



piSync

User Manual

Version 1.1

Keeps a KiwiSDR or WebSDR web receiver in step with piHPSDR or an FLRig transceiver

October 2026

Contents

1 Introduction.....	3
1.1 What piSync does.....	3
1.2 One-way and Two Way Sync.....	3
1.3 How it works.....	3
2 What you need.....	3
3 Installing piSync.....	4
3.1 Graphical installer (no terminal needed).....	4
3.2 Setup script (terminal).....	6
4 Preparing your radio.....	7
4.1 piHPSDR.....	7
4.2 FLRig.....	7
5 Starting piSync.....	8
5.1 The piSync launcher (no terminal needed).....	8
5.2 From a terminal.....	10
5.3 Running as a service (optional).....	11
6 The piSync page.....	11
6.1 Choosing a receiver.....	11
6.2 Naming your receivers.....	12
6.3 The control bar.....	13
6.4 Status and readout.....	13
7 Using KiwiSDR and WebSDR receivers.....	14
7.1 Starting audio.....	14
7.2 Frequency, mode and band.....	14
7.3 Finding receivers.....	15
8 Two Way Sync.....	15
8.1 Turning it on and off.....	15
8.2 What is sent.....	15
8.3 Rules piSync follows.....	15
8.4 Messages.....	16
9 Configuration reference.....	16
10 Troubleshooting.....	17
11 Appendices.....	18
11.1 Files and locations.....	18
11.2 Safety summary.....	18
11.3 Screenshot placeholders.....	18
11.4 Credits.....	19

1 Introduction

1.1 What piSync does

piSync links your radio to a web receiver in your browser. Tune your radio, and the KiwiSDR or WebSDR you are listening to jumps to the same **frequency, mode and band**. It is a browser-based take on what CatSync does on Windows.

It works with:

- **piHPSDR** (piHPSDR-Improved), read over its TCP CAT port.
- **Any transceiver supported by FLRig**, read over FLRig's XML-RPC server.
- **KiwiSDR** receivers and classic PA3FWM **WebSDR** receivers (for example the one at Hack Green). The type is detected automatically.

1.2 One-way and Two Way Sync

By default sync runs **from the radio to the web receiver**: the receiver follows your radio and your radio is only ever read.

With **Two Way Sync** ticked, sync also runs **from the receiver to the radio**. If you click on the waterfall, type a frequency or change mode in the web receiver, your radio is tuned to match. Two Way Sync is off by default and is described in detail in chapter 8.

1.3 How it works

piSync is a small server that runs on your computer, plus a control page you open in your web browser. The server:

- talks to your radio (piHPSDR CAT or FLRig) to learn its frequency, mode, active receiver and whether it is transmitting;
- acts as a go-between for the web receiver you choose, so that the receiver and the control page share one address. That is what allows piSync to use the receiver's own tuning functions rather than simulating mouse clicks or key presses.

The control page (http://127.0.0.1:8090/_pisync/) shows what the radio is doing and holds the receiver in a frame below it.

Note: piSync never keys your transmitter. It never sends **TX**;, **RX**; or **ZZTX** to piHPSDR and never calls **rig.set_ptt** in FLRig. With Two Way Sync it only ever changes frequency and mode, and never while the radio is transmitting.

2 What you need

- A Linux computer (Raspberry Pi OS, Linux Mint, Kubuntu and similar all qualify) with **Python 3.10 or newer**.

- The system packages `python3-venv` and `python3-tk` (the installer tells you if they are missing).
- A web browser. Chrome-based browsers and Firefox both work; see chapter 7 for the audio start-up click.
- Either **piHPSDR** with its TCP CAT server enabled, or **FLRig** with its XML-RPC server enabled.
- An internet connection to reach the KiwiSDR or WebSDR you want to listen to.

3 Installing piSync

Put the piSync folder where you want to keep it, then install it using either the graphical installer or the setup script. Both do the same job: they create piSync's private Python environment (`.venv`) inside the folder, install the requirements, and add a **piSync** launcher to your application menu (Network / Ham Radio) and, optionally, to the Desktop.

3.1 Graphical installer (no terminal needed)

Open the piSync folder in your file manager and double-click **piSync-installer**. Choose *Run* if asked. If it only opens as text, right-click it, open *Properties* > *Permissions* and tick *Allow executing file as a program*. The installer needs the system `python3-tk` package, which most desktop distributions include.

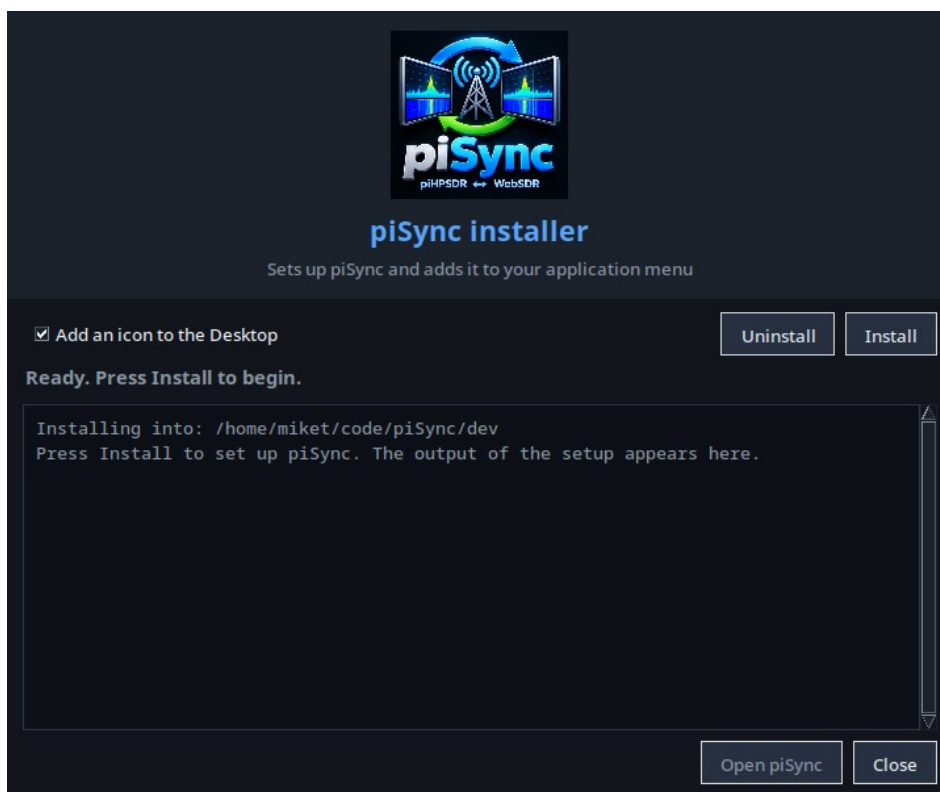


Figure 1: The piSync installer, ready to start

The window shows the piSync logo, a tick-box and three groups of buttons:

Control	What it does
Add an icon to the Desktop	Ticked by default. Untick it if you only want the application-menu entry.
Install	Runs the setup. The output of the setup scrolls by in the black field and a progress bar shows the installer is working.
Uninstall	Removes the launchers. The piSync folder and its .venv are left alone.
Open piSync	Greyed out until an installation has completed successfully; then starts the piSync launcher.
Close	Greyed out while the installer is working, so it cannot be closed half way.

Press **Install** and wait. If the **python3-venv** package is missing, the installer asks before installing it and the desktop shows its own password box. When the output stops, a *piSync is installed* message appears and **Close** becomes active. If something fails the output says which step failed; read the messages and press **Install** again.

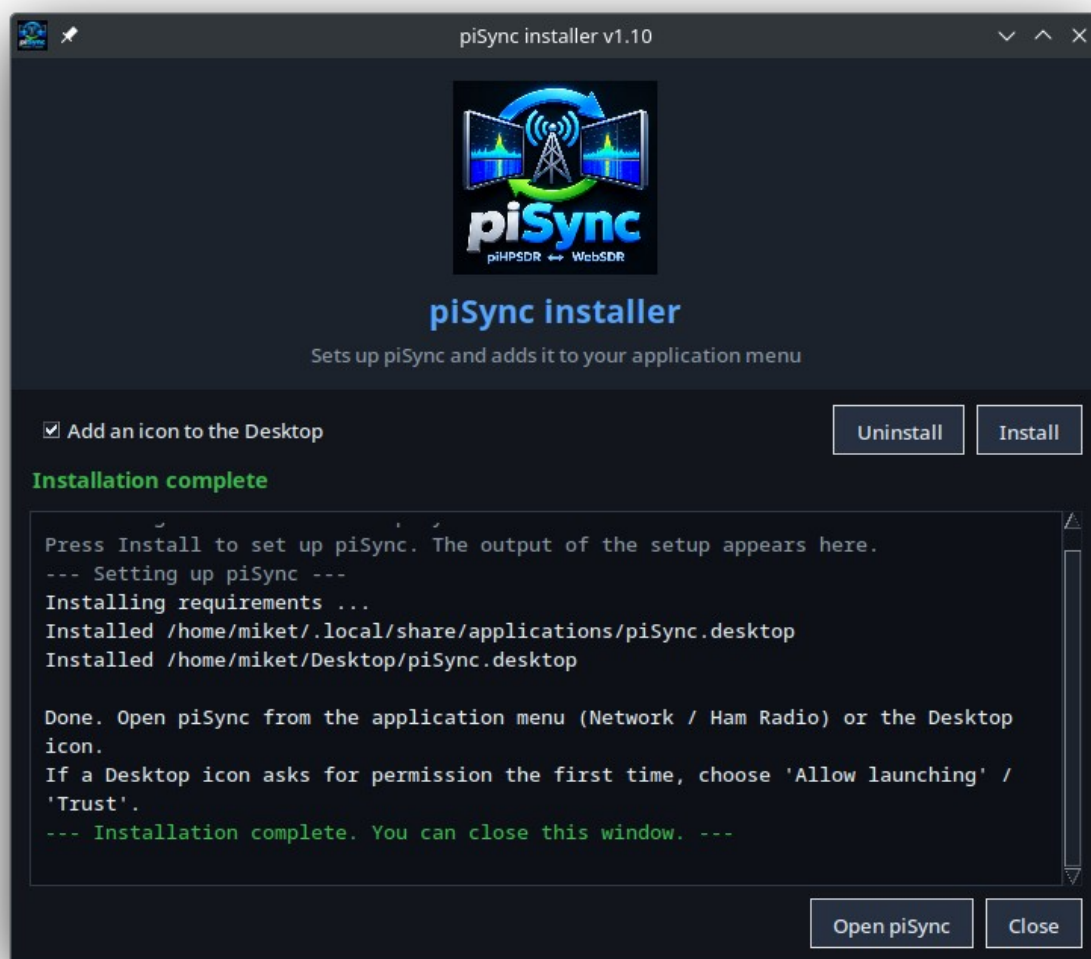


Figure 2: The installer while setup is running

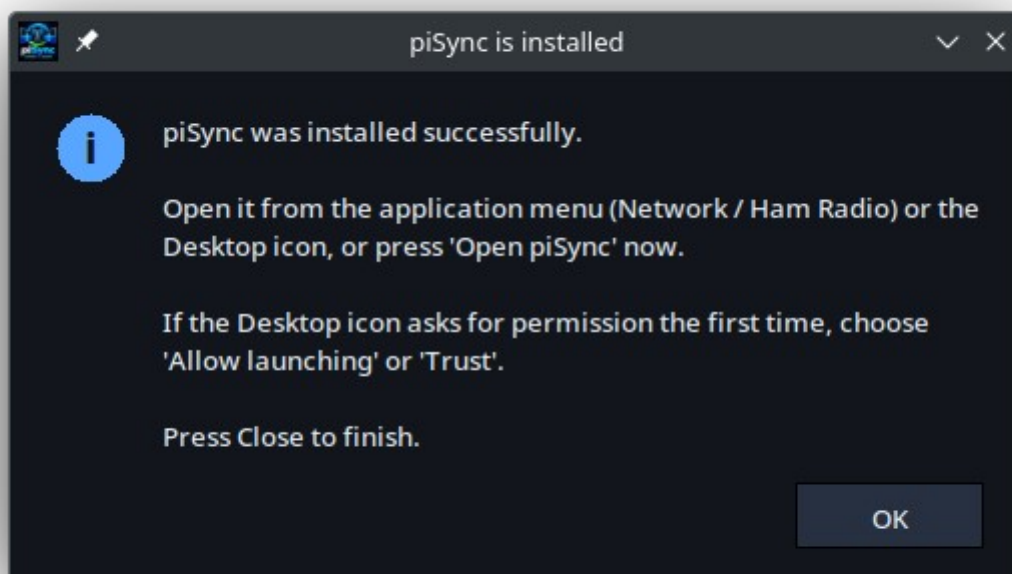


Figure 3: Installation finished

If a Desktop icon asks for permission the first time, choose *Allow launching* or *Trust*.

3.2 Setup script (terminal)

From the piSync folder run:

```
./setup.sh
```

If it reports missing packages it prints the exact command to run, normally:

```
sudo apt install python3-venv python3-tk
```

Options:

Command	Effect
<code>./setup.sh</code>	Set up piSync and add it to the application menu and the Desktop.
<code>./setup.sh --no-desktop-icon</code>	As above, but only the application-menu entry.
<code>./setup.sh --uninstall</code>	Remove the launchers (the folder and <code>.venv</code> stay).
<code>./setup.sh --help</code>	Show these options.

Note: Run the installer or `./setup.sh` again if you move the piSync folder: the launchers contain the folder's path and need refreshing.

4 Preparing your radio

4.1 piHPSDR

1. Start piHPSDR.
2. Open its **RigCtl** menu and enable **TCP CAT** (default port **19090**). piSync uses piHPSDR's Kenwood-style CAT commands.
3. If piHPSDR runs on a different computer, note its address and set it in `config.toml` (chapter 9).

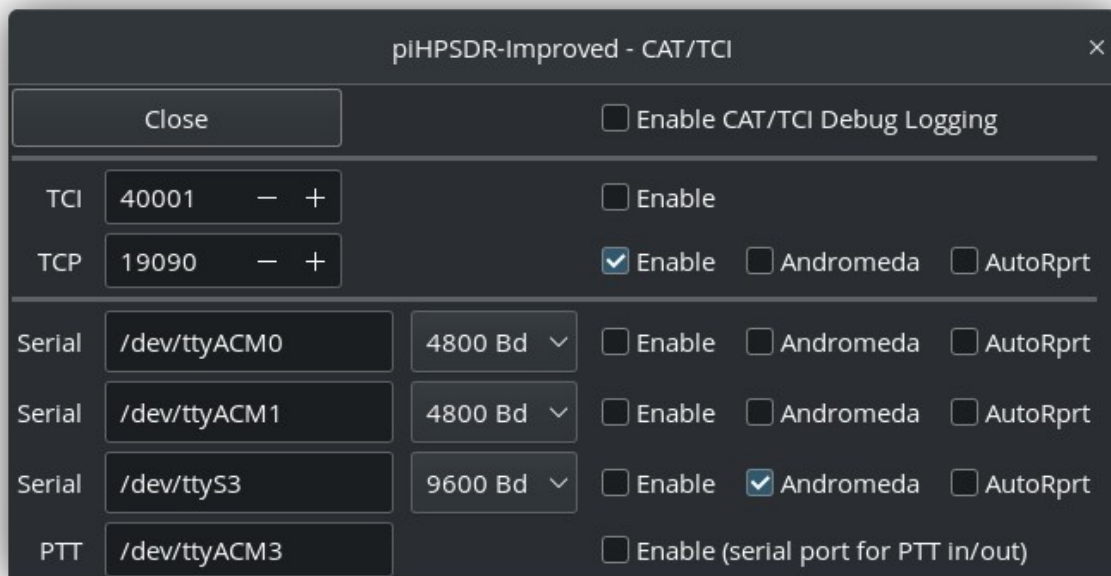


Figure 4: Enabling TCP CAT in piHPSDR

piHPSDR ties receiver 1 (RX1) to VFO A and receiver 2 (RX2) to VFO B. piSync asks which receiver is active and follows that receiver's VFO. The control page shows **RX1 · VFO A** or **RX2 · VFO B**, and switching receivers retunes the web receiver.

4.2 FLRig

1. Start FLRig and connect it to your transceiver as usual.
2. Enable the XML-RPC server: **Config > Setup > UI > Server** (default port **12345**).
3. In the piSync launcher, tick the FLRig box instead of the piHPSDR box (chapter 5), or set `type = "flrig"` in `config.toml`.

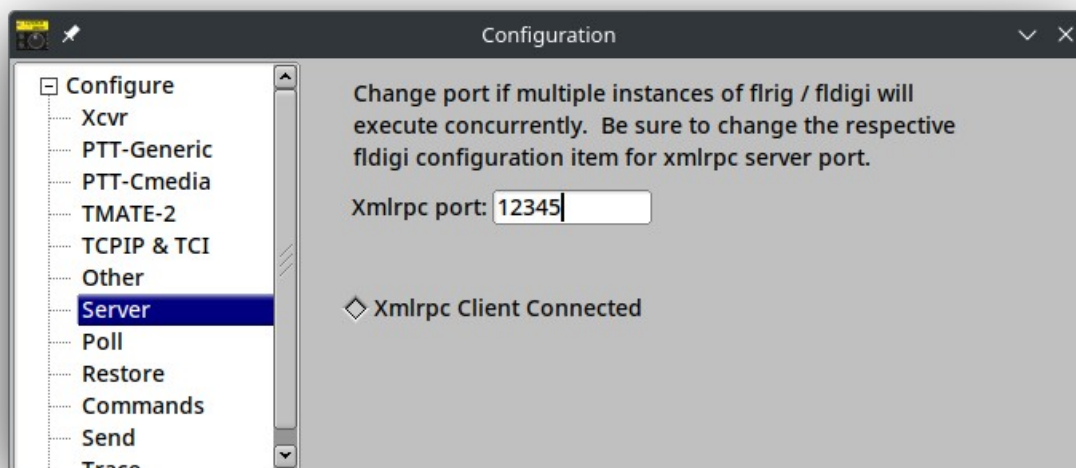


Figure 5: Enabling the XML-RPC server in FLRig

piSync follows whichever VFO FLRig reports as in use (A or B). **Mute on TX** only works if your rig and FLRig report PTT.

5 Starting piSync

5.1 The piSync launcher (no terminal needed)

Double-click the **piSync** icon on the Desktop or in the application menu. The window shows the piSync icon, its version and a one-line description. The version is read from the **VERSION** file and shown as vX.XX, so release 1.1 appears as v1.10.

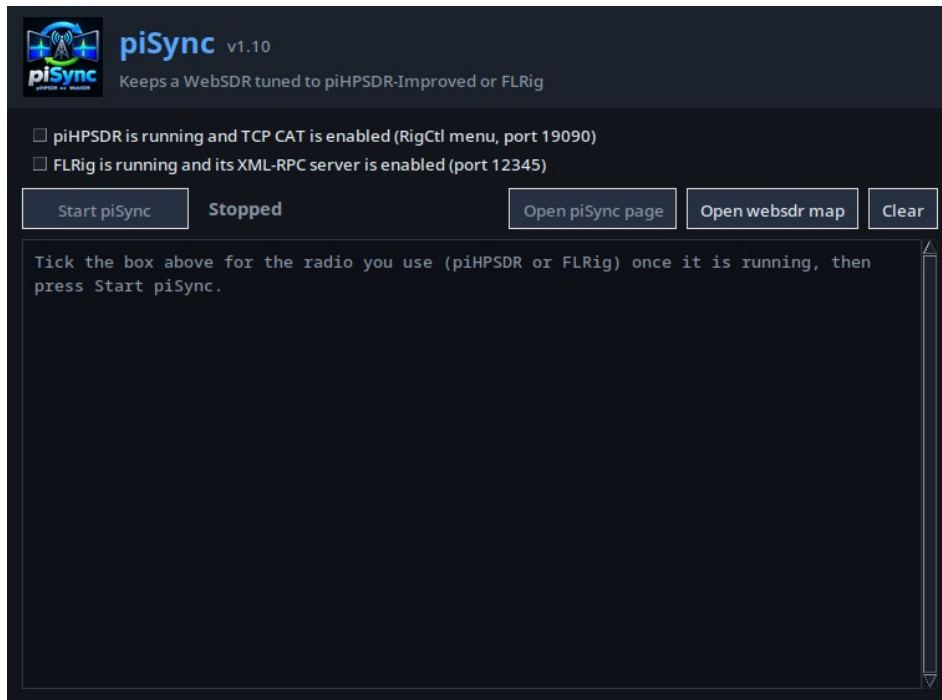


Figure 6: The piSync launcher, stopped

1. Start piHPSDR (or FLRig) and enable its CAT / XML-RPC server (chapter 4).
2. Tick "**piHPSDR is running and TCP CAT is enabled**" or "**FLRig is running and its XML-RPC server is enabled**". The two boxes are alternatives: ticking one unticks the other. **Start piSync** stays greyed out until one is ticked.
3. Press **Start piSync**. Its output scrolls in the field below, and the button becomes **Stop piSync**. The two tick-boxes are locked while piSync runs.
4. Press **Open piSync page** to open the control page in your default web browser. This button is only enabled while piSync is running. If no browser can be started automatically, the launcher shows the address to type in.

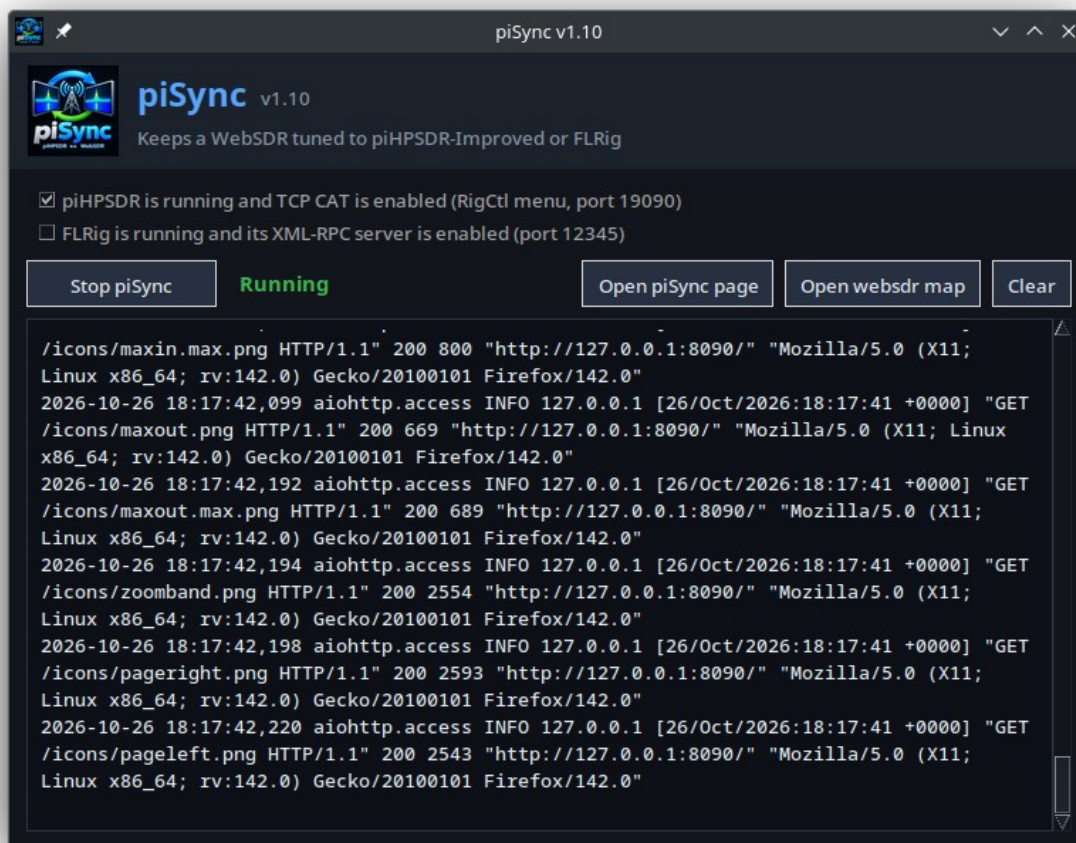


Figure 7: The launcher with piSync running

Button	What it does
Start piSync / Stop piSync	Starts or stops the server. Closing the launcher window also stops piSync.
Open piSync page	Opens the control page (normally http://127.0.0.1:8090/_pisync/) in your browser. Enabled only while piSync runs.
Open websdr map	Always enabled. Opens https://rx-tx.info/map-sdr-points in your browser, a handy way to find a receiver.
Clear	Empties the output field.

If the web port is already in use, for example because piSync is already running in a terminal, the launcher says so in its output.

5.2 From a terminal

Edit `config.toml` first if piHPSDR or FLRig is on another computer, then from the piSync folder:

```
.venv/bin/python -m piSync
```

Option	Meaning
<code>--config FILE</code>	Use a different configuration file (default <code>config.toml</code> in the current folder).
<code>--source pihpsdr / --source flrig</code>	Choose the radio to follow, overriding <code>[source]</code> in the configuration.
<code>-v, --verbose</code>	Show debug messages.

piSync prints the address to open, normally `http://127.0.0.1:8090/_pisync/`.

5.3 Running as a service (optional)

The file `piSync.service` is an example systemd unit that starts piSync at boot. It contains absolute paths and a user name for the computer it was written on, so edit `WorkingDirectory`, `ExecStart` and `User` to match your installation before copying it to `/etc/systemd/system/` and enabling it with `sudo systemctl enable --now piSync`.

6 The piSync page

Open the control page in your browser. Along the top is the control bar; the receiver fills the rest of the window.

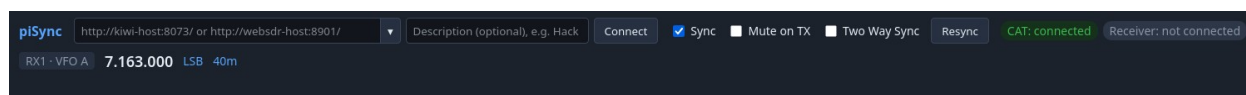


Figure 8: The control bar before a receiver is connected

6.1 Choosing a receiver

Type the address of a KiwiSDR or WebSDR in the first box, for example `http://g3sdr.com:8073/` or `http://hackgreensdr.org:8901/`, and press **Connect**. The receiver loads in the frame below. The first time, click once inside a Kiwi to start audio (browser autoplay rules). On a WebSDR click its green audio start button.

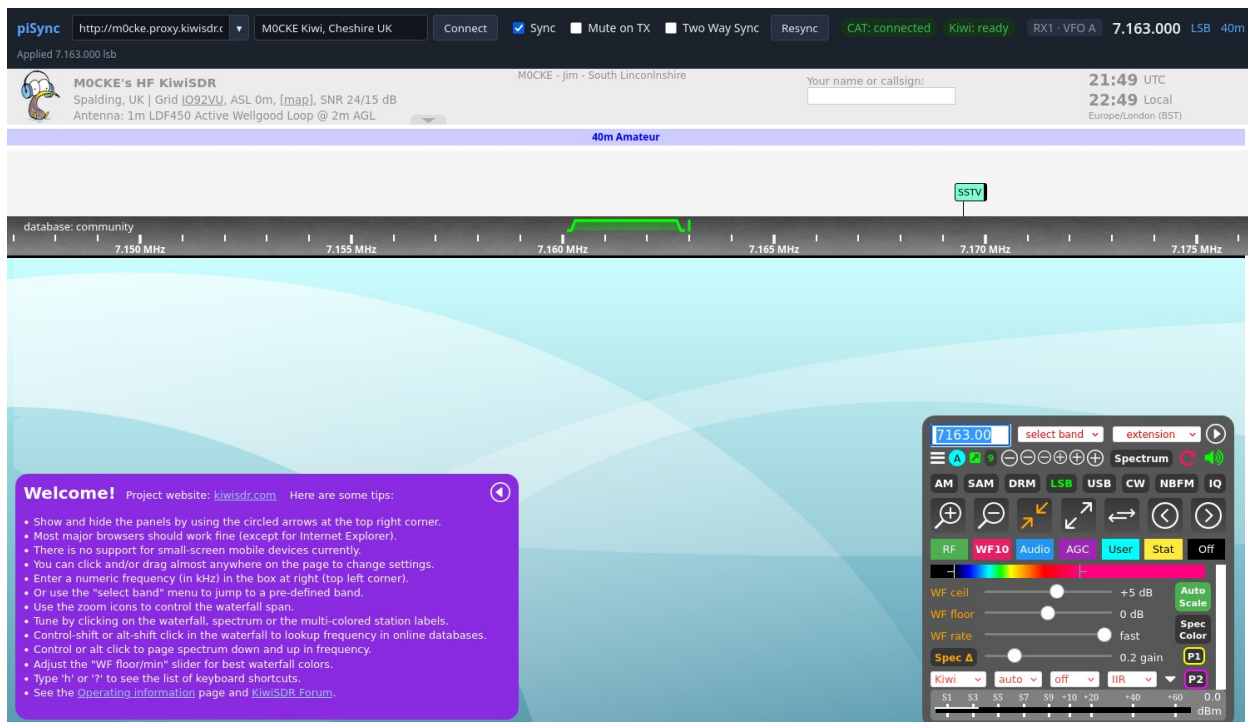


Figure 9: piSync connected to a KiwiSDR: the Kiwi is tuned to the radio's frequency and mode

6.2 Naming your receivers

Raw addresses rarely say where a receiver is. Type your own name in the **Description (optional)** box beside the address before pressing Connect, for example *Hack Green WebSDR, UK*. piSync remembers the address and the description together.

The **drop-down arrow** shows every receiver you have used, most recent first. The list shows **your description**, with the address in small type underneath; a receiver with no description shows just its address. Click a row to fill in both boxes and connect.

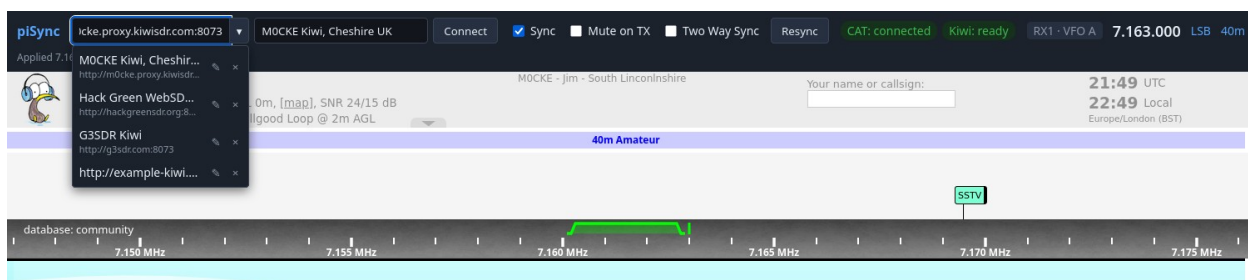


Figure 10: The receiver drop-down showing descriptions, with the address underneath

In the list	What it does
Click a row	Connects to that receiver and fills in its address and description.
Pencil button	Asks for a new description for that receiver. Leave it empty to remove the description.
Cross button	Removes the receiver from the list.
Arrow keys, Enter, Esc	Move through the list, connect, or close it.

Connecting again to a receiver that is already in the list keeps its description unless you type a new one. If you type a different address, the description box is cleared so the new address does not inherit the old name.

The list is saved in `~/.config/piSync/urls.json` (change the location with `urls_file` under `[server]`), so it is back after piSync restarts or when you open the page from another browser. Each entry is a pair of `url` and `name`. You can edit the file by hand while piSync is stopped; an older file that contains plain addresses still loads.

6.3 The control bar

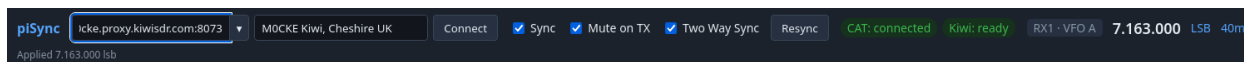


Figure 11: The control bar with Mute on TX and Two Way Sync ticked

Control	What it does
Address box and drop-down	The receiver to use (see above).
Description box	Your own name for the receiver, shown in the drop-down.
Connect	Loads the receiver and remembers it.
Sync	When ticked the receiver follows the radio. Untick to stop following (the default is set by <code>enabled</code> under <code>[sync]</code>).
Mute on TX	Mutes the receiver's audio while your radio transmits, to avoid audio feedback, and restores it afterwards. A mute you set yourself is left alone. Remembered by the browser; the default is <code>mute_on_tx</code> under <code>[sync]</code> .
Two Way Sync	Also tunes your radio when you tune the receiver (chapter 8). Off by default.
Resync	Pushes the radio's frequency and mode to the receiver again now, for example after you have tuned around in the receiver.

6.4 Status and readout

Item	Meaning
CAT: connected / disconnected (or FLRig:)	Whether piSync can talk to the radio. Green when connected, red when not.
Kiwi: ready / loading... (or WebSDR:)	Whether the receiver has finished loading and can be tuned. The name shows which type of receiver was detected. Receiver: not connected appears before you press Connect.
RX1 · VFO A / RX2 · VFO B	The radio receiver and VFO piSync is following (shown in amber for RX2).
Frequency, mode, band	The radio's current frequency in MHz, its mode (LSB, USB, CW, FM, AM...) and the amateur band, e.g. 40m.
TX	A red badge shown while the radio is transmitting.
Message line	The last action or problem, for example <i>Applied 7.163.000 lsb</i> or <i>Not applied: ...</i> (see chapter 10).

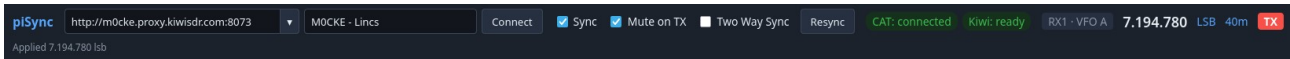


Figure 12: The TX badge while transmitting

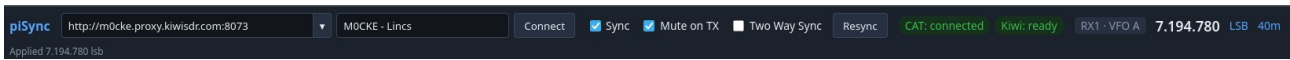


Figure 13: The control bar connected to a WebSDR (note the right indicator reads WebSDR: ready)

7 Using KiwiSDR and WebSDR receivers

7.1 Starting audio

- **KiwiSDR:** click once inside the Kiwi (the *Click to start KiwiSDR* overlay) to start audio. Browsers do not allow pages to start audio on their own.
- **WebSDR:** click the green audio start button on the receiver page (or wait for the audio to start).

7.2 Frequency, mode and band

- The receiver is tuned to the radio's **dial frequency**. A frequency outside the receiver's coverage is reported in the message line and skipped. A WebSDR only covers its own bands (Hack Green: 160, 80, 60, 40, 20 and 17 m), so a 30 m or 6 m frequency is skipped there.
- When the **band** changes, piSync re-zooms a Kiwi's waterfall to the band (set `zoom_on_band_change = false` to switch this off). On a WebSDR a band change fully zooms the waterfall out.
- On a **WebSDR** modes are reduced to its five: LSB, USB, CW, AM and FM. Kiwi's narrow and wide variants and SAM become the plain mode. CW is tuned so the pitch lands on the dial frequency.

The radio's mode is converted to a receiver mode like this (change it with `[modes]` in `config.toml`):

Radio mode (CAT digit)	Kiwi mode	WebSDR mode
LSB (1)	lsb	LSB
USB (2)	usb	USB
CW (3) and CW-R (7)	cw	CW
FM (4)	nbfm	FM
AM (5), SAM	am	AM
DIGL (6)	lsb	LSB
DIGU (9)	usb	USB

DSB, SPEC and DRM have no Kenwood equivalent and are treated as LSB, as piHPSPDR itself does.

7.3 Finding receivers

The launcher's **Open websdr map** button opens a map of public receivers. Copy a receiver's address into piSync's address box. Please respect each receiver operator's time limits and rules.

8 Two Way Sync

With **Two Way Sync** ticked, changes you make in the web receiver are sent to your radio, as well as the radio's changes being sent to the receiver. The control bar's checkbox is the only switch you need.

8.1 Turning it on and off

1. Connect to a receiver and wait for the receiver indicator to turn green (*Kiwi: ready* or *WebSDR: ready*) and for the first *Applied...* message.
2. Tick **Two Way Sync**.
3. Tune the receiver: click the waterfall, type a frequency or choose a mode. After you stop for about a third of a second your radio follows.

The message line shows *Sent to radio: 7.074.000 USB* each time something is sent. Untick the box to stop; nothing more is sent to the radio.

Important: Two Way Sync changes your radio's frequency and mode. It is **off by default** and is switched off again every time piSync restarts, so your radio is never writable unless you have just turned it on. To have it on at start-up, set `radio_follows = true` under `[sync]` in `config.toml`.

8.2 What is sent

Radio	Sent	Notes
piHPSDR	Frequency (FA / FB) and, on RX1, mode (MD)	The frequency goes to the VFO of the receiver in use. piHPSDR's CAT cannot set RX2's mode, so with RX2 active only the frequency is sent.
FLRig	Frequency and mode (<code>rig.set_vfoA/B</code> , <code>rig.set_modeA/B</code>)	The mode is chosen from the names your rig offers; if none matches only the frequency is sent.

8.3 Rules piSync follows

- **Never while transmitting**, and not when the radio is not connected.
- **Frequency and mode only.** Never PTT, TX, receiver selection or any other setting.
- **Last change wins.** If you change the radio and the receiver at nearly the same time, the most recent change is kept. piSync does not repeat its own changes back and forth.
- A receiver mode with no radio equivalent (DRM, IQ) changes the frequency only. Narrow and wide variants count as the same mode (LSB and the Kiwi's narrow LSB are both LSB).

- The receiver is only read after it has been tuned to the radio once, so a freshly loaded receiver's default frequency never overwrites your radio.

8.4 Messages

Message	Meaning
<i>Radio not updated: radio busy or not connected</i>	The radio is transmitting or piSync has lost its link to it.
<i>Radio not updated: radio_follows is disabled</i>	The piSync server has Two Way Sync switched off (for example after a restart); the checkbox unticks itself. Tick it again.
<i>Could not change Two Way Sync: ...</i>	The page could not reach the piSync server.

9 Configuration reference

piSync reads `config.toml` from the folder it is started in (or the file given with `--config`). All settings are optional. Lines starting with `#` are comments. Restart piSync after editing.

Setting	Default	Meaning
<code>[server] host</code>	<code>127.0.0.1</code>	Address piSync listens on. Use <code>0.0.0.0</code> to reach it from another computer on your LAN. See the security note below.
<code>[server] port</code>	<code>8090</code>	Port of the control page.
<code>[server] urls_file</code>	<code>~/.config/piSync/urls.json</code>	Where the receiver drop-down list (addresses and descriptions) is saved.
<code>[cat] host</code>	<code>127.0.0.1</code>	Where piHPSDR runs.
<code>[cat] port</code>	<code>19090</code>	piHPSDR's TCP CAT port.
<code>[flrig] host</code>	<code>127.0.0.1</code>	Where FLRig runs.
<code>[flrig] port</code>	<code>12345</code>	FLRig's XML-RPC port.
<code>[source] type</code>	<code>"pihpsdr"</code>	<code>"pihpsdr"</code> or <code>"flrig"</code> . The launcher's tick-boxes and <code>--source</code> override it.
<code>[kiwi] url</code>	<code>(empty)</code>	A receiver to load at start-up; you can also set it from the page.
<code>[sync] enabled</code>	<code>true</code>	Start with Sync ticked.
<code>[sync] zoom_on_band_change</code>	<code>true</code>	Re-zoom the receiver's waterfall when the band changes.

Setting	Default	Meaning
[sync] radio_follows	false	Start with Two Way Sync on. The checkbox switches it at run time, and it resets to this value on restart.
[sync] mute_on_tx	false	Default for the Mute on TX checkbox.
[modes]	(none)	Overrides of the CAT mode digit to Kiwi mode table.

Example: use narrow USB for digital modes (DIGU is CAT digit 9):

```
[modes]
9 = "usn"
```

Note: Bind to **127.0.0.1** (the default) unless you really need LAN access. piSync passes your browser's requests on to whatever receiver you enter, so do not expose it to untrusted networks.

10 Troubleshooting

What you see	What to do
CAT: disconnected (red)	piHPSDR is not running, TCP CAT is not enabled, or the port is wrong. Check the RigCtl menu and [cat] port. piSync keeps retrying and reconnects by itself once the radio is available.
FLRig: disconnected	Start FLRig, enable its XML-RPC server and check [flrig] port (12345).
piSync server: reconnecting...	The page has lost its link to the piSync server. Check that piSync is running; the page reconnects automatically.
Receiver: not connected	No receiver has been loaded. Enter an address and press Connect.
Kiwi: loading... stays amber	The receiver is still starting, is full, or its address is wrong. Try again or choose another receiver.
<i>Not applied: ... is outside this WebSDR's bands</i>	The radio is on a frequency the WebSDR does not cover. Choose a receiver that covers it.
<i>Not applied: ... is outside this Kiwi's range</i>	The Kiwi's coverage (and any frequency offset) does not include that frequency.
<i>Not applied: unsupported Kiwi mode ... or mode ... not supported by this WebSDR</i>	The receiver cannot do that mode. Use [modes] to map it to one it can.
No sound from the receiver	Click once inside the Kiwi, or press the WebSDR's audio start button (see chapter 7). Check you have not muted the receiver yourself.
Receiver does not follow the radio	Check Sync is ticked and press Resync . Make sure the indicator says ready . The receiver follows the active radio receiver (RX1 or RX2).

What you see	What to do
Mute on TX does nothing (FLRig)	Your rig or FLRig does not report PTT, so piSync cannot see transmissions.
The launcher says the port is already in use	piSync is probably already running (in a terminal or from another launcher). Stop it, or change <code>[server] port</code> .
The launcher says piSync is not set up	The <code>.venv</code> folder is missing. Run the installer or <code>./setup.sh</code> (chapter 3).
Two Way Sync does not move the radio	See the messages in chapter 8. Check the radio is connected, not transmitting, and that the box is ticked after the receiver is <i>ready</i> .

Running piSync from a terminal with `-v` shows what it is doing, including every frequency change it receives from the radio and every change it sends back.

11 Appendices

11.1 Files and locations

File	Purpose
<code>config.toml</code>	piSync's settings (chapter 9).
<code>~/.config/piSync/urls.json</code>	The receiver list with your descriptions. A list of <code>{"url": ..., "name": ...}</code> entries.
<code>VERSION</code>	The release number shown in the launcher.
<code>setup.sh</code> , <code>piSync-installer</code>	Installation (chapter 3).
<code>piSync.service</code>	Example systemd unit.
<code>CHANGELOG</code>	History of changes, newest first.

11.2 Safety summary

- piSync reads from piHPSDR with `IF;`, `FR;`, `FB;`, `ZZME;` and `AI2;`, and from FLRig with `rig.get_*` calls.
- With Two Way Sync it may also write the frequency (`FA`, `FB`) and mode (`MD`) to piHPSDR, or call `rig.set_vfoA/B` and `rig.set_modeA/B` in FLRig. Nothing else.
- It never sends `TX;`, `RX;`, `ZZTX`, `FR<digit>;`, `FT<digit>;` or `rig.set_ptt`. The software's own tests check this.

11.3 Screenshot placeholders

The yellow boxes in this manual mark screenshots that could not be captured automatically (for example because they need your radio to be transmitting, or a window state that cannot be reached without changing your system). Replace each box with a picture in your word processor.

Figure	What to capture
Figure 2	The installer while it is running: output scrolling in the black field, progress bar beside the status line, Close greyed out.
Figure 3	The "piSync is installed" message box shown when the installation has finished.
Figure 4	The piHPSDR RigCtl menu with TCP CAT enabled and port 19090 visible.
Figure 5	FLRig's Config > Setup > UI > Server page with the XML-RPC server enabled (port 12345).
Figure 7	The launcher with the piHPSDR box ticked and piSync running: "Running" status, "Stop piSync" button, "Open piSync page" enabled and the CAT connection lines in the output field.
Figure 12	The control bar while the radio is transmitting: the red TX badge visible and, if ticked, Mute on TX active.
Figure 14	The control bar with Two Way Sync ticked and the message line showing "Sent to radio: <frequency> <mode>" after tuning the receiver.
Figure 15	The red message line showing a "Radio not updated: ..." message.

11.4 Credits

piSync builds on piHPSDR-Improved, FLRig, KiwiSDR and PA3FWM's WebSDR. Please respect each receiver operator's time limits and rules.