Time Code Tool

Quick reference — reading, converting & generating timecode

A menu-bar timecode utility. **Read** LTC from any audio input or MTC from any MIDI port, and regenerate it as clean LTC audio and/or MTC — converting the frame rate on the way if you need to. Or run it as a **free-run master**, generating LTC and MTC together from one clock with no input at all. It also renders LTC audio files offline. Console-agnostic — useful to anyone who touches timecode.

Menu bar icon

Click	Show / hide the panel.
Right-click	Toggle the live timecode readout, or quit the app (no Dock icon or app menu — this is the only way to quit).
Icon colour	Grey = stopped or no signal. Green = locked, or generating. Amber = freewheel. Red = input clipping (overrides the others — fix the level first).
Live timecode text	Optional timecode in the menu bar, tinted by state (off by default; on a small display it can crowd the bar).

Choosing a mode

The switch at the top of the panel. Blue = reading, green = generating.

Read Timecode	Follow an incoming source: read LTC or MTC, show it, and regenerate it to the outputs.
Generate Timecode	Be the source: free-run as a timecode master from a start time and frame rate, with no input needed.
Start / Stop	Runs the mode you have selected. The mode and input source are locked while running — stop first to change them.

Input (Read Timecode)

LTC (audio)	Choose the interface carrying LTC and which channel to decode (mono). LTC often arrives on one channel of a multichannel device.
MTC (MIDI)	Read MIDI Time Code instead — from the app's own "TCT MTC In" port (select it as an MTC output in your DAW) or from any MIDI interface.
Readout	Large HH:MM:SS:FF; updates every frame while locked. Drop-frame shown with a ';'.
Frame rate	From LTC, auto-detected: 23.976 / 24 / 25 / 29.97 DF / 29.97 ND / 30. Shows "measuring…" briefly — telling 23.976 from 24 (and 29.97 ND from 30) needs a couple of seconds of signal.
Level meter	LTC input only. Distinguishes 'no LTC' from 'no cable', and warns when the feed is too hot — clipping distorts LTC and makes the readout jump.
Discontinuities	Counts frames that aren't previous + 1 (drop-frame aware), and says which kind: **Missed** (the signal path dropped frames; the timecode did not move), **Misread ignored** (an isolated frame with a bit error — the app drops it and carries on, no relatch, no MTC locate), **Jumps** (a real relocate, confirmed by the frame after it). Reset from the panel. If a source shows misreads, the app rides them out, but a clean source is still the fix — check a recording before the show with `tools/ltc_file_check.py`.

Important — MTC cannot distinguish 23.976 from 24, or 29.97 ND from 30. Timecode read over MTC therefore arrives labelled 24 or 30. Read it over LTC when that distinction matters.

Output

Both outputs are independent: either, both or neither. Each switch reveals its own settings.

Regenerate as MTC	Sends the timecode as MTC. Pick the destination: the app's own "TCT MTC Out" virtual port, or any MIDI interface, IAC bus or network session.
MTC rate	"Match input", or convert to a fixed rate on the way out. The panel shows the rate actually being transmitted.
Reading MTC?	MTC arrives with no sample clock of its own, so MTC output needs **Regenerate as LTC** switched on too — its audio device supplies the clock. The panel says so if it is missing.
Regenerate as LTC	Sends clean, freshly generated LTC audio to an output device — useful to reshape a ragged feed as well as to convert it.
Output device / Level	Any audio output, including the optional "Time Code Tool LTC" virtual device. Level in dBFS (default −18, ≈ 0 dBu at common calibration; range −24 to −3).
LTC rate	"Match input", or convert to a fixed rate — the same choice as MTC rate.
Freewheel	How many frames to keep generating through an input dropout before falling silent (default 5). Applies to both outputs.
Rescan devices	Re-scans audio and MIDI, so gear connected after launch appears without restarting.

Important — MTC cannot represent 23.976 or 29.97 ND. Those are sent under the nearest whole rate (24 and 30) and the panel shows a badge saying so. Every other rate is transmitted exactly, and LTC output has no such limit.

Generate Timecode (free-run master)

Start / Rate	The timecode to start from and the frame rate to generate (default 01:00:00:00 at 25).
Send MTC / Send LTC	Turn on either or both. Both run from one clock, so they stay locked to each other.
The clock	The audio output device provides the sample clock that paces both outputs. With Send LTC off, that device still runs as the clock — it just sends no LTC — so it stays selectable for MTC-only use.
Frame rate	Set once in the transport, so the per-output rate menus don't apply here.

Turn on at least one output — the app will not start with both switched off.

Virtual ports & devices

TCT MTC Out	A MIDI source the app creates. It stays available the whole time the app is running, so a DAW's routing survives stopping and restarting the transport.
TCT MTC In	A MIDI destination the app creates for reading MTC — select it as an MTC output in the sending application.
Time Code Tool LTC	An optional virtual audio device, offered as a tick-box when you run the installer — run it again later if you skipped it. With the device, other apps take this LTC as an audio input directly: no cabling, no spare interface channels. Needs macOS 13 or later; removal instructions are in Uninstalling.txt on the disk image.

Generate LTC File (offline)

Start / End / Duration	Start timecode, plus either an End timecode or a Duration in minutes (End takes precedence when set).
Rate / Sample rate / Bit depth	23.976–30 fps; 44.1 / 48 / 96 kHz; 16 / 24-bit. Mono.
Level	Output level in dBFS (default −18).
Render WAV	Offline render to a WAV you choose; the projected file size is shown first. WAV only — lossy formats smear the timecode square wave.

Running it

First launch	macOS prompts for microphone access the first time you read an audio input — allow it, or the app receives only silence (LTC on an input counts as ‘microphone’ to macOS).
Start at login	Optional, at the foot of the panel.
Decoder	LTC decoding uses libltc, bundled with the app.

If something goes wrong

Export log	Click **Export log** in the panel footer. A copy of this session’s log lands on your Desktop as a text file — send that with your report. Every launch is logged automatically to **~/Library/Logs/dylanmaudio/Time Code Tool/**; nothing to switch on.
Send feedback	Also in the footer — a short message straight to me, no email client needed.

What's new in v1.1.5

• Tidier popover: the Bitfocus Companion settings and the on-screen hints now sit behind disclosures, closed by default (a green dot on the Companion header shows when Companion is connected); the trademark notice at the bottom has gone — it's in the Quick Reference and the licence. Companion can now also bring the app's window forward.
• New: this app can now be controlled from Bitfocus Companion — Start/Stop, Read/Generate, the input source, MTC and LTC output, Reset counters and the generator's start time and rate all show up as Companion buttons, with the live timecode, lock state and frame rate reflected back on them (a multi-button timecode readout is one preset away). On by default; two new switches in the app let you turn Companion control off or lock the show-critical ones (Stop, and switching an output off). Mode and input source can't be changed from Companion while running — stop first, as in the app.
• The generator's start time (and the WAV generator's start and end) is now checked against the frame rate as you type it. A time that doesn't exist — 24:00:00:00, frame 25 at 25 fps, frame 00 at the top of a drop-frame minute — is refused with the legal range shown, instead of being stored and going wrong at Start.