



PRODUCT DOCUMENTATION

PicoRecorder

User Manual

The complete reference for setup, operation, and care of your PicoRecorder.

Contents

■ 1	Overview	5
■ 2	The PicoMic	6
■ 3	First steps	7
■ 4	Main screen	8
■ 5	Wireless microphones (1-6)	9
■ 5.1	Pairing your PicoMic	9
■ 5.2	Connect / disconnect	9
■ 5.3	In dock	9
■ 5.4	Mute	10
■ 5.5	Solo	10
■ 5.6	Mic settings	11
■ 6	Audio inputs (A-B)	12
■ 7	Audio outputs	13
■ 7.1	Analog output	13
■ 7.2	USB audio	13
■ 8	Understanding the audio path	15
■ 9	Recording	17
■ 9.1	Files split	17
■ 9.2	Multitrack file tracks behavior	18
■ 9.3	SD card format	18
■ 10	Timecode	19
■ 10.1	How it works	19
■ 10.2	What timecode is	19
■ 10.3	Timecode settings	20
■ 10.4	Timecode colors	21
■ 10.5	How to use	22
■ 11	Settings menus	23
■ 11.1	General settings	23
■ 11.2	Audio routing	23
■ 11.3	Audio output settings	24
■ 11.4	Record / SD card	25
■ 11.5	Timecode	25
■ 12	Other features	26
■ 12.1	Screen lock	26
■ 12.2	Automatic shut-down	26
■ 12.3	Suggest your idea!	26

1 Overview

Thank you for purchasing a PicoRecorder! We hope this product will bring you satisfaction for many years to come.

Take some time to read this user manual since the PicoRecorder is more complex than basic wireless audio receivers, and you may miss some features.

The PicoRecorder software is regularly updated with improvements and new features, and this manual covers the latest firmware version.



The PicoRecorder is composed of the following:

- **Two mono 32-bit audio inputs (A and B)**, including a plug-in power supply for external microphones.
They can be used as LINE-IN or MIC-IN, thanks to a very large dynamic range.
- **A headphones output** for monitoring the L + R mix output.
- **A high-quality TRS stereo output**, that is typically used to connect to a camera mic input.
The analog output gain (i.e. the signal level reached when the signal is at 100% of the VU-meter) is settable in the settings menu.
- **A full-size SD port**. SDHC and SDXC cards are supported.
- **A USB-C connector**, used for charging, USB audio output, and pairing / updating the PicoMics.
Using a USB-C to USB-C cable, the PicoRecorder can be connected to smartphones or computers, allowing a full-digital setup.
- **A matte touch screen with haptic feedback**, that can be locked using the "dot" key.

2 The PicoMic

The PicoMic is a high-end, ultra-compact wireless microphone that integrates a digital microphone capsule with a wide dynamic range, and also offers the option to connect an external microphone.

It is provided with many accessories, including color windshields, color clip covers and a USB-C to microphone cable.



The PicoRecorder is compatible with the PicoMic 3 and PicoMic 3 Pro.

If you already have some, it means that you don't have to purchase new PicoMics to use the PicoRecorder!

3 First steps

Before you can use your PicoMic wireless system, make sure to:

- **Charge your receiver and microphones.** Put the microphones in the travel case, and connect the case to USB-C (on the back).
Connect the receiver to any USB power source (wall charger, computer, power bank) and wait about 2 hours for the charge to finish completely, both the receiver and the microphones.
- **Update your firmware.** The PicoMic system is an advanced system that can greatly evolve with the free updates that we regularly release.
We recommend following the procedure [here](#) to update your firmware and benefit from the latest features and bug fixes.
- Pair your PicoMics by connecting them to the PicoRecorder with the provided USB-C cable.

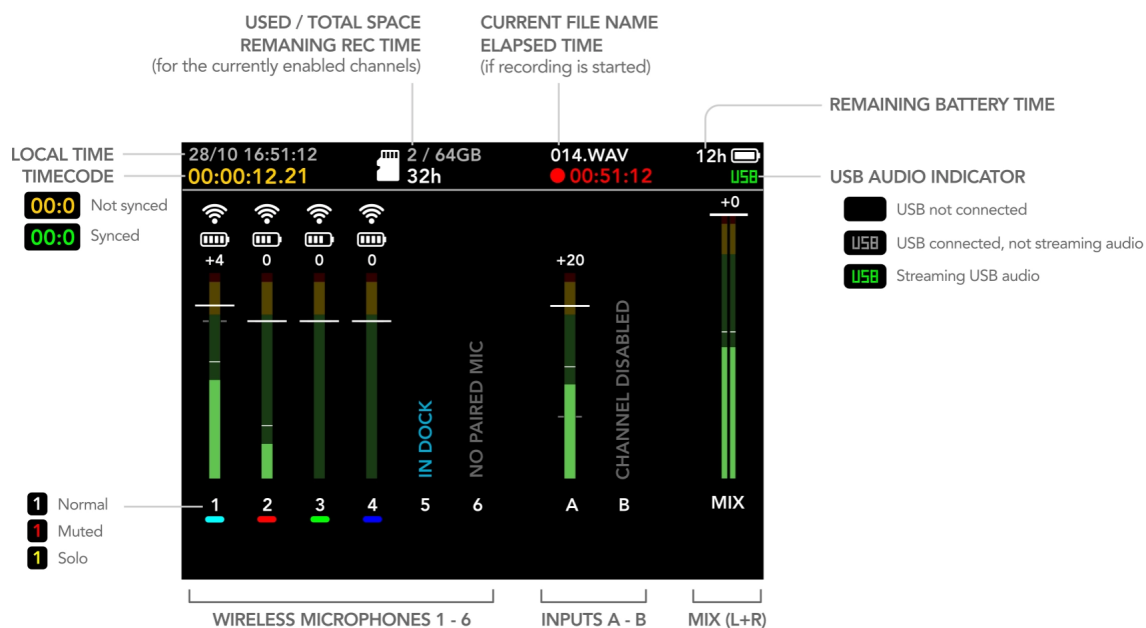
Please note that the **POWER** button is not touch-sensitive; it must be pressed firmly to start or stop the recorder:



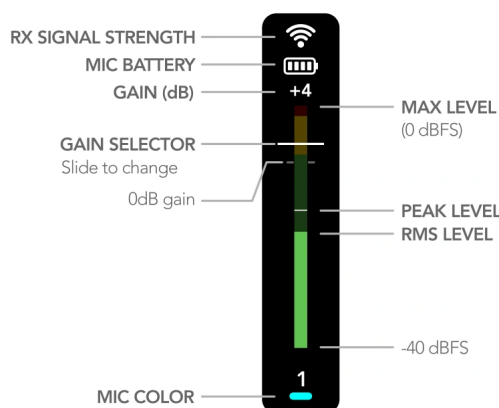
4 Main screen

On the main screen, you will find all your microphones (up to 6), the two audio inputs, and the stereo output.

Tap a microphone or channel to show a small toolbar with more options.



Each microphone shows a **smart VU-meter** that is used both to monitor the mic audio and set the gain that is used to make up the output:

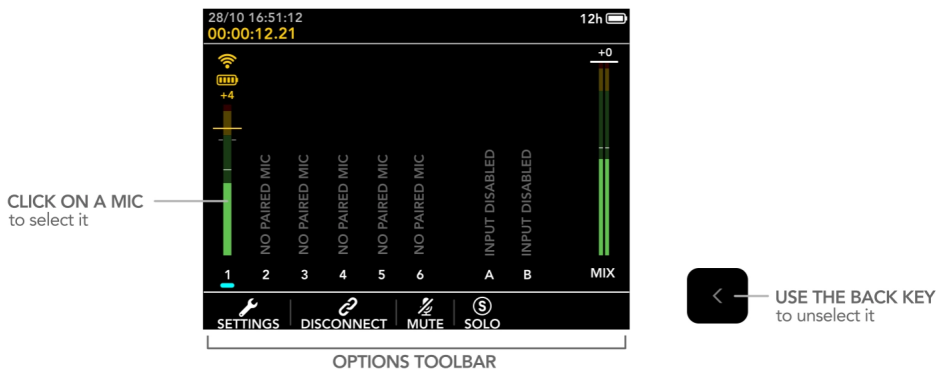


Unlike some recorders, each gain selector is hardwired to its own channel. You can't reassign a selector to another channel or use it to control a selectable mix of channels.

5

Wireless microphones (1-6)

Tap a mic to select it (it will become orange), and display a toolbar with more options:



5.1 Pairing your PicoMic

Pairing links your PicoMic to the receiver, so the PicoRecorder can automatically connect to it when powered on.

Just connect your PicoMic to the PicoRecorder using the provided USB-C – USB-C cable.

If needed, the PicoRecorder will update your mic firmware, and the pairing is automatic.

My PicoMic won't pair! What should I do?

1. Make sure your PicoMic is charged. A PicoMic with an empty battery is a frequent cause of pairing failure.
2. Make sure to use the provided USB-C cable. Other USB-C cables may still work, but with some of them the connection will only work when the USB-C plug is inserted in one specific orientation on the PicoMic side (you may need to flip the connector).
3. Make sure you are on the main screen — pairing will not start if you are on another page, like the settings menu, etc.

5.2 Connect / disconnect

Your PicoRecorder will automatically connect to your PicoMic when started.

However, you can disconnect it by tapping the corresponding toolbar button ("**DISCONNECT**"), and reconnect it by tapping "**CONNECT**".

5.3 In dock

When the PicoMic is placed in the travel case USB-C dock, it will appear as "IN DOCK" and will effectively be disconnected from the PicoRecorder (and won't consume any power).

Just remove it from the USB-C dock and it will reconnect automatically.

5.4 Mute

Each mic can be muted on the left or right of the PicoRecorder audio output.
When a mic is muted on both L and R output, the VU-meter becomes grayed as a visual indication:



⚠ Muting is not the same as disconnecting the mic.
In this mode, you can still see the audio levels, and the mic is still recorded to the SD card in the RAW multitrack file.
However, the mic audio won't be on the output at all. These mics are also given lower priority over the radio link, so they will be the first to drop out in strong interference situations.

This mode allows you to completely mute a microphone on the output, while still recording it for potential use later.

If your mic is...	Muted	Disconnected
Radio link	Audio is streamed normally to the PicoRecorder	No radio connection
Mic power	Power consumption is the same as a connected mic	Powered off – can last several months
SD card recording	Mic is recorded in the RAW multitrack file	The mic is not recorded
Audio output	Mic audio is not on the output	Mic audio is not on the output

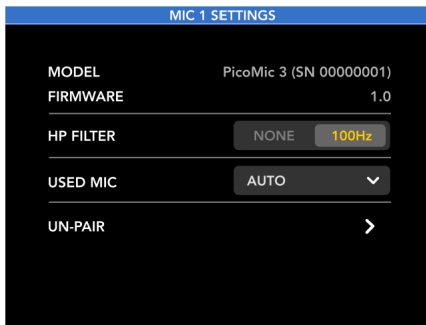
5.5 Solo

If you need to quickly limit the audio to a single microphone or input, use SOLO mode. It will mute all other channels and restore the original configuration when you exit SOLO mode.

Like Mute, Solo only affects what you hear on the output mix: it never changes the RAW multitrack recording, and it is independent of the Audio routing assignment.

5.6 Mic settings

To access the settings menu, select the mic on the main screen, and tap the “SETTINGS” button in the bottom toolbar.



Model / firmware: the model and firmware version of the PicoMic.

Currently, the PicoMic 3 and PicoMic 3 Pro are supported.

HP filter: enable / disable a 100 Hz high-pass filter.

This filter helps reduce the typical hum from AC systems, fans, and similar sources. We generally recommend leaving it enabled.

Used mic: allows you to force the use of the PicoMic internal mic, or an external one (connected with the provided USB-C adapter).

In AUTO mode, the PicoMic will detect the small power consumption of any mic connected on its USB-C adapter, and automatically use it instead of the internal microphone.

Un-pair: allows you to unpair your PicoMic from the receiver (i.e. if you don't plan to use a PicoMic for a long time, etc.). PicoMic firmware updates are applied automatically as soon as you connect the PicoMic to the receiver via USB-C.

6

Audio inputs (A-B)

The PicoRecorder provides two 32-bit audio inputs (A and B), each one being a mono input. Both inputs provide plug-in power (3.0 V), so they can power small lavalier microphones, or mics that need a voltage bias to work.

Once again, tap an input to show a toolbar with options:

- **Mute / Solo:** same behavior as for the PicoMics.
- **Enable / disable:** the audio inputs are disabled by default.
Enabling any will increase the power consumption of the PicoRecorder significantly, so we recommend leaving them fully disabled if you don't use them. Also, it will prevent having any analog noise if nothing is connected to the input.

7

Audio outputs

7.1 Analog output

This is the main analog output: a high-quality stereo output with a locking TRS connector.



We strongly recommend using the supplied locking TRS-to-TRS cable.

We do not recommend using TRRS cables (i.e., connectors with four contacts instead of three), as they can cause unstable audio.

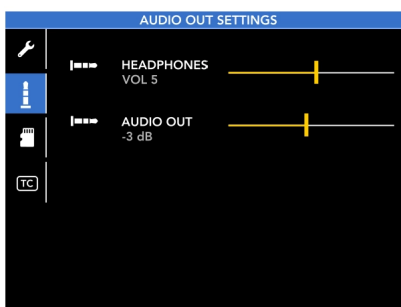
It can be used to connect to:

- **Cameras / DSLRs external mic inputs;**
- PA systems inputs;
- Audio recorders inputs.

It supports a wide dynamic range and can produce both low-level (mic-level) and high-level (line-level) signals.

The maximum output level (in dB) is set in the Settings menu / Audio output:

- 0 dB is a typical "mic level" signal; it is typically the value to use when the audio output is connected to a camera mic input.
- Lower values can be used with sensitive mic inputs.
- +24 dB is a typical "line out" level.



7.2 USB audio

The PicoRecorder can be connected to other devices in USB-C and will act as a USB stereo microphone (outputting the L / R OUT signal).

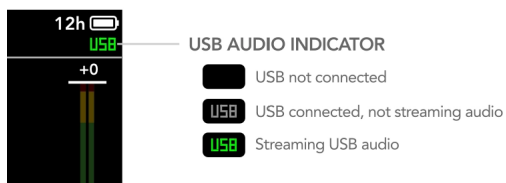
A big advantage is that the signal path is entirely digital, so you won't have any analog noise issues.

This works with virtually all USB-C host devices, such as:

- Smartphones (iPhones with USB-C, Android)
- Tablets
- PCs (Windows / Linux)
- Macs
- And even some action cameras that can accept USB-C microphones on their USB connector.

You have a small indicator that shows the streaming status.

Audio streaming is initiated by the smartphone/tablet (typically when an app enables the microphone), and the PicoRecorder indicates whether audio is actually streaming over USB or not.

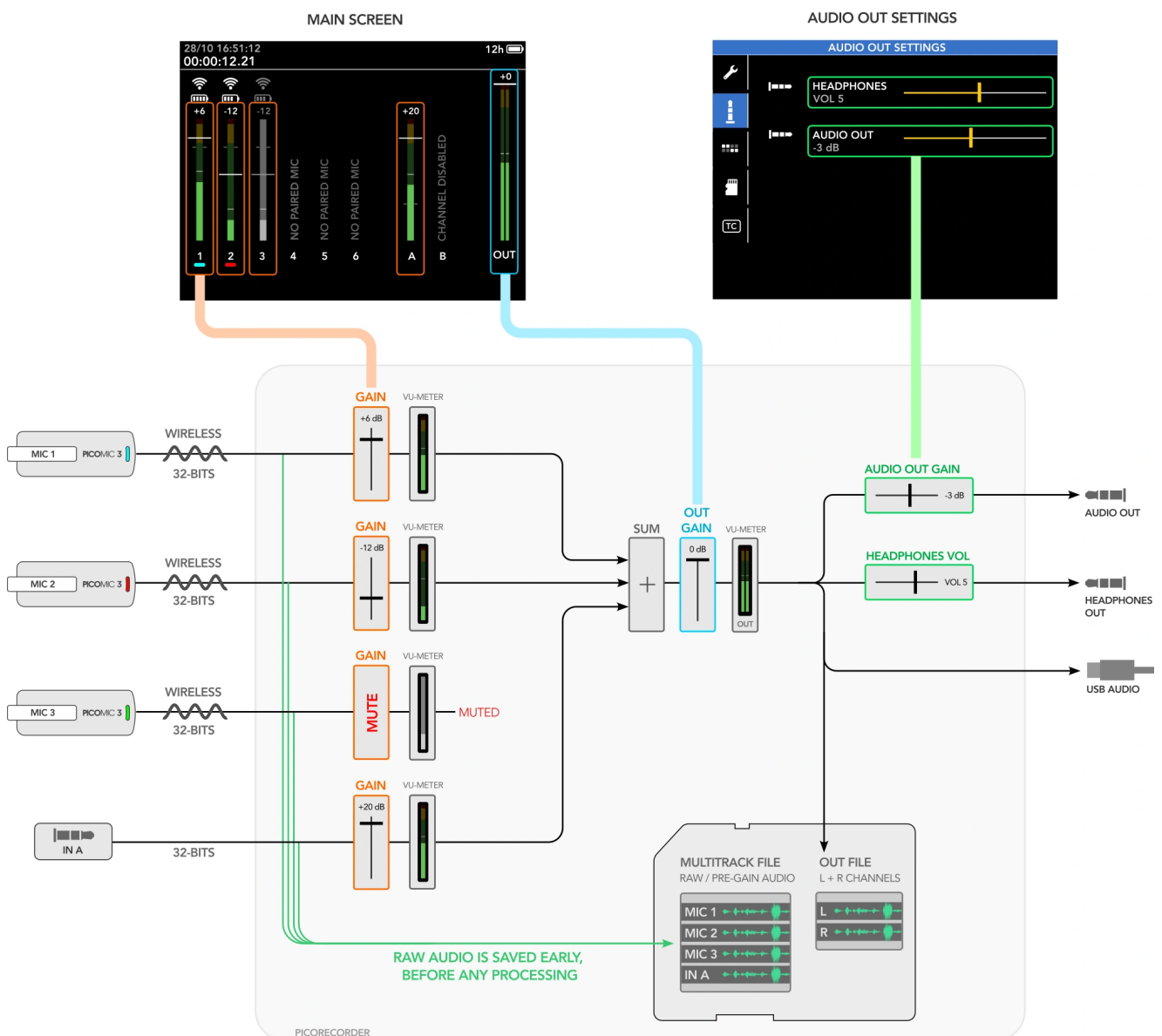


8

Understanding the audio path

The PicoRecorder offers several different gains, and it's important to know what each one does. Check the diagram below, showing a setup where we have:

- **MIC1 and MIC2** connected;
- **MIC3** connected but muted;
- **IN A** enabled.



The audio path is explained as follows:

- **Each PicoMic samples the audio in 32-bit directly**, and sends it in 32-bit to the receiver, thanks to PicoGear X³² wireless technology.
No gain is applied in the PicoMic and you get the raw signal at the input of the PicoRecorder.

- If the recording is enabled, each microphone / audio input is saved in 32-bit float into a multitrack file.
- **Each microphone / input has its own gain (in orange)** used to make the sum of all the non-muted channels into the OUT stereo track.
- **This OUT channel also has an optional gain** (in blue). The resulting stereo audio goes:
 - Into an OUT file on the SD card (if recording is enabled)
 - To the AUDIO OUT (with an amplification setting, shown in green, that should match your camera input level)
 - To the HEADPHONES output (with a selectable volume)
 - To the USB audio output

The different gains uses are explained below:

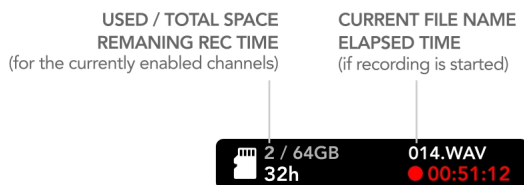
- **Microphones / input gains** (in orange).
Use these to match the recording situation. If multiple people are mic'd, adjust each channel so everyone comes through at a similar level.
- **Output gain** (in blue).
Normally leave this at 0 dB (unity) so the mix isn't altered. Only lower it when you need overall level moves, like a fade-out.
 **Important:** don't use this gain to "match" the PicoRecorder to your camera's input level. Use **Audio Out gain (green)** for that.
- **Audio out gain** (in green).
This sets how loud the PicoRecorder's OUT stereo track is when it leaves the AUDIO OUT jack and goes into your camera.
Set it so the two devices "agree" on level: when the PicoRecorder's OUT track is hitting 100%, your camera's audio meter should also be close to 100% (without clipping). Once it's matched to your camera's input, you usually don't need to change it again.

9 Recording

The PicoRecorder is able to record the audio on an SD card.

To start / stop the recording: tap the REC key on the touch screen.

Once recording is running, a new panel will appear on top of the screen:



- **Used / total space:** shows the currently used space and the total space of the main SD card partition. It may be smaller than the full SD capacity if there are multiple partitions.
- **Current file name:** shows the currently recording file number; it is incremented each time a recording is started.
- **Elapsed time:** the total recording time for this file.

Each time a recording is started, two files are created (with a new file number):

- **20260410_143620_001_OUT.WAV:** a stereo file of the PicoRecorder output (L + R channels), saved in 32-bit float.
- **20260410_143620_001_RAW.WAV:** a multitrack file containing all the currently enabled channels, in raw 32-bit float, without any gain applied.

The number of tracks depends on the currently connected mics / enabled inputs.

The files are organized into daily folders, such as /20260410/....

9.1 Files split

Due to limitations of the WAV file format, WAV files can't be bigger than 4 GB.

Once a WAV file reaches 4 GB, it will be closed and a new file will be created:

...001_RAW.WAV — [split] →001_RAW.1.WAV — [split] →001_RAW.2.WAV etc.

No audio sample is lost; the files can be concatenated together to recreate the audio as if no split had happened.

The same split behavior will happen when a PicoMic connects while the recording is already in progress (since a new track needs to be added).

9.2 Multitrack file tracks behavior

To avoid creating huge files with little to no audio, the multitrack RAW file only includes tracks for currently connected microphones and enabled inputs.

If a PicoMic becomes connected while the recording is in progress, there will be a split of the multitrack file to add the new track (since the number of tracks in a multitrack file can't be changed dynamically). If the microphone is disconnected, the track won't be removed, it will stay in the multitrack file but have no audio on it (this avoids creating many files if the connection is unstable, for example).

9.3 SD card format

Only modern SDHC or SDXC cards are supported. Old SDSC cards are not supported.

Both FAT32 and exFAT partitions are supported. We normally recommend using exFAT (most SD cards come already formatted in this format).

About SD cards performance: recording up to 10 channels in 32-bit float can result in fairly high throughput. Two factors need to be considered for performance:

- **The clusters size of the filesystem** (chosen when formatting the SD card). Very small clusters (<32 kB) will result in poor write performance (you will get a warning if you have such small clusters). The PicoRecorder uses 32 kB clusters when formatting your SD card.
- **Free space on the card.** The write performance of any SD card degrades once the card is nearly full. Even if you delete files, or format the card using standard PC/Mac tools, performance may not recover; we recommend formatting the card with the PicoRecorder regularly, as explained below.

High-performance SD erase / formatting: thanks to its low-level access to the SD card, the PicoRecorder can issue special commands that instruct the card to perform a full erase (almost no SD card reader can do that). This restores the card's original write performance, and you can use it on any of your SD cards; even ones you also use for video, etc.

If you have SD performance issues (you will get a warning if anything happens during a recording), we strongly recommend trying formatting your SD card with the PicoRecorder and check the results.

10

Timecode

The PicoRecorder includes an internal full-fledged timecode generator.

10.1 How it works

The point of timecode is to synchronize all your audio / video devices including timecode generators before your day of shooting.

Then having all the audio / video files timestamped with the same timecode allows to synchronize the audio(s) and video(s) timelines perfectly, without needing to synchronize them manually (which typically needs the famous "clapperboard").

This implies that all of your devices include a timecode generator and use it to timestamp all files — typically by storing the file's start timecode in the file metadata.

This is what the PicoRecorder does: each saved WAV file includes, in its metadata, the current timecode at the moment the file's first audio sample is recorded. You can see it when you import the file in software that supports timecode (e.g. DaVinci Resolve) where you can see the timecode being shown:

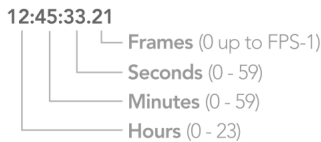
In theory, you should also use a camera that has a built-in timecode generator and puts the timecode in the video files metadata.

However, in many cases, you won't have such a camera and this is where LTC timecode comes to help. The PicoRecorder allows you to send the timecode in an audio way to the camera, so that the timecode is recorded on one track of your camera video files. Typically on your video files, you will have the L audio being the timecode, and the R audio being the PicoRecorder main audio output.

Using appropriate software able to decode that audio timecode (i.e. "LTC timecode"), you can now have video files with timecode — even though your camera doesn't support any timecode at all. Software like DaVinci Resolve supports that audio timecode in video files.

10.2 What timecode is

Timecode is just a way of expressing a number of video frames since a "zero time" reference. But instead of using large numbers to do so, it uses the H/M/S notation:

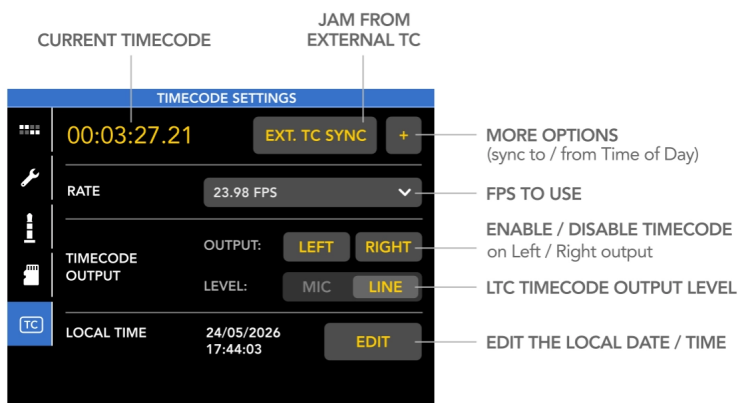


Let's say you are shooting at 25 FPS, the timecode last part will go from 00 to 24 in one second, then increment the number of seconds, etc. This leads to two important pitfalls:

- **Timecode is worthless if we don't know the FPS.** There is no way to convert between the timecode and the actual time if we don't say which FPS is used; this is why you need to tell all timecode generators and software which FPS you use. Normally, you use your main camera video rate.
- Depending on the FPS (check below), the Hours / Minutes / Seconds may not be the actual time of the day, and it can also slowly drift in time. Never assume that the HH:MM:SS in the timecode is an actual Hours / Minutes / Seconds, it is not the case.

10.3 Timecode settings

Go to the Settings / Timecode tab:



Ext. TC Sync: lets you sync the PicoRecorder's timecode to an external timecode source — useful when another device (a camera, a master timecode generator, or a second recorder) is acting as the timecode "master" for your shoot and you want everything to share the exact same time reference. To sync:

1. Tap **EXT. TC SYNC**. The screen will display "Connect external timecode to input B."
2. Connect your external timecode source to **IN B**.
3. The PicoRecorder detects the incoming signal and "jams" its internal generator to it — meaning it copies the master's timecode and aligns its own generator to match.
4. When the screen shows **"Timecode was synced successfully,"** you're done.

From that point, the PicoRecorder keeps running on its own ultra-stable TCXO clock, so you can disconnect the master and the two will stay in sync throughout your shooting day.

More options (+ button):

- **Time of Day to TC:** sets the PicoRecorder timecode to the current time of day.

- **TC to Time of Day:** sets the PicoRecorder clock to the current timecode value.

FPS to use: lets you select the timecode rate (use your camera video rate), among:

- 23.98 FPS (warning: the HH:MM:SS of the timecode will drift compared to the actual HH:MM:SS)
- 24 FPS
- 25 FPS
- 29.97 FPS (warning: the HH:MM:SS of the timecode will drift compared to the actual HH:MM:SS)
- 29.97DF FPS (same but the HH:MM:SS will not drift compared to the actual time)
- 30 FPS

LTC timecode output: you can select to output the audio / LTC timecode on either LEFT, RIGHT or both of the AUDIO OUT of the PicoRecorder.

Note that this will be the only place where the LTC timecode is output; it won't be output in the headphones output for example. This is the same control as the **TC** buttons in the Audio routing dialog. You can also select the level of that LTC timecode:

- **MIC:** will output an audio-level LTC, below the maximum level that is defined in the AUDIO OUT gain setting.
Used to feed your camera mic input.
- **LINE:** will output a digital signal at high level.
Used to feed other timecode generators and some cameras timecode inputs.

10.4 Timecode colors

The color of the timecode shown on the screen indicates whether the PicoRecorder has been synchronized recently with an external timecode source.

Timecode color	Meaning
	The PicoRecorder has not been externally synchronized within the last 10 hours.
	The PicoRecorder was successfully synchronized from an external source connected to IN B less than 10 hours ago.

After an external sync, the timecode remains green for **10 hours**. This gives you a quick visual confirmation that the recorder was jam-synced recently.

The 10-hour period is intended to cover a typical production day. Over time, the PicoRecorder's internal clock may drift slightly from the external reference, so a new sync is recommended before the next session or whenever precise timecode alignment is required.

After 10 hours, the timecode returns to yellow to indicate that the last external sync is no longer considered recent.

10.5 How to use

First, timecode is only useful if you record audio to the SD card and later need to synchronize that audio with your video files.

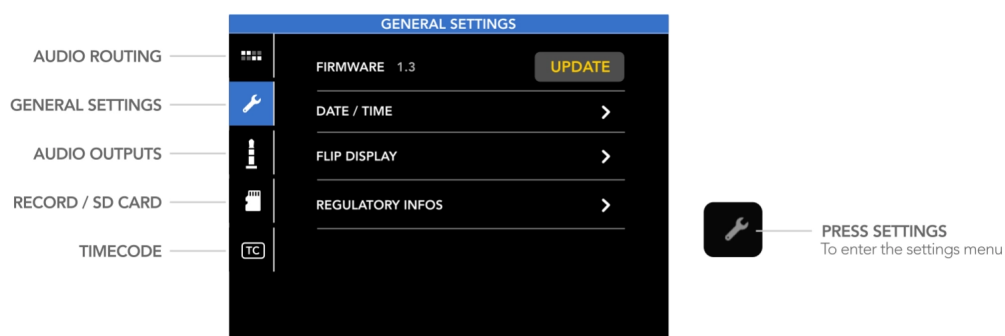
If you connect the PicoRecorder's audio output to your camera's mic input and use that as your main audio, timecode isn't necessary.

If you need to synchronize your audio and video files (basic setup):

- **Your audio files** in the PicoRecorder SD already have the timecode included in them (metadata). You have nothing special to do.
- If you are a beginner with timecode, it is likely that your camera doesn't support including timecode **in the video files, so you will need to output the audio / LTC timecode to your camera audio track:**
 1. Connect the AUDIO OUT of the PicoRecorder to your camera mic input;
 2. Go to the Settings / Timecode and enable the LEFT channel timecode output. You will have a strong digital-like signal on the left audio of all your video tracks, this is the LTC timecode, and the right audio in your video files will typically be the normal audio output of the PicoRecorder;
 3. Once you have your files in post, you need to use software that supports synchronizing the audio and video timelines with the timecode (not all of them support this).

11 Settings menus

Tap the "SETTINGS" key on the right of the screen to enter the settings menu:



11.1 General settings

Firmware: shows the PicoRecorder firmware version. Tap the UPDATE button to start the [Firmware update](#) procedure.

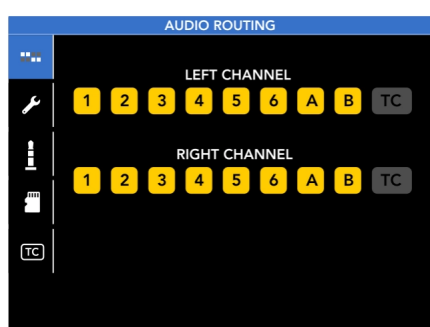
Date / time: opens a dialog allowing you to set the current date and time, that is used to mark your audio files date/time.

This time is retained as long as the PicoRecorder's main battery isn't completely drained. If you leave the PicoRecorder for several months with the battery nearly empty, the date and time may reset.

Flip display: rotate the screen by 180° to allow mounting the PicoRecorder upside-down.

Regulatory info: shows radio certification information.

11.2 Audio routing



The **Audio routing** dialog lets you choose, source by source, what goes to the left and right channels of your stereo mix.

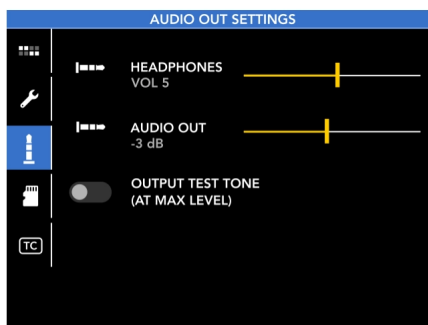
Each row is one side of the output. Tap a source to toggle it — orange means it is sent to that side, gray means it isn't. Light a source under both rows to keep it centered, or under one row only to pan it hard. The buttons are **1-6** (the PicoMics), **A / B** (the audio inputs), and **TC** (timecode).

Routing only shapes the stereo mix — the L + R OUT feeding the AUDIO OUT jack, headphones, USB audio, and the OUT.WAV recording. It does not affect the RAW multitrack file, which always records every source untouched, so nothing is ever lost.

⚠ Routing is separate from MUTE / SOLO: it decides where a source goes, not whether it is silenced. A source is heard on a side only if it is routed there **and** not muted there.

Timecode (TC) is a special case: it goes **only** to the AUDIO OUT jack — not the headphones, USB, or recording. This is the same control as the LTC output in **Settings / Timecode**, where you also set its level (MIC or LINE).

11.3 Audio output settings



Headphones: set the headphones output volume.

Audio out: set the amplification of the analog audio output.

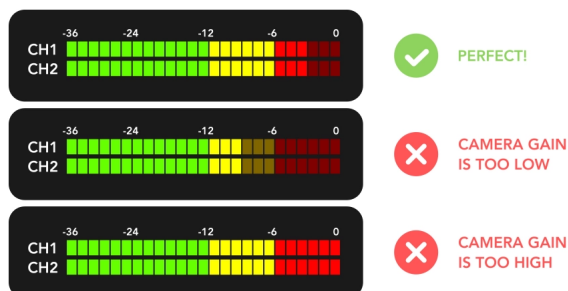
Changing this value won't affect the recorded level of the OUT channel, nor the USB audio output level.

Output test tone: when you connect the PicoRecorder to a camera, you need to set the camera's input level correctly. If it's set too low, your recordings will be quiet and hissy; if it's set too high, loud sounds will distort. The Output test tone makes this easy: it plays a steady tone at the loudest level the PicoRecorder will ever send out. You then simply turn up your camera's input until its level meter sits just below the top — and you know your camera can handle anything the PicoRecorder sends without distorting.

How to calibrate your camera:

1. Connect the PicoRecorder's analog output to your camera's microphone input.
2. On your camera, switch the audio input to **manual** (turn off any automatic level / AGC setting), and if your camera has an input **limiter**, turn it off for now (it can interfere with the calibration). Start with the camera's input level set fairly low.

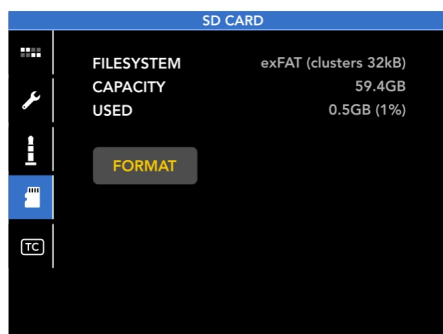
3. On the PicoRecorder, open the **Audio output settings** and turn on **Output test tone**. You'll now hear the tone and see it on your camera's level meter.
4. Slowly raise your camera's input level until the meter is **almost at the top** — close to the maximum, but not hitting the last red squares:



5. Turn **Output test tone** back off. Your camera is now calibrated.

That's it! Your camera input is now matched to the PicoRecorder. Loud moments won't distort, and quiet moments will stay clean and clear.

11.4 Record / SD card



This dialog will show you the SD card information, and allows you to format it with the PicoRecorder recommended settings.

11.5 Timecode

Check the timecode settings dialog.

12 Other features

12.1 Screen lock

The touch screen can be locked by tapping the "DOT" key on the right of the screen. When the screen is locked, an unlock slider will appear on the bottom of the screen:



12.2 Automatic shut-down

The PicoRecorder will automatically shut down after 5 minutes if no PicoMics are connected and no audio inputs are enabled.

In all other cases, it will not shut down automatically unless the battery becomes low.

12.3 Suggest your idea!

The PicoRecorder is made to be an evolutive receiver, and many features can be added through software updates.

At PicoGear, we take firmware updates very seriously; our previous systems have received updates for 5-6 years.

If you have any feature ideas or requests, please let us know!

We'll do our best to include them in future updates whenever possible.