link: MENU > SET > Connectors

Menu	Value	Why
CI-V > CI-V Baud Rate	the **same speed** as in the wizard	otherwise the transceiver does not respond
MOD Input > **DATA MOD**	**USB**	the voice arrives through USB; the station transmits only in **data mode** (USB-D / LSB-D)
USB SEND, USB Keying (CW), USB Keying (RTTY)	**OFF**	no line of the USB port must be able to put the transceiver into transmission
USB MOD Level	to adjust (part 3)	level of the voice arriving through USB: to set with the ALC

The transceiver accepts USB only in DATA MOD **or** DATA OFF MOD. With the second port set (menu 2.7), the station checks DATA MOD at each connection and warns you if it is not on USB.

Safety: MENU > SET > Function

Menu	Value	Why
Time-Out Timer (CI-V)	**3 min**	last barrier, in the transceiver itself; the station reads it again at each operator connection and warns you if it is OFF
Tuner > [TUNER] Switch	**Auto**	built-in tuner: state stored per band
Tuner > PTT Start	**ON**	tuning started by itself at the first transmission on a new frequency

The station checks the tuner settings at each connection (if you answered “yes” to its question in the wizard), without ever changing them.

Other recommended settings

Setting	Value	Why
VOX key	off	the station switches it off at each operator connection; on this transceiver, the VOX state depends on the mode (off in data mode, still on in phone)
Power (RF POWER, MULTI knob)	the power you allow yourself	ceiling of the transceiver itself
Split	with the transceiver's **SPLIT key**	the station never writes it: it reads it and refuses to transmit if the transmit frequency leaves your bands

Audio on Windows

Settings > Sound: the transceiver's “USB Audio Device” output **at 100 %**, enhancements disabled; the transmit level is set with **USB MOD Level** in the transceiver. An output at reduced volume gives a transmission without power. The transceiver's card must never be the PC's default sound (part 1).

HF spectrum through the second port

Menu **2.7** of openRemoteTRX Console: choose the transceiver's **second** port. The station reads the transceiver's screen there (HF spectrum of the page and of the application) and the FIL1/2/3 filter presets. The spectrum **freezes during transmission**: the transceiver stops sending it, it is not a fault.

Mode mapping

Web page / Hamlib	IC-7300MK2 screen
USB / LSB	USB / LSB (receive; transmission refused: the voice goes through the data modes)
PKTUSB / PKTLSB	USB-D / LSB-D
FM / PKTFM	FM / FM-D
AM / CW / RTTY	AM / CW / RTTY

Points still to test

- Remote power on and off (menu 2.8): a switched-off IC-7300MK2 refuses the reading of its power state; to test on your transceiver before relying on it.
- Transmission with the transceiver microphone while the station is running (test 6 of part 3.5).

2. Install the station

Installation never controls the transceiver: it can stay on and connected. Choose your system in the tabs below; the choice applies to the whole page. At the end, **openRemoteTRX Console** opens and offers the configuration wizard: that is part 3.

Debian

Debian 12 or 13, with a desktop.

1. Download `trx-remote-debian-X.Y.Z.zip` (part 0) into **Downloads**, right-click > “Extract Here”.
2. In the resulting folder: right-click in an empty area > “Open in Terminal”, then:

   ```
   sudo bash install.sh
   ```

   Your session password is requested (nothing is displayed while you type: this is normal). The script installs Hamlib, Python, the modules, the service account and the station services.
3. At the end, **log out and log back in** (your account's permissions change).
4. Open **openRemoteTRX Console** from the applications menu (or type `trx-console` in a terminal).

Hamlib and recent transceivers. Debian 13 ships Hamlib 4.6, which does not know recent transceivers (IC-7300MK2…): `install.sh` installs version 4.7 from Debian's **official backports** by itself. On Debian 12, if your transceiver is missing from `rigctl -l`, the installation tells you: then run `sudo bash compiler-hamlib.sh` in the same folder (compiles the latest Hamlib, 10 to 30 minutes).

Without a desktop (server), everything is done over SSH: copy the zip, then

```
unzip -q trx-remote-debian-*.zip && cd trx-remote-debian-*/ && sudo bash install.sh
```

close the SSH connection, reconnect and type `trx-console`.

Raspberry Pi

Raspberry Pi 4 or 5, Raspberry Pi OS 64-bit. The Pi uses the **same package** as Debian.

> **The Pi as a station.** Already in service with a beta tester, but its load has not yet been measured with everything running (SDR receiver, FreeDV, several visitors): start with the basic functions. A Pi 3 or older is too slow for live audio.

The hardware.

Item	Advice
The Pi	Pi 4 with **4 GB** (2 GB minimum), or Pi 5
Power supply	the **official** unit (Pi 4: USB-C 5.1 V 3 A; Pi 5: 5 V 5 A). A phone charger causes undervoltage: the Pi slows down and the transceiver's USB drops out
Cooling	heatsink with fan, or ventilated case: without it, the Pi slows down around 80 °C
Storage	32 GB “High Endurance” class A2 SD card, or better an **SSD over USB 3** (the Pi boots from it, and it lasts longer than a card that writes day and night)
Network	Ethernet cable recommended; Wi-Fi works, with more audio dropouts

1. Prepare the card, on your PC, with **Raspberry Pi Imager** (raspberrypi.com/software):

1. Device: your Pi model. Operating system: **Raspberry Pi OS (64-bit)** if a screen will stay connected; otherwise *Raspberry Pi OS (other)* > **Raspberry Pi OS Lite (64-bit)**, without a desktop, everything will be done from your PC.
2. Customisation, **before** writing the card: a name (for example `trx-pi`), an account and its **password** (required for installing), the time zone and keyboard, Wi-Fi only if there is no cable, and **SSH enabled** (preferably with a public key).
3. Write the card, put it in the Pi, connect the network, the transceiver, then the power supply. The first boot takes 2 to 3 minutes (the Pi restarts once by itself).
4. Find the Pi's address in your router's list of devices, and **reserve it** (DHCP reservation, part 4.1).

2. Update the system, in a terminal on the Pi (with a desktop) or from your PC (without a desktop: `ssh -l <account> <pi-address>`):

```
sudo apt update && sudo apt full-upgrade -y && sudo reboot
```

3. Install the station.

- **With a desktop**: exactly as on Debian (Debian tab): zip downloaded and extracted, `sudo bash install.sh` in a terminal opened in the folder, then log out and log back in.
- **Without a desktop**, from your PC (PowerShell on Windows, terminal on Linux or macOS), in the folder of the downloaded zip:

  ```
  scp -o User=<account> trx-remote-debian-X.Y.Z.zip <pi-address>:
  ssh -t -l <account> <pi-address>
  ```

  then, on the Pi:

  ```
  sudo apt install -y unzip
  unzip -q trx-remote-debian-X.Y.Z.zip && cd trx-remote-debian-X.Y.Z && sudo bash install.sh
  ```

  At the end, type `exit` and reconnect (`ssh -t -l <account> <pi-address>`): your account's permissions have changed. Then `trx-console`.

What the installation does in addition on a Pi.

- It shows the model and reports a Pi that is too old, too little memory or **undervoltage** (power supply too weak): fix it before going further.
- Raspberry Pi OS 13 ships Hamlib 4.6: it installs 4.7 from Debian's official backports (recent transceivers such as the IC-7300MK2). On Raspberry Pi OS 12, if your transceiver is missing from `rigctl -l`: `sudo bash compiler-hamlib.sh` (10 to 30 minutes on a Pi).
- It enables the Pi's **hardware watchdog**: if the system freezes for more than 15 seconds, the Pi restarts by itself, and the station comes back in receive.
- With a desktop, it adds the font for openRemoteTRX Console's icons.

4. Check the power supply and the temperature, after a few minutes of running:

```
vcgencmd get_throttled; vcgencmd measure_temp
```

Expected: `throttled=0x0` and less than 70 °C. Any other value of `throttled` = power supply too weak or overheating: change the power supply or add a fan. Once the station is running, its supervisor will send you an alert (part 6) if the Pi is slowed down, or if the memory or the disk fill up.

▼ **Working without a screen day to day**

- **openRemoteTRX Console over SSH**: `ssh -t -l <account> <pi-address> trx-console` (the `-t` is essential: the console asks questions). Windows Terminal displays the frames, colours and the QR code of menu 4.1 correctly.
- **More comfortable: the console in the browser.** Once, menu **4.6** (part 7.5); then `https://<pi-address>:8444/` from the PC, the tablet or the phone, without SSH.
- **Dashboard**: `ssh -t -l <account> <pi-address> trx-monitor`, or the vital signs dashboard in the browser (part 7.1).
- **Shut down the Pi**: `sudo poweroff`, then wait until the green LED stops blinking before cutting the power (a sudden power cut can damage the SD card).

Windows

Windows 10 or 11, 64-bit.

1. Install **Hamlib 64-bit** (`hamlib-w64-X.Y.Z.exe`, version **4.7** or later: recent transceivers such as the IC-7300MK2 do not exist before) from https://github.com/Hamlib/Hamlib/releases, in the suggested folder.
2. Copy `trx-remote-windows-X.Y.Z.zip` to the PC. Right-click > Properties > tick **Unblock** > OK.
3. Right-click > Extract All, then double-click **`installer.cmd`**. If Windows shows “Windows protected your PC”: More info > Run anyway.
4. Accept the administrator request and answer the questions (Python is installed if it is missing, with your consent).
5. Essential Windows settings:

- home network set to the **Private** profile (the firewall only opens the station on private networks);
- **sleep and hibernation disabled** (Settings > System > Power);
- *Settings* > *Privacy* > *Microphone*: access allowed for desktop apps (otherwise the audio received from the transceiver is silence);
- transceiver sound card (*Settings* > *System* > *Sound*, the transceiver's "Speakers" output): **volume at 100**, **audio enhancements disabled**, format 16 bit 48000 Hz; the transmit level is set on the transceiver (Icom: USB MOD Level). A reduced volume (38 seen on one PC) gives a transmission with no power. Its name ("2- USB Audio Device"…) changes if the transceiver cable moves to another USB socket: then redo menu 2.6.

The full notice (`notice-windows.html`) is in the package.

openRemoteTRX Console on Windows. The installation creates **a single shortcut**: Start menu > **openRemoteTRX**. It opens openRemoteTRX Console, with the same numbered menus as on Debian: this guide applies as is (1.1 start, 2.1 configuration, 6.1 diagnostic…). Differences:

- **Updates** (menus 7.1 to 7.3): on Windows, extract the new zip and run its `installer.cmd`; the configuration is kept. Going back: run `installer.cmd` of the old version again.
- **Indicator next to the clock**: menu 8.4 (start it, or start it at each session login).
- **Uninstall**: Settings > Apps > **openRemoteTRX** > Uninstall (the configuration is kept unless you ask to erase it).
- The configuration is in `C:\ProgramData\trx-remote\station.toml`; the commands (`trx-console`, `trx-setup`, `trx-monitor`…) can also be run in a terminal opened **after** the installation.

3. Integrate the TRX and test it

> **Safety.** Throughout this part: **dummy load** on the antenna socket, **reduced power** (10 W for example), nobody near the disconnected antenna.

3.1 The configuration wizard (menu 2.1)

The transceiver is only queried **read-only**: no transmission during the wizard. Nothing is saved if you give up halfway. The values specific to your transceiver are in its sheet:

Question	Answer
Transceiver model	type its name (`ic-7300`, `ftdx10`...) and choose from the list (Hamlib number in the sheet: IC-7300MK2 = 3094, FTDX10 = 1042). For some transceivers, the wizard reminds you of the menus to check (part 1)
CAT serial port	the **control** port (Debian and Raspberry Pi: offered automatically; Windows: the COM noted in part 1)
Speed	**identical** to the CAT speed set in the transceiver
PTT method	**CAT** (offered)
Does the voice go over USB?	**yes**, if the transceiver modulates over USB in voice (part 1); otherwise transmission will be limited to data modes
Sound cards (receive, transmit)	the **transceiver's** for both (its name is in the sheet)
HF spectrum through a second port	the transceiver's **second port** if it has one (Icom IC-7300MK2…); otherwise **none**
Bands allowed for transmission	the bands of **your licence**, only. Empty list = receive only
Network access	"**Local network or VPN, over HTTPS with a certificate generated here**" (recommended)
Accounts	a username (letters, digits, `_`, `-`) and a **long password** (Enter = a strong password is generated: write it down); Enter on an empty username to finish
Maximum transceiver power	your transceiver's (often **100** watts)
Maximum duration of a transmission (TOT)	**90** seconds (default value, ceiling 120)
Replay of recordings	**no** (default answer)

At the end, the wizard shows the address to open (`https://<station-address>:8443`) and the **certificate fingerprint** (useful for the Android app, part 4).

3.2 The diagnostic (menu 6.1)

It reviews everything without changing anything: each line is **OK**, to **watch** or **failed**, with what to do. Fix each failure before going further (menu 8.1 for a setting, or run the wizard again).

Common failures at this stage: wrong port (the transceiver's second port instead of the control one; Yaesu: Standard instead of Enhanced), transceiver speed different from the one in the wizard, transceiver sound card taken by the desktop (part 1.4); on a Raspberry Pi, a power supply that is too weak ("undervoltage", part 2).

3.3 The guided transceiver test (menu 2.2)

The station is stopped during the test.

1. **Choice 1, receive only**: CAT link, received audio, frequency and mode change, filter widths. Check that the received audio is clean and at a correct level (otherwise: transceiver setting given by its sheet, then receive audio gain in menu 2.9).
2. **Choice 2, receive then transmit**: dummy load, reduced power. Each switch to transmit asks you to type `TRANSMIT` (or `EMETTRE`).

Copy the **final summary** shown on screen: it is used for the validation report.

3.4 Start the station and set the modulation

1. Menu **1.1**: the station starts (the transceiver is always put back in receive at startup). Run a diagnostic again (menu 6.1) a few seconds later.
2. Open the page `https://<station-address>:8443` from a phone or a PC on the network (details and certificate warning: part 4), and log in.
3. Allowed band, dummy load, voice mode that your transceiver modulates over USB (USB or LSB; on an Icom, USB-D or LSB-D, part 1): press **PTT** (toggle: one press to transmit, one press to go back to receive), speak normally for a few seconds, press again.
4. Watch the **ALC** in the page: it must move on voice peaks without staying at maximum. Adjust the transceiver's USB input level until you get this result (Icom: **USB MOD Level**; Yaesu: **RPORT GAIN** in the menu of the mode in use; the exact name is in the transceiver sheet).

3.5 The safety tests

Do **all** of them, on a dummy load, before any transmission on an antenna. Each one must end with the transceiver back in receive (check on the transceiver's screen, not only in the page).

#	Test	Expected result
1	During a transmission, the page's **STOP** button	immediate return to receive
2	Transmit without touching anything	countdown and beep 10 s before the end, return to receive after **90 s**
3	During a transmission, turn off the phone's Wi-Fi (or close the tab)	return to receive in about **2 s**
4	During a transmission from the phone, lock the screen	return to receive
5	Set a frequency outside your allowed bands, press PTT	transmission **refused**, message in the page
6	Transmit **with the transceiver's microphone** (without the station)	the page shows the local operator and blocks remote PTT during the transmission and for 2 minutes after
7	During a transmission, the **STOP TX** key of the dashboard (menu 1.3)	return to receive
8	During a transmission, stop the station (menu 1.2)	return to receive
9	Restart the PC (or the Pi), transceiver on	the transceiver **never** goes into transmit during startup

A test that does not give the expected result: **do not use the station to transmit**, stop it (menu 1.2) and ask for help (part 8) with the report from menu 6.3.

3.6 Going further (optional)

- **Transceiver qualification** (menu 8.3, `trx-qualif`): for an expert who wants to check everything Hamlib declares for your transceiver. Start with choice 2 (exploration without transmission); the report it produces is valuable for validating the transceiver sheet. Choice 6 (“qualification case”) runs everything in sequence for a team member who came to qualify the transceiver: reading of the transceiver before and after (what must be restored by hand), exploration, qualification, the timed safety tests of § 3.5, and a zip report to hand over to the developers.
- **Transceiver profile** (menu 2.3): saves the configuration under a name (“IC-7300 home”) to load it again later if you change transceivers.

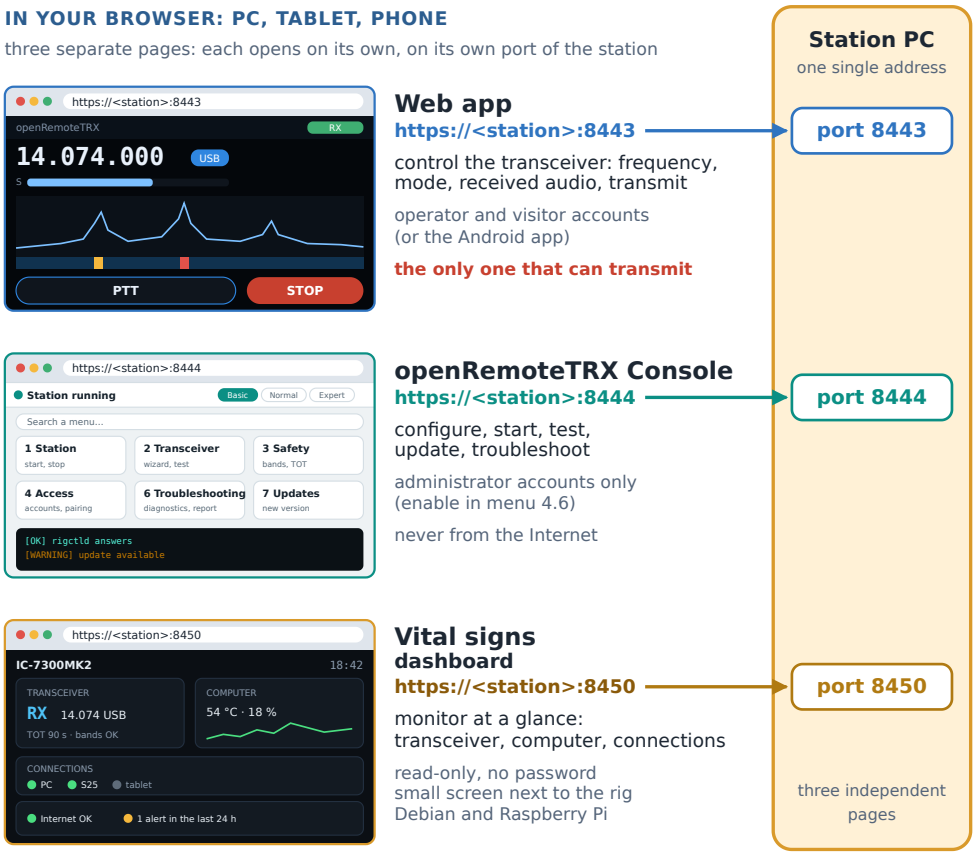
3.7 Report the validation

If your transceiver is not yet validated (list: https://bob56fr.github.io/openRemoteTRX/postes/), your result counts: open a “Report a bug” issue (even if everything is fine, title “<your transceiver>: commissioning tests”) with the summary of the guided test, the result of the 9 safety tests and, if it has a sheet, the points it says to check.

4. Using the station on the local network

The local network is your home Wi-Fi or cable: the phone and the station sit behind the same internet box.

Three station pages open in the browser, each one separately, on its own port of the same address:



Three station pages in the browser ·

4.1 A fixed address for the station

In your box, **reserve an IP address** for the station PC (DHCP reservation). The certificate created by the wizard carries the PC's address: if it changes, the browser and the app will no longer recognise it. After an address change, run the wizard again (menu 2.1) to recreate the certificate.

The address to use is shown at the top of openRemoteTRX Console and by the diagnostic.

4.2 The web page

1. On the phone or a PC: `https://<station-address>:8443`
2. The browser reports an **unrecognised certificate**: this is expected (certificate created by your station, not by an authority). “Advanced”, then “Proceed”. This is done only once per device.
3. Log in with the account created by the wizard (Enter in the password field is enough).

> **Password remembered by the browser.** Chrome **never** saves a password on a page whose certificate is not recognised, even after “Proceed”. The solution: install once, on each device, your **station's authority** (below). After that, no more warning and the password is remembered.

Install the station authority (once per device)

1. On the station, openRemoteTRX Console, menu **4.4** “Station certificate”: choice **1** (recreate the certificate, signed by the station authority), then restart the station when the console offers it. The certificate fingerprint changes: **pair the Android app again** (menu 4.1).
2. On the device, open the station page: under the password, tap **“Install the station certificate”** (or open `https://<station-address>:8443/station-ca.crt`).
3. **Windows**: open the downloaded file, “Install Certificate”, “Current User”, “Place all certificates in the following store”, **“Trusted Root Certification Authorities”**, Finish, then accept the Windows warning.

Close **all** Chrome windows and open it again. Careful: with “Automatically select the certificate store”, Windows puts it in “Personal”, where Chrome ignores it (address still struck through in red): install it again and choose the right store.

4. **Android**: Settings, Security (or “Security and privacy”), More security settings, Install from device storage (or “Encryption and credentials”), **“CA certificate”**, choose the downloaded file and confirm. Open Chrome again.
5. Open the page: no more warning; log in, Chrome offers to save the password.

> This authority can vouch **only** for private addresses (local network, VPN) and your station's names: even if its key were stolen, it could not impersonate another site (bank…) on your devices.

What you should see: frequency and mode following the transceiver, S-meter, received audio (tap the sound button if the browser blocked it).

To **transmit**, the page needs the **microphone**: the browser asks for it at the first transmission. It only offers it over HTTPS, hence the certificate.

> **Safety.** The PTT is a **toggle**: one press to transmit, one press to go back to receive. The state shown is the one the station confirms, never a guess. **STOP** cuts at any time. **STOP held for 3 seconds** (large countdown in the middle of the screen, releasing cancels) opens a window: “Disconnect”, or “Power off the transceiver remotely” (with a confirmation; only if remote power-on is in service, menu 2.8). A hidden page (other tab, locked screen) stops the transmission.
>
> **Transceiver off.** A red “RIG OFF” banner says so. If remote power-on is in service, your connection powers the transceiver on by itself: a window shows the steps (serial ports checked, command, start-up, read-back) and, if it fails, a clear message with “Retry” and “Continue without the rig”; the “Power on the rig” button of the banner does the same. An automatic reconnection never powers the transceiver on. The app does the same (its STOP cuts as soon as it is pressed); in the car, Android Auto shows “RIG OFF” or “POWERING ON THE RIG…” as the title.

4.3 Several devices, several users

Simple rule: **one account per device**, named after the device (`PC`, `S25`, `tablet`…), even if they are all yours. The same account opened on a second device **closes the connection of the first** (and cuts its transmission if it was transmitting). With one account per device, the station knows which one is transmitting, and you can remove a single device (lost phone) without touching the others.

Everything is done in menu **4.3 “Operator accounts”** of openRemoteTRX Console:

- **add** an account: username, password (Enter = a strong password is generated, write it down), then the **pairing QR code** of this account for the app;
- **change a password** (a forgotten password is replaced: the station does not know it in clear);
- **disable** an account (lost or lent phone), then enable it again;
- **delete** an account.

The station applies the changes within a few seconds, without restarting; the connection of an account that is disabled, deleted or whose password changed is closed at once. The last active operator account can be neither disabled nor deleted. Only one person can transmit at a time. For listeners without any control, prefer **visitors** (menu 4.2, part 9).

4.4 The Android app

The app (“openRemoteTRX”, Android 10 or later) brings what the page cannot do: transmission that continues with the screen off, headset button, Android Auto, driving mode.

1. Install the app (invitation-only testing, part 0).
2. On the station, menu **4.1**: openRemoteTRX Console shows the address, the accounts, the certificate fingerprint and a **QR code** (without password).
3. In the app: “**Scan the station's QR code**”. The address, username and fingerprint are filled in; enter the password.
4. Log in: same check as for the page (frequency, received audio), then a transmission test on a dummy load with STOP.

The certificate fingerprint protects the link: the app refuses a station whose certificate has changed. After a new certificate (address change), do the pairing again.

> **Safety.** In the car, the phone's touch PTT is hidden (driving mode): use Android Auto, voice commands or the steering-wheel buttons, and obey the highway code.

5. Using the station remotely (outside the local network)

The station does not manage the network: it listens on an address and a port, and protects access with a username and password. You choose how your phone reaches it from outside. The full technical guide: https://github.com/Bob56fr/openRemoteTRX/blob/main/docs/acces-distant.md

5.1 Choose: VPN or open port

	A. VPN (recommended)	B. Open port on the box
Principle	a VPN links the phone to the station's network	the box forwards an Internet port to the station
Examples	WireGuard (often built into the box), Tailscale, OpenVPN	box + domain name + recognised certificate
Visible from the Internet	nothing	the login page
Audio (WebRTC, UDP)	goes through easily	often goes through, **not always** (strict box, address shared by the provider)

Choose the VPN if in doubt: nothing is exposed, and the audio goes through.

5.2 Option A: VPN

1. Install the VPN of your choice: server on the box or on the station PC, client on the phone. For Tailscale, a dedicated guide: https://github.com/Bob56fr/openRemoteTRX/blob/main/docs/acces-distant-tailscale.md
2. **One VPN profile per device**, never shared. Two devices on the same profile steal the link from each other: dropouts every 15 to 30 seconds.
3. If the VPN lands **on your local network** (VPN of the box): nothing to change on the station, use the same address as at home. If the VPN gives the station PC **a separate address** (Tailscale, WireGuard on the PC): run the wizard again (menu 2.1, network access) so that the certificate also covers this address, then pair the app again (menu 4.1).
4. On the phone, on 4G, VPN active: open the same page `https://<station-address>:8443`.

On Android, only one VPN can be active at a time. With **Android Auto**, route only the station's network through the tunnel (not all traffic), otherwise the car refuses the link: detailed settings in the technical guide, section “Android Auto et VPN”.

5.3 Option B: open port on the box

For those who know their box and certificates.

1. **Check that it is possible**: compare the “WAN” address of your box with the one given by a “what is my IP address” website. If they differ, your provider shares your address (CGNAT): no port can be opened, use a VPN.
2. A **domain name** pointing to your box (dynamic DNS if the address changes).
3. A **recognised certificate**: the simplest is an HTTPS proxy (Caddy) on the station PC, which obtains and renews a Let's Encrypt certificate by itself; the station then keeps listening locally. Example and settings in the technical guide, section “Option B”.
4. On the box, forward port 443 (TCP) to the station PC.

> **Safety.** With an open port, the **password is your only barrier**: long and unique for each account. The station slows down attempts (5 failures per minute and per origin, 30 per quarter of an hour and per account), but a stolen password allows transmitting in your allowed bands. The transmission safeguards stay active in all cases, and so does the transceiver's transmit timer (part 1).

5.4 Tests from outside

Phone **on 4G** (Wi-Fi off), dummy load on the transceiver:

1. The page opens (option B: padlock without warning).
2. A wrong password is refused.
3. The right password connects: frequency, S-meter, **received audio**.

4. Transmission on the dummy load: PTT, then **STOP**; then transmit and switch to airplane mode: back to receive.
5. If the page opens but the audio does not establish: UDP is not getting through (option B): switch to the VPN.

In case of repeated dropouts, menu 6.4 (network quality) says which segment fails: phone, mobile network, tunnel, box or station.

6. An unattended station

A remotely controlled station runs most of the time with nobody next to it. This part prepares for that.

6.1 Alerts on your phone (ntfy)

The station's supervisor can send you notifications: transmission cut for safety, failure, safety lockout, memory or disk almost full, link dropout, and a silent “Station in service” message every hour (if it stops arriving, the station or its connection has failed).

1. Install the **ntfy** app on the phone.
2. Choose a **long topic name that is impossible to guess** (for example 20 random letters and digits): on the public ntfy.sh server, anyone who knows the name reads the messages.
3. Subscribe to this topic in the ntfy app.
4. On the station, menu **3.5** “Alerts on the phone”: choice 1, paste the address `https://ntfy.sh/<your-topic>`, save, accept the restart. The screen never shows the full address.
5. Check: choice 4, “Send a test alert now”; the phone must ring at once (code ORT-8025). After that, the “Station in service” message arrives every hour (choice 3 to change this rhythm).

Each alert carries an “ORT-NNNN” code with a link to its explanation (part 8). Two alerts arrive at restart after a shutdown that was not clean: **ORT-8030** (Debian, Raspberry Pi: the machine stopped abruptly, power cut or crash) and **ORT-8031** (the supervisor alone stopped); each gives the time of the last sign of life. A normal stop (menu 1.2, PC shutdown) sends nothing.

6.2 The safety lockout

After repeated failures that restarts did not fix, the supervisor **puts the station in safety lockout**: transceiver in receive, access cut, alert sent. The safety lockout **survives restarts**: find the cause (menus 6.1 and 6.2), then lift it with menu **1.4**.

6.3 After a power cut

Item	To set
PC	in the BIOS / UEFI: “Restore on AC power loss” = **Power On** (the name varies with the motherboard)
Raspberry Pi	nothing: it restarts as soon as power returns
Debian or Raspberry Pi station	nothing: it starts with the machine, without opening a session
Windows station	it starts at **session login**: without automatic login of your session, it stays stopped
Transceiver	most transceivers do **not** power on by themselves. The station can power it on through CAT (menu **2.8**, off by default, with a power off / power on test): your connection from the page or the app powers it on
Box	powers on by itself; check that the VPN does too

A **UPS** for the PC, the box and the transceiver's power supply avoids most of these cases.

6.4 Hardware protection

- **Lightning**: a station away from any human presence is exposed. Lightning arrester on the antenna cable, and antenna grounded or disconnected when a storm is forecast.
- **Power**: the transceiver's ceiling (HF MAX POWER, part 1) remains the last barrier.
- **Transceiver timer** (TX TIME OUT TIMER): set to 3 min, it cuts even if everything else has failed.
- **Temperature**: the supervisor watches the PC's processor and alerts when it stays above 70 °C for one minute (a peak of a few seconds does not count); the alert says which programs were heating. On Debian and on a Pi, the history of the last 24 h (one reading every 10 s) is attached to the troubleshooting report (menu 6.3). On a Pi, a fan is essential. The transceiver's temperature cannot be read by the station on the transceivers tested (IC-7300MK2, FTDX10): proper ventilation around the transceiver.
- **Local operator**: if someone uses the transceiver on site, remote transmission is blocked during their transmission and then for 2 minutes; the station never cuts them for this reason. A transmission with the microphone is cut by the absolute TOT (135 s), unless the operator touched the transceiver (volume, frequency, mode, leaving DATA mode…) in the 2 minutes before speaking or while speaking: the station

knows they are present and lets them speak for up to 5 minutes (`local_tx_max_s`), with a simple alert in between. The transceiver timer, if set shorter, cuts first. These settings are in menu **3.4** “Operator on site”: keep “shared station”; otherwise, a microphone held down leads to the safety lockout.

7. Day-to-day operation

7.1 Start, stop, monitor

openRemoteTRX Console (on Windows: Start menu > **openRemoteTRX**):

Need	Menu
Start / restart	1.1
Stop (before unplugging the transceiver, or for another program)	1.2
Live status, STOP TX key	1.3 (`trx-monitor`)
Permanent station indicator	8.4: indicator in the top bar (Debian), next to the clock (Windows)
Log	6.2

The station always returns to receive at start and stop. Stop it before unplugging the transceiver's USB cable.

Vital signs dashboard (Debian and Raspberry Pi): a read-only page, made for a small screen next to the transceiver or for a quick look from the phone: transceiver, transmission, connections (one dot per operator account), machine health (temperature, load, memory), Internet, recent alerts, 24 h history. On the station: `http://localhost:8450/`; from the local network or the VPN: the address on the “Web dashboard” line at the top of openRemoteTRX Console (`https://<station-address>:8450/`). No command is possible there: no password, no access from the Internet.

7.2 Update

- **Debian and Raspberry Pi**: when a “NEW VERSION … AVAILABLE” frame appears at the top of openRemoteTRX Console, menu **7.1**: it shows what is new in each version, checks the signature, installs the new version next to the old one and keeps your configuration. If there is a problem: menu **7.2** (back to the previous version, without Internet). Station without Internet: drop the zip on the Desktop and use menu **7.3**.
- **Windows**: download the new zip, run its `installer.cmd`; configuration and profiles are kept. Keep the zip of the old version to go back.

The update is refused during a transmission.

7.3 Back up

Copy these files elsewhere (USB stick, another PC) after each configuration change:

	Debian, Raspberry Pi	Windows
Configuration (accounts, bands, safeguards)	`/etc/trx-remote/station.toml`	`C:\ProgramData\trx-remote\station.toml`
Certificate and its key	`tls` folder next to `station.toml`	same
Transceiver profiles, Breizh key	`/etc/trx-remote/profils/`, `breizh.key`	in `C:\ProgramData\trx-remote\`

These files contain **secrets** (certificate key, password hashes): keep the backup in a safe place. To move a Windows station to Debian or to a Raspberry Pi, put its `station.toml` next to `install.sh` before installing. On a Pi, also keep a copy of the SD card (or SSD) once everything is set: it is the fastest way to start again if it fails.

7.4 Change a setting

- One part of the configuration, without going through the whole wizard again: only its questions are asked, Enter keeps each current setting.

To change	Menu
the transceiver, its port, the speed, the PTT (with a link test)	**2.5**
the sound cards	**2.6**
the HF spectrum through the second port (Icom)	**2.7**
remote power on and off of the transceiver (off by default)	**2.8**

To change	Menu
the receive audio gain and its limiter (received level shown, suggested value)	**2.9**
the allowed bands	**3.2**
the maximum power and the duration of a transmission (TOT)	**3.3**
network access and HTTPS	**4.5**

- A specific setting without a menu: menu **8.1** (checked copy, differences shown, old version kept). The description of each setting: https://bob56fr.github.io/openRemoteTRX/station-toml/
- A new device, a password to change, a lost phone: menu **4.3** (operator accounts).
- Start everything again (new transceiver, new PC): menu **2.1**; accounts and safeguards are kept.
- The full manual of openRemoteTRX Console: https://bob56fr.github.io/openRemoteTRX/console/en/ (or the `h` key in the console: `h 8.1`).

7.4 bis Fewer menus: the display level

openRemoteTRX Console shows more or fewer menus depending on the **level** chosen in menu **0** (“Language & level”) or, in the browser, with the Basic / Normal / Expert buttons at the top of the page:

Level	What is shown
Basic	start, stop, dashboard, full wizard, transceiver test, pairing a phone, accounts, diagnostic, report, update (and “reactivate the station” when it is in safety lockout)
Normal	in addition, all the usual settings (bands, power, alerts, visitors, features…)
Expert	in addition, temperature thresholds, `station.toml` by hand, automatic tests, transceiver qualification

The level only hides: a number typed (`3.3`) always opens its menu, and the browser search finds all the menus. Without a saved choice, the console starts in Basic. In the browser, the language can also be changed with one click at the top of the page (FR, EN, DE, ES).

7.5 The console in the browser

The same menus, in a web page: live station status, themes as tiles, search (“ntfy”, “TOT”…), help for each menu when the mouse hovers over it (at the first tap on a tablet). The chosen menu opens in a panel on the right (full screen on a phone), **without a terminal**: its questions become buttons (choices, Yes / No, Confirm), fields (number with its limits, password), and what the commands display scrolls in a dark frame, with the [OK], [ATTENTION] and [ERREUR] lines in colour. The configuration wizard (2.1) and its parts (2.5 to 2.7, 3.2, 3.3, 4.5), like the DX Cluster setting, also ask their questions as buttons and fields; each answer is written in the frame, under its question. A refusal is shown in red. “Other answer…” always lets you type your answer if a button is missing. The station's administration password (sudo) is asked once per session, in the panel. The dashboard (1.3) is shown full screen in the page: same sections as in the terminal (radio, SDR receiver if it is in service, audio, connections, health, computer, network with its “Detail” button, log), updated continuously, with the **STOP TX** button (it can only cut a transmission). Editing station.toml (8.1) is done in a large text field: “Check and save” has the station check the file, show the differences and ask for confirmation before writing; a wrong file is refused in red and nothing is written. “Open in the terminal”, at the top of the panel, replays any menu in the terminal at the bottom of the page. Same page on Debian, on a Raspberry Pi and on Windows.

- **On the station PC**: key **w** in the full screen of openRemoteTRX Console, or `trx-console --web` in a terminal. The browser opens by itself.
- **From any device on the local network or the VPN** (PC, tablet, phone): once, on the station, menu **4.6** “openRemoteTRX Console on the network”: put it in service, then choose the **administrator** operator accounts (yours). Then open `https://<station-address>:8444/` and log in with your usual username and password. Logged in to the control page with an administrator account, you also find the link in the “openRemoteTRX Console (administration)” section, under “Connection”.

Safety. An administrator account opens the station's configuration: strong password, not to be lent. The console can never be reached from the Internet (public addresses refused); after 5 failed attempts, the address is blocked for 15 minutes; each connection sends an alert to the phone (ntfy); the session closes after 30 minutes without action, and a new connection closes the previous one. The station's sudo password is still asked once per session, at the first administration menu.

8. Troubleshooting

8.1 The first reflex

1. **Diagnostic** (menu 6.1): it tells you what is wrong and what to do.
2. **Log** (menu 6.2): messages start with an “ORT-NNNN” code.
3. Each code has its own explanation page (cause, what to do): https://bob56fr.github.io/openRemoteTRX/erreurs/ (search field: type the code).

8.2 Common problems

Symptom	Likely cause	What to do
The diagnostic does not find the transceiver	transceiver off; wrong port (the transceiver's second port, or “Standard” on Yaesu); transceiver speed different from the one in the wizard; driver missing (Windows)	switch on; go through menu 2.5 again with the command port; set the speed (transceiver sheet)
No received audio	transceiver sound card taken by the desktop; microphone forbidden to applications (Windows); level too low (transceiver setting, gain in menu 2.9)	part 1.4; part 2, Windows tab; transceiver sheet
Transmission starts but nothing is modulated	in this mode the transceiver takes the voice from its microphone, not from USB (Icom: DATA MOD; Yaesu: MOD SOURCE and REAR SELECT)	transceiver sheet, “Voice through USB”
Only DATA modes can transmit	the wizard noted that the voice does not go through USB	set up the transceiver, then run the wizard again and answer “yes”
ALC at maximum, distorted audio	transceiver USB input level too high (Icom: USB MOD Level; Yaesu: RPORT GAIN)	part 3.4
The transceiver goes into transmission when the PC starts	an RTS or DTR line of the USB port keys the transmitter (Icom: USB SEND / USB Keying; Yaesu: RPTT SELECT, PC KEYING)	**unplug the USB**, set these settings to OFF (transceiver sheet)
Slow Raspberry Pi, choppy audio, transceiver USB dropping out	power supply too weak or overheating (`vcgencmd get_throttled` other than `throttled=0x0`)	official power supply, fan (part 2, Raspberry Pi tab)
“Microphone unavailable” in the page	page opened over HTTP	open the address with **https://**
Page unreachable from the phone	station stopped; address changed; network on the Public profile (Windows)	menu 1.1; part 4.1; part 2, Windows tab
Connected but no audio remotely	UDP does not get through (open port on the router)	VPN (part 5)
Dropouts every 15 to 30 s	two devices on the same VPN profile	one profile per device
Transmission cut by itself	maximum time reached, phone no longer responding, frequency outside the bands, local operator	normal: the reason is shown in the page and the log
Nothing responds any more, “station in safety lockout” alert	repeated failures	part 6.2
Transceiver test refused: “ORT-9313 … the station is running”	the test needs the transceiver for itself	stop the station (menu 1.2), run the test again, restart the station
A tool stops with “ORT-9312 … unexpected error”	software fault in an unforeseen case (new transceiver, sound card…)	the message gives the detail file (`plantages` folder): attach it to the help request (part 8.3)

8.3 Asking for help

1. Menu **6.3**: creates a troubleshooting report (zip on the Desktop, or in the account folder on a Pi without a desktop) **without any password** (Windows as well as Debian). From the web console, the **“Download the report”** button fetches it directly to your PC or phone. The traces of tools that stopped on an error (code

ORT-9312) are in the `plantages` folder of the station's state folder, and the report includes them. To read it yourself: menu **6.6**, which gives the link to the **analyser**; drop the zip there and the analysis runs in your browser (the report is sent nowhere).
2. From the web page: “Report a problem” section (text reviewed before sending).
3. Open an issue on https://github.com/Bob56fr/openRemoteTRX/issues with the **English** form (“Report a bug”), describe what you were doing, what you expected, what happened, and attach the report.
4. **Getting help remotely** (station-to-station monitoring, menu **6.5**, to be enabled first: part 9): the person helping you creates a 10-character code and reads it out to you; type it at choice 4, then compare out loud the **6 digits** shown on both sides. Different: answer “no”, the link is erased. Then choose what you send (“Diagnostic” recommended; frequency and mode only if you want). They then see the state of your station and its error codes, **masked** (no password, no address, no serial number), and can **ask** you for assistance mode (news every 30 seconds, for a limited time): you are the one who accepts (choice 7). They **cannot command anything** on your station. You stop everything whenever you want (choice 9). They can also ask you for your **troubleshooting report** (the one in point 1, fully masked); you can also send it yourself (choice 10). On their side, they get an alert on their phone if your station stops giving news. **Away from your station**, everything can also be done in the web page, “Monitoring” section: button “Get help: I have a code”, comparison of the 6 digits, choice of what is sent, answer to requests, end.

9. Optional features

All of them are **disabled** at the start. Enable them once the station is validated (parts 3 to 5): menu **5.5** “Features in service” switches them on or off by number (it also starts the feature's module); a feature that needs other settings is refused, with the menu where to make them.

Feature	What it is for	Where to enable it
Receive-only **Visitors**	accounts that listen to the station and see the screen, with **no** command at all (radio club, demonstration)	menu 4.2, then the “Visitor listening” switch in the page
DX Cluster	spots on the spectrum and badges on the band buttons	menu 5.1
FreeDV RADE V1 (reception)	FreeDV digital voice decoded by the station	decoders to compile (`sudo bash compiler-freedv.sh` in the Debian package folder), then menu 5.5 (“FreeDV reception”); the locator for FreeDV activity: `[freedv] locator` (menu 8.1)
SELCALL	group-call squelch in SSB	menu 5.5, then the SELCALL section of the page
SWR measurement	SWR curve of the antenna over a band, in FM, at reduced power	web page or application; the station transmits: allowed bands only
Group presence	see which friends equipped with openRemoteTRX are in a QSO, and join them	menu 5.3
Station-to-station monitoring	get help remotely from someone you trust, or help another operator: station state and error codes, masked and encrypted, without VPN or open port; no command possible towards the station being helped	menu 5.5 (“Monitoring between stations”), then menu 6.5 or the “Monitoring” section of the page (part 8.3)
Breizh Audio Box	voice processing on another PC of the local network	menu 5.2
Live broadcast	let a few **listeners** on the Internet hear the transceiver's audio, without an account on the station: they listen on the page of a broadcast server (private channel with username and password, or public); **DIFFUSION** indicator always visible in the page; your voice during your transmissions is broadcast only if you ask for it; outside your bands, silence for the listeners (unless set otherwise): their page gives the reason (outside the bands, transmission without your voice, no more audio from the transceiver), as do menu 5.6 and the 🔇 indicator; alert on the phone at the start and at the end	menu 5.6 (server and secrets given by the server administrator), menu 5.5 (“Live broadcast”), then menu 5.6 or the “Live broadcast” section of the page to start and stop
Replay of a recording	transmit again a recording of the reception	granted in the wizard; generally **forbidden** in the amateur service

Feature	What it is for	Where to enable it
SDR receiver (SDRplay)	second waterfall and listening (AM, FM, SSB, CW) of a receive-only SDR, next to the TRX; SDR audio muted during any transmission; TRX / SDR balance and, on headphones, one source per ear; waterfall zoom (“Zoom − / +” around the listened frequency, ◄► to move the window), colour palette, width (at most the one chosen in menu 2.4) and antenna A / B from the page; DX Cluster spots on the SDR waterfall (tap a spot: the SDR listens to it, the TRX does not move); the station slows down or suspends the SDR by itself if it is too loaded; SDR IN STANDBY when nobody has shown its section or listened to it for 1 minute (less load and heat), woken in a few seconds when the section is opened (“SDR: leaving standby, a few more seconds…”); SDR chosen and set in menu 2.4 of openRemoteTRX Console (search on all USB ports and hubs); “TRX QSY” to a frequency read on the SDR; SDR running: below the TRX frequency, a purple “SDR” line (frequency, mode, filter being listened to; tapping it opens the SDR section on its waterfall), on the right two bargraphs of the audio heard (TRX: tap = mute in the page, reset at each connection; SDR: tap = listening started or stopped, like “Listen”), “S-TRX” and “S-SDR” S-meters (S-SDR while listening to the SDR, approximate “≈” until it is calibrated: SDR on the TRX frequency with “QSY TRX → SDR”, a steady signal, then “Calibrate S-SDR to the TRX” in the SDR section; calibration kept in the browser)	first install the manufacturer's SDRplay API, then the station; `trx-sdr --list` gives the serial number; `[sdr]` and “sdr” in `[modules] run` (menu 8.1)

SDR receiver: protect its input. The SDR cannot withstand the power of your TRX. Separate antenna, or a switch and sequencer that ground the SDR input before any transmission: it is up to you to provide this, according to your equipment. The station cannot do it for you (a transmission with the transceiver microphone is only seen after the fact). The SDR is identified by its **serial number**: on a machine that has several, never enter the one of an SDR used by another application. Plug it into its own USB port, never into the transceiver's hub. The “USB ports” line of the COMPUTER card (dashboard, menu 1.3) tells you whether a USB bus is overloaded or whether the SDR is slowed down; the details and what to do are in the diagnostic (menu 6.1). A blue socket (USB 3) does not speed up a USB 2 SDR: it shares the 480 Mbit/s of the computer's USB 2 part.

Safety. Retransmitting to third parties what the station receives (visitors, broadcast listeners) makes you responsible, especially outside the amateur radio bands; by default, visitors and listeners hear only silence outside your bands (two independent settings).

Depending on the transceiver, two features of the page may be missing: the **transceiver's HF spectrum** (only through the second port of an Icom such as the IC-7300MK2; otherwise only the audio spectrum is shown) and the **FIL1/2/3 filter presets** (specific to Icom).

Glossary

The words of the guide, arranged by theme. The diagram of the building blocks is in part 0.

The hardware

Term	What it is
Transceiver, **TRX**, **Rig**	your radio transmitter-receiver
Station PC	the computer next to the transceiver, always on, that runs the station: a PC running Debian or Windows, or a Raspberry Pi
Raspberry Pi	small single-board computer, low power; the Pi 4 (4 GB) or the Pi 5 running Raspberry Pi OS 64-bit can serve as the station PC
Hardware watchdog	a chip of the Raspberry Pi that restarts it if the system freezes for more than 15 s; enabled by the installation
Station	the whole transceiver + PC + openRemoteTRX software; “start the station” = start its services
CAT	*Computer Aided Transceiver*: control of the transceiver by the computer (frequency, mode, transmission…)
Enhanced COM Port / **Standard COM Port**	the two serial ports that some Yaesu transceivers (FTDX10…) create on the PC through their USB cable: the first one is for CAT, the second one (RTS/DTR lines) is not used by the station
Transceiver sound card	the sound card built into the transceiver, seen by the PC through the USB cable (“USB Audio CODEC” on Yaesu, “USB Audio Device” on the IC-7300MK2): received audio to the PC, voice to the transceiver
Dummy load	50 Ω resistor that replaces the antenna: the transceiver transmits without radiating. Mandatory for all tests
ALC	transceiver meter that shows whether the modulation is too strong; used to set the voice level
SWR	standing wave ratio: matching of the antenna. The station can measure it across a band
VFO A / VFO B	the transceiver's two working frequencies; the page shows which one is active
Split	receive on one frequency, transmit on another

The station software

Term	What it is
openRemoteTRX	the name of the project; the Android app has this name, the station is also called “openRemoteTRX”
Hamlib	the free library that can talk to about 300 transceiver models; each model has a **number** (IC-7300MK2 = 3094, FTDX10 = 1042)
rigctld	the Hamlib program that keeps the link with the transceiver; it listens only on the PC itself, never on the network
Gateway (`trx-gateway`)	the heart of the station: serves the web page, checks the accounts, carries the audio and **decides on transmission**
Supervisor (`trx-supervisor`)	the independent guardian: watches the gateway, rigctld and the PC; it can **only cut** a transmission, restart a service or put the station in safety lockout; sends the alerts
openRemoteTRX Console (`trx-console`)	the station's control panel in a terminal: install, configure, start, test, update, troubleshoot (numbered menus “2.1”, “2.2”…)
Configuration wizard (`trx-setup`)	the questions that create the configuration (menu 2.1); it only reads from the transceiver
Dashboard (`trx-monitor`)	the live status in a terminal (menu 1.3); a single action: STOP TX
Vital signs dashboard	read-only page (port 8450, Debian and Raspberry Pi) for a small screen near the transceiver or a phone
Indicator	the station's status light in the top bar (Debian) or next to the clock (Windows); read-only

Term	What it is
Guided test (`trx-hwtest`)	the step-by-step test of the real transceiver (menu 2.2)
Qualification (`trx-qualif`)	the complete exploration of what Hamlib declares for a transceiver (menu 8.3), for experts
`station.toml`	the station's configuration file: transceiver, ports, accounts, allowed bands, safety settings
Transceiver profile	a transceiver configuration saved under a name, to change transceivers (menu 2.3)
Module	an optional feature that runs next to the gateway (DX Cluster, group presence…)
Signed update	the Debian packages are signed: the update over the Internet (menu 7.1) refuses a package whose signature does not match; an unsigned zip (menu 7.3) asks for a confirmation

User side

Term	What it is
Web page	the station's interface in a browser, at the address `https://<station-address>:8443`
Android app	the openRemoteTRX app: the same as the page, plus transmitting with the screen off, the headset, the car
Android Auto	the app's display on the car screen: listening, memories, PTT, no graphics
Driving mode	the app's reduced screen in the car: no touch PTT
Operator account	username + password that allow controlling the transceiver and transmitting; **one per device** (`PC`, `S25`…), managed in menu 4.3
Visitor	receive-only account: listens and sees the screen, no command
Pairing	adding the station to the app, with the **QR code** of menu 4.1

Transmission safety

Term	What it is
PTT	*Push To Talk*: switching to transmit
Toggle PTT	one press to transmit, one press to go back to receive (never “as long as you press”)
STOP	immediate return to receive, always available
TOT	*Time Out Timer*: maximum duration of a transmission (90 s by default, 120 s at most), with a beep 10 s before the cut
Absolute TOT	duration beyond which the supervisor cuts any transmission, even one made with the transceiver's microphone (135 s; 5 minutes if the operator touched the transceiver just before or during it)
Heartbeat	the sign of life the phone sends twice a second during transmission; if it is missing for 2 s, back to receive
Allowed bands	the frequency ranges where the station accepts to transmit; empty list = receive only
Local operator	someone using the transceiver in front of it: remote transmission is blocked during their transmission and then for 2 minutes
Transmit audio gate	the audio path to the transceiver, opened by the gateway only during an authorised transmission
Safety lockout	the state the supervisor puts the station in after repeated failures: stopped safely (receive, access cut), it waits for an intervention; reactivate it with menu 1.4 (“Reactivate the station”)
Transceiver timer (TX TIME OUT TIMER)	the transceiver's own cut, independent of the station: the last barrier

The network

Term	What it is
Local network (LAN)	the home network, behind the internet box
Port 8443	the port where the gateway receives the page and the app
HTTPS, **certificate**	the encryption of the link; the station creates its own certificate (“self-signed”): the browser warns the first time
Certificate fingerprint	the summary of the certificate that the app remembers to refuse a fake station
WebRTC	the technique that carries the live audio, over **UDP**
VPN	an encrypted tunnel that puts the phone “inside” the home network, from anywhere (WireGuard, Tailscale...)
VPN profile	the keys of one device in the VPN: one profile per device, never shared
Open port	the internet box letting traffic in from the Internet to the station; requires a real certificate and a strong password
SSH	the encrypted connection that opens a terminal of another machine from your PC: `ssh -l <account> <address>`; used to drive a Raspberry Pi without a screen
CGNAT	address shared by the provider between several customers: no port can be opened, a VPN is needed
ntfy	the notification service that carries the station's alerts to your phone
ORT code	the number of a station message (“ORT-5031”); each code has its explanation page

10. Summary: where are you?

Tick as you go. On the guide's web page, boxes are ticked online and stay ticked in your browser; on GitHub, print this page or copy the list.

0. Before you start

- [ ] Remote operation conditions checked for my licence
- [ ] Dummy load available for the tests
- [ ] System chosen (Debian recommended, Raspberry Pi or Windows) and package downloaded

1. Prepare the transceiver

- [ ] Manufacturer's USB driver installed before plugging in the cable (Windows)
- [ ] Transceiver command port identified (transceiver sheet)
- [ ] Transceiver CAT speed noted, menus of the transceiver sheet set
- [ ] Voice through USB set in each mode used
- [ ] No RTS / DTR line of the USB port keys the transmitter
- [ ] VOX off, transceiver transmit timer at 3 min, maximum power set
- [ ] Transceiver sound card removed from the PC's default sound

2. Install the station

- [ ] Software installed (install.sh or installer.cmd)
- [ ] Session reopened (Debian, Raspberry Pi); Private network profile and sleep disabled (Windows)
- [ ] Raspberry Pi: official power supply, fan, `throttled=0x0`

3. Integrate the TRX and test it

- [ ] Wizard completed (model of my transceiver, command port, PTT by CAT, bands of my licence)
- [ ] Diagnostic without failure
- [ ] Guided test in receive only passed
- [ ] Guided test with transmission passed on a dummy load
- [ ] Station started, page open, modulation set with the ALC (transceiver USB input level)
- [ ] Test 1: STOP cuts the transmission
- [ ] Test 2: cut after 90 s, warning 10 s before
- [ ] Test 3: link cut during transmission, back to receive
- [ ] Test 4: screen locked during transmission, back to receive
- [ ] Test 5: transmission refused outside my bands
- [ ] Test 6: transmission with the transceiver microphone, remote PTT blocked
- [ ] Test 7: STOP TX from the dashboard

- [ ] Test 8: station stopped during transmission
- [ ] Test 9: no transmission when the PC (or the Pi) starts
- [ ] Test results reported, if my transceiver is not yet validated (issue “<my transceiver>: commissioning tests”)

4. Use on the local network

- [ ] IP address reserved for the station PC in the router
- [ ] Web page opened over https, certificate accepted, microphone allowed
- [ ] One account per device (menu 4.3), long passwords
- [ ] Android application paired with the QR code (menu 4.1), transmission test and STOP

5. Use remotely

- [ ] Choice made: VPN (recommended) or open port
- [ ] One VPN profile per device
- [ ] From 4G: page, received audio, transmission on a dummy load, STOP
- [ ] From 4G: airplane mode during transmission, back to receive

6. Unattended station

- [ ] ntfy alerts received (long and secret topic), “Station in service” message seen
- [ ] PC set to power on again after a power cut (a Pi does it by itself)
- [ ] UPS, surge protector, transceiver ventilation
- [ ] I know how to reactivate the station after a safety lockout (menu 1.4) once the cause is found

7. Day-to-day operation

- [ ] I know how to start, stop and monitor the station
- [ ] I know how to update and go back to the previous version
- [ ] Configuration and certificate backed up elsewhere
- [ ] Transceiver profile saved (menu 2.3)

Station in service

- [ ] First transmission on the antenna, in my bands, with identification