AetherCTR2 User Guide

Firmware for the Lynovation CTR2-Max-5 controller, working with FlexRadio through AetherSDR



The AetherCTR2 face and meters are based on **SmartMTR** by **Dariusz Socha, SQ5X** (**https://smartmtr.net/**), used with his permission.

Edition of 2026-10-13 · private build, not for distribution · covers AetherCTR2 at the time of writing, with the AetherSDR changes noted in section 13

Contents

1. What AetherCTR2 is
2. Connecting to AetherSDR
3. The face
4. Knobs, buttons and the dial
5. Touch: badges, pickers and digits
6. The settings list
7. Giving a knob your own setting
8. CW operation
9. The CW keyer panel
10. SSB and voice operation
11. The audio meter and the readouts
12. Tuning, ATU and the Tuner Genius XL
13. Requirements and version notes
14. Troubleshooting
15. Appendix: updating the firmware

1. What AetherCTR2 is

AetherCTR2 replaces the firmware in a Lynovation CTR2-Max-5 so it works as a hardware front panel for a FlexRadio, through AetherSDR's CTR2 USB relay. The controller connects to the PC by USB; AetherSDR passes its SmartSDR traffic to the radio and back. Everything the Max shows comes from the radio's own status, so what you change in AetherSDR appears on the Max and the reverse.

It adds:

- a SmartMTR-style face with an S-meter, the other slice, a 32-bar audio meter and transmit meters;
- CW: a built-in keyer (Iambic A/B, Ultimatic, Bug, Straight) that keys the radio, a zero-beat audio meter and a scrolling CW decode;
- a settings list of about 60 radio settings, and knobs you can assign to any of them;
- touch pickers for mode, band and antennas, and tuning by any frequency digit.

> AetherCTR2 has no registration key of its own. Lynovation's own firmware and registration are separate; going back to stock firmware is covered in the appendix.

2. Connecting to AetherSDR

1. Plug the Max into the PC by USB. It shows **Waiting for AetherSDR**.
2. In AetherSDR, open the **CTR2 Proxy** applet, set **Mode** to **USB**, pick **AetherCTR2** and press **Start**.
3. When AetherSDR is connected to a radio, the Max binds to your station and shows the face.

The Max follows the **active slice** of the AetherSDR station it binds to. When you change slice, band or mode in AetherSDR, the face follows.

The Max talks to the radio only through AetherSDR. If AetherSDR closes or the USB link drops, the Max returns to the waiting screen and anything it was keying is released.

Some features need a newer AetherSDR than the current release; see section 13.

3. The face



Top to bottom:

- **Slice row**: the slice letter (in AetherSDR's slice color), the mode, the RX antenna (blue), the TX antenna (red) and the **TX** badge.
- **Band badge** and **frequency**. The frequency turns red while tune lock is on.
- **S-meter**, linear in dB from S0 to S9+40, with SmartMTR ballistics and peak marker.
- The **other slice** (letter, frequency, mode, its own meter), or in its place the CW decode or the extremes readout (section 11).
- The **audio meter**: 32 bars of what you hear, red while transmitting.
- **PWR, SWR, MIC, COMP** transmit meters.
- The **knob tiles** A to D: what each knob does now, and its value. A lit tile means that function is on.



CW: the tiles change to RIT, filter, speed and APF, and the audio meter centers on your pitch.



Transmitting: TX lit, the audio meter shows the transmit audio, power and SWR read.



One slice: the SmartMTR extremes readout below the S-meter.

4. Knobs, buttons and the dial

The large knob at the bottom is the **VFO dial** (knob E). Knobs A to D are at the corners; buttons 1 to 4 are beside the dial. What the knobs do depends on the mode of the slice the face shows.

In voice and other modes

Control	Turn	Push	Push and hold
VFO dial	Tune (by the step, or by a tapped digit)	Next tuning step	Tune lock on/off
Knob A	AF level	Mute	
Knob B	Filter width	Free: assign it a setting	
Knob C	AGC threshold	Next AGC mode	
Knob D	NR level	NR on/off	
Button 1		Tune: Tuner Genius XL or the radio's ATU	Next mode
Button 2		PTT (also the Key/PTT jack's ring)	
Button 3		Next band	Previous band
Button 4		NB on/off	ANF on/off

In CW

Control	Turn	Push	Other
VFO dial	Tune	Autotune once (green flash behind the frequency)	Hold: autotune loop on/off (green background while on)
Knob A	RIT offset (or XIT)	RIT (or XIT) on/off	Double push: offset to 0. Hold: switch between RIT and XIT
Knob B	Filter width, centered on your pitch	Free	
Knob C	Keyer speed, also the radio's CW speed	Break-in on/off (lights the tile)	
Knob D	APF level	APF on/off	
Button 2		Straight key (also the Key/PTT jack's ring)	

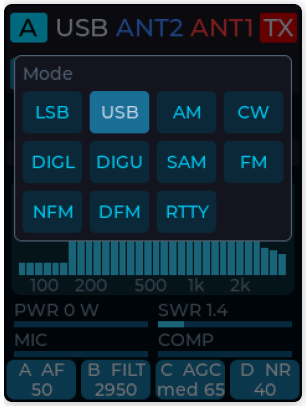
Knob B's filter width follows each mode's rule: in SSB the low cut stays where it is and the width grows from it; CW and AM stay centered; digital modes stay centered on their offset. 50 Hz per detent.

> A single push of knob A in CW waits about 0.3 s before acting, to tell it from a double push. RIT and XIT are never both on: switching from one to the other turns the old one off.

5. Touch: badges, pickers and digits

Touch	What it does
Mode (e.g. **USB**)	Opens the mode picker.
Antennas	Opens two rows: RX antenna and TX antenna, each set on its own.
TX badge	On the TX slice: turns TX off for it. On another of your slices: moves TX there.
Band badge	Opens the band picker.
First frequency digit	Opens the band picker.
Any other digit	The dial tunes by that digit's value (shown in gold). Tap it again to go back to the slice's step.
The other slice's row	Makes that slice the active one; the face follows it.
A knob tile	Opens the list to give that knob a setting (section 7).

In a picker the current choice is highlighted. Tap a choice to set it, or tap outside to close.



Mode picker.

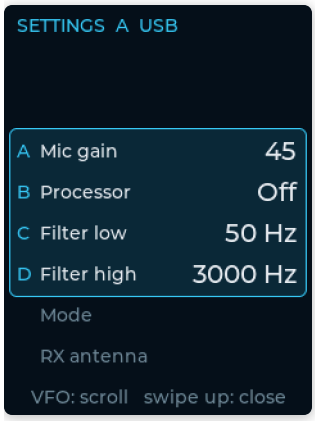


Antennas: RX and TX rows.



The 100 Hz digit tapped: the dial now tunes in 100 Hz steps, keeping the digits below.

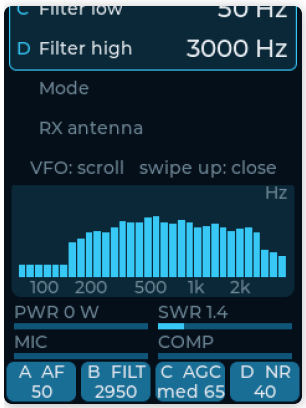
6. The settings list



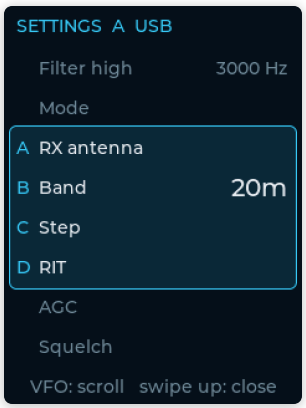
Swipe down from the top of the screen (start within the top half inch) to pull the list down like a page. In CW the same swipe brings down the CW keyer panel instead (section 9). **Push the page back up**, or push the VFO dial, to close it.

A window in the middle frames four rows, **A to D**. The **VFO dial scrolls** the list through the window; **each knob works the row beside it**, and changes take effect at once:

- a level: turn to adjust;
- a switch: turn right for on, left for off, or push to toggle;
- a level with a switch (DEXP, VOX, monitor, RIT): turn for the level, push for on/off;
- a choice (mode, band, step, antennas, processor): turn to step through it.



Halfway through a pull: the page follows your finger.



Scrolled five rows: four new settings in the window.

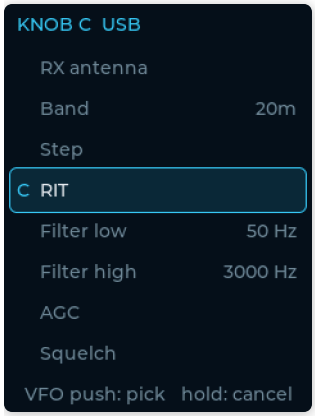
> Turning a knob on **Band** or **Mode** changes band or mode on every detent. **Tune** and **Two-tone tune** transmit a carrier as soon as they are switched on.

What is in the list

Only the rows that apply to the slice's mode are shown. The list opens with mic gain, the speech processor and the filter edges in the window.

Group	Rows
Slice	Mode, RX antenna, Band, Step, RIT, Filter low and high, AGC, Squelch, NR mode and amount, ANF mode and amount, WNB, APF (CW), NB, AF gain, Mute, Pan, Diversity, DAX channel, Tune lock, TX antenna, XIT
Mode-specific	DIGU/DIGL offset (digital); RTTY mark and shift (RTTY); FM tone, CTCSS tone, Repeater (Down/Simplex/Up) and Rptr offset (FM); ESC, ESC phase and gain (diversity on); AM carrier (AM, SAM)
Transmit	RF power, Tune power, Mic gain, Processor (Off/NOR/DX/DX+), DEXP, VOX and VOX delay, TX filter low and high, Mic input, Mic boost, Mic +ACC, Mic bias, TX DAX, Monitor, Tune, Two-tone tune
CW	CW speed, CW pitch, Break-in, Break-in delay

7. Giving a knob your own setting



Tap a knob's tile at the bottom of the face. The list opens with one row highlighted, lettered for that knob. Move the highlight with the VFO dial, then pick it with a **VFO push**, that **knob's push**, or a **tap** on the row. **Hold the VFO** to cancel.

On the face, the knob then works that setting the same way the list does, its push included, and its tile shows the setting's name and value. **Default**, at the top of the list, gives the knob back its own function.

Assignments are kept separately for CW and for the other modes, and are saved across power cycles. If a setting does not apply in the current mode, the knob falls back to its own function there.



Knob C given RIT: its tile reads **C RIT**.

8. CW operation

Keying the radio

Plug a paddle into the paddle jack and/or a straight key into the Key/PTT jack. The keyer runs on the Max itself, so timing never waits for USB. It keys the radio when **all** of these hold:

- the TX slice belongs to your AetherSDR station and is in **CW**;
- the radio's **break-in** is on. Break-in is the arm switch: push knob C in CW to turn it on or off (the C tile lights while it is on), or use AetherSDR's switch.

Button 2 and the jack's ring are a straight key in CW. If keying is not allowed when you press (break-in off, wrong mode), nothing is sent, and a paddle or key held while that changes has to be released before it keys again. A paddle held longer than 10 seconds is treated as stuck and released.

> Test transmitting into a dummy load first, with any amplifier in standby.

Zero beat and decode

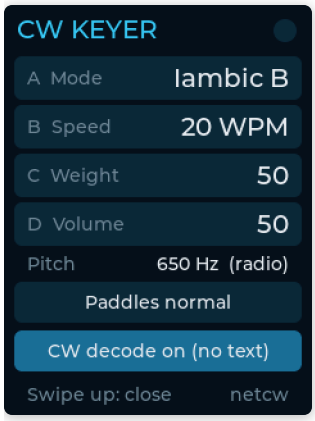
In CW the audio meter shows your filter's passband centered on the pitch, with a gold line at the pitch: a signal on the line is zero beat. Below the S-meter, AetherSDR's CW decode scrolls in from the right, colored by confidence (green, yellow, orange, red). Both need the AetherSDR changes in section 13. Push the VFO dial for autotune once, or hold it for autotune loop.

Sidetone

The Max makes its own sidetone through its speaker and line-out, following the radio's CW pitch.

> Known issue: the sidetone currently sounds harsh rather than a clean tone. Use the radio's or AetherSDR's sidetone if that bothers you.

9. The CW keyer panel



In CW (or with no radio connected, to set up and practice), **swipe down from the top** to bring down the keyer panel. Push it back up, or push the VFO dial, to close it.

Knob	Sets
A	Keyer mode: Off, Straight, Iambic A, Iambic B, Ultimatic, Bug
B	Speed, 5 to 60 wpm (also the radio's CW speed)
C	Weight, 25 to 75 (50 is standard)
D	Sidetone volume, 0 to 100

- **Paddles normal / reversed**: tap to swap dit and dah.
- **CW decode on/off**: whether the decode line is shown. "(no text)" means AetherSDR is not sending any.
- Bottom right: how keying reaches the radio (**TCP**, or **netcw**).

The pitch follows the radio's CW pitch. Settings are saved on the Max.

Weight

Weight moves time between each element and the gap after it without changing the speed. At 50 a dit is one unit on and one off; above 50 elements are longer and gaps shorter (a heavier sound, which also makes up for keying delays in an amplifier or QSK); below 50, the reverse.

10. SSB and voice operation

- **Compression is the speech processor** (Processor: Off, NOR, DX, DX+). **DEXP** is the downward expander, a noise gate for the mic.
- **Monitor**, **VOX** and **DEXP** are a level plus a switch: turn for the level, push for on/off.
- **TX filter low and high** set the transmitted bandwidth; **Filter low and high** the receive passband.
- **PTT** is button 2 or the Key/PTT jack's ring. It needs a TX slice of your station, and it will not start while another client is transmitting.
- **Two-tone tune** transmits the two-tone test signal for adjusting your audio and amplifier.

11. The audio meter and the readouts

- **Audio meter**: 32 bars of what you hear (after AetherSDR's receive processing), or of your transmit audio while transmitting. In CW it becomes the zero-beat view (section 8).
- **Extremes readout**: below the S-meter, the minimum and maximum signal over the last 3 seconds, as on AetherSDR's SmartMTR VFO flag: white markers on the S-meter, and the S unit over the

dBm beside each. It fades out for a steady signal (under 3 dB of change) or near the noise floor.

- **Where it shows**: that space holds the CW decode in CW when decode is on; otherwise the extremes readout, in CW or whenever you have one slice. With two slices outside CW, it shows the other slice.

12. Tuning, ATU and the Tuner Genius XL

- **Step**: push the VFO dial (outside CW) for the next step, or pick it from the list's Step row.
- **Digit tuning**: tap a frequency digit to tune by it (section 5).
- **Tune lock**: hold the VFO dial (outside CW). The frequency turns red.
- **Button 1** starts a tune. With a Tuner Genius XL in **operate**, it runs the TGXL's autotune; with the TGXL in standby or bypass, or none, the radio's internal ATU. Push again while the TGXL tunes to stop it.

13. Requirements and version notes

Feature	Needs
Everything except the two below	AetherSDR with self-named CTR2 devices (merged 2026-10-12; newer than v26.10.2)
CW zero-beat audio meter	AetherSDR with link extension AUDIO_SPECTRUM_LINEAR (PR #6431, merged 2026-10-13)
CW decode line	AetherSDR with link extension CW_TEXT (PR #6431, merged 2026-10-13), with AetherSDR's CW panel open

#6431 is on AetherSDR's main branch; until it is in a release, use a build of main for these two. The Max works with older AetherSDR builds; those features are simply not shown.

Getting the AetherSDR test build

AetherSDR's CI built main at the merge itself (47aa4098d, 2026-10-13). Open the run for your platform, scroll to **Artifacts** at the bottom, and download the one named below. GitHub requires you to be signed in to download artifacts; each download is a .zip holding the installer.

Platform	Artifact	Run
Linux	AppImage-x86_64 or AppImage-aarch64	github.com/aethersdr/AetherSDR/actions/runs/38369678506
Windows	Windows-Installer (the other artifacts are not needed)	github.com/aethersdr/AetherSDR/actions/runs/38369688472
macOS Apple Silicon	macOS-DMG-apple-silicon	github.com/aethersdr/AetherSDR/actions/runs/38369683864
macOS Intel	no build in this run	

These artifacts expire on 2027-01-11; by then the extensions should be in a regular release.

Keying goes to the radio over TCP. The radio refuses its faster netcw stream to a controller like the Max, which is fine for normal speeds.

14. Troubleshooting

Symptom	Check
Stays on "Waiting for AetherSDR"	The CTR2 Proxy applet is in USB mode, AetherCTR2 is picked and started, and AetherSDR is connected to the radio.
Paddles key the sidetone but not the radio	The TX slice is in CW and is yours, and break-in is on (push C; the C tile lit).
A list row shows no value	The radio has not reported that setting yet, or it does not apply to this radio (ESC needs diversity).
No zero-beat line or decode	AetherSDR needs the changes in section 13; decode also needs AetherSDR's CW decoder running and decode on in the keyer panel.
Button 1 does nothing on a TGXL	TGXL firmware 4.2 broke tuning relayed through the radio for some users; AetherSDR's direct TGXL connection is not available to the Max.
A knob does something unexpected	It may have been given a setting: its tile shows which. Tap the tile and pick Default.

Appendix: installing and updating the firmware

AetherCTR2 is installed beside Lynovation's firmware, not over it. Lynovation's firmware stays in the first app slot (`app0`) with its settings and registration; AetherCTR2 uses two slots of its own (`app1` and `app2`) and a settings partition of its own. A small boot selector (`otadata`) decides which firmware starts.

1. Put the Max in download mode. On a development build, run `tools/enter-bootloader.py` from the AetherCTR2 source tree; otherwise hold BOOT (through the hole in the back) and tap RESET beside the USB-C connector.
2. It appears as a **USB JTAG/serial debug unit** with its serial number in the device name. Check it is the Max before writing.
3. The first install writes the bootloader, the partition table, AetherCTR2 into `app1` and a boot selector that starts it (`tools/flash-max.sh`). Never erase the flash, and never write AetherCTR2 to `app0`: that is where Lynovation's firmware lives.
4. After that, AetherSDR updates AetherCTR2 over the USB cable (CTR2 Proxy applet, **Firmware** row; not yet in an AetherSDR release, and `tools/fw-update.py` does the same from the source tree). The new image goes into whichever of `app1`/`app2` is not running and is checked against its signature. It is kept only once it has connected to AetherSDR; until then, a reset returns the Max to the previous version.

5. To start Lynovation's firmware, tap **CTR2-Max firmware** on the status screen twice within four seconds. Returning to AetherCTR2 rewrites only the boot selector: `tools/otadata.py aetherctr2` in download mode, until Lynovation's firmware offers it in its menu.

AetherCTR2's face and meters are based on **SmartMTR** by **Dariusz Socha, SQ5X**, https://smartmtr.net/, used with his permission. The CTR2-Max-5 hardware and its stock firmware are by Lynovation.