

# User Manual

This document has the same layout as the manpage, but it contains links to other pages for more information: [\[FluidFeatures\]](#) and [\[FluidSettings\]](#).

## NAME

[\[FluidSynth\]](#) - a [\[SoundFont\]](#) synthesizer

## SYNOPSIS

fluidsynth [\[options\]](#) [\[ \\_SoundFonts \]](#) [\[ midifiles \]](#)

## DESCRIPTION

[\[FluidSynth\]](#) is a real-time MIDI synthesizer based on the [SoundFont](#) 2 specifications. It can be used to render MIDI input or MIDI files to audio. The MIDI events are read from a MIDI device. The sound is rendered in real-time to the sound output device. See [\[FluidFeatures\]](#) for a comprehensive list of features implemented and working.

The easiest way to start the synthesizer is to give it a [\[SoundFont\]](#) on the command line: 'fluidsynth soundfont.sf2'. fluidsynth will load the [\[SoundFont\]](#) and read MIDI events from the default MIDI device using the default MIDI driver. Once [\[FluidSynth\]](#) is running, it reads commands from stdin. There are commands to send MIDI events manually, to load or unload [SoundFonts](#), and so forth. All the available commands are discussed below.

[\[FluidSynth\]](#) can also be used to play a list of MIDI files. Simply run [\[FluidSynth\]](#) with the [\[SoundFont\]](#) and the list of MIDI files to play. In this case you might not want to open the MIDI device to read external events. Use the -n option to deactivate MIDI input. If you also want to deactivate the use of the shell, start [\[FluidSynth\]](#) with the -i option: 'fluidsynth -ni soundfont.sf2 midifile1.mid midifile2.mid'.

Run fluidsynth with the --help option to check for changes in the list of options.

## OPTIONS

fluidsynth accepts the following options:

-a, --audio-driver=[\[label\]](#)

The audio driver to use. "-a help" to list valid options

-C, --chorus

Turn the chorus on or off [\[0|1|yes|no, default = on\]](#)

-c, --audio-bufcount=[\[count\]](#)

Number of audio buffers

-d, --dump

Dump incoming and outgoing MIDI events to stdout

-E, --audio-file-endian

Audio file endian for fast rendering or aufile driver ("-E help" for list)

-f, --load-config

Load command configuration file (shell commands)

-F, --fast-render=[\[file\]](#)

Render MIDI file to raw audio data and store in [\[file\]](#)

-G, --audio-groups

Defines the number of LADSPA audio nodes

-g, --gain

Set the master gain [\[0 < gain < 10, default = 0.2\]](#)

-h, --help

Print out this help summary

-i, --no-shell

Don't read commands from the shell [\[default = yes\]](#)

-j, --connect-jack-outputs

Attempt to connect the jack outputs to the physical ports

-K, --midi-channels=[\[num\]](#)

The number of midi channels [\[default = 16\]](#)

-L, --audio-channels=[\[num\]](#)

The number of stereo audio channels [\[default = 1\]](#)

-l, --disable-lash

Don't connect to LASH server

-m, --midi-driver=[\[label\]](#)

The name of the midi driver to use [\[oss,alsa,alsa\\_seq,...\]](#)

-n, --no-midi-in

Don't create a midi driver to read MIDI input events [\[default = yes\]](#)

-O, --audio-file-format

Audio file format for fast rendering or aufile driver ("-O help" for list)

-o

Define a setting, -o name=value ("-o help" to dump current values). See [\[FluidSettings\]](#) for details

-p, --portname=[\[label\]](#)

Set MIDI port name (alsa\_seq, coremidi drivers)

-R, --reverb

Turn the reverb on or off [\[0|1|yes|no, default = on\]](#)

-r, --sample-rate

Set the sample rate

-s, --server

Start [\[FluidSynth\]](#) as a server process

-V, --version

Show version of program

-T, --audio-file-type

Audio file type for fast rendering or aufile driver ("-T help" for list)

-v, --verbose

Print out verbose messages about midi events

-V, --version

Show version of program

-z, --audio-bufsize=[\[size\]](#)

Size of each audio buffer

## SETTINGS

All settings are non-realtime (have no effect if set after startup), except for those indicated as realtime.

## SYNTHESIZER

synth.audio-channels INT [\[min=1, max=128, def=1\]](#)

Number of audio channels (DOCME!).

synth.audio-groups INT [\[min=1, max=128, def=1\]](#)

Number of audio groups (DOCME!).

synth.chorus.active BOOL [\[def=True\]](#)

Chorus effect enable toggle.

synth.cpu-cores INT [\[min=1, max=256, def=1\]](#)

Number of CPU cores to use for multi-core support.

synth.device-id INT [\[min=0, max=126, def=0\]](#) REALTIME

Device ID to use for accepting incoming SYSEX messages.

synth.dump BOOL [\[def=False\]](#)

No effect currently.

synth.effects-channels INT [\[min=2, max=2, def=2\]](#)

No effect currently.

synth.gain FLOAT [\[min=0.000, max=10.000, def=0.200\]](#) REALTIME

Master synthesizer gain.

synth.ladspa.active BOOL [\[def=False\]](#)

LADSPA subsystem enable toggle.

synth.midi-channels INT [\[min=16, max=256, def=16\]](#)

Total MIDI channel count (must be multiple of 16).

synth.midi-bank-select STR [\[def='gs' vals:'gm', 'gs', 'xg', 'mma'\]](#)

MIDI Bank Select message style.

synth.min-note-length INT [\[min=0, max=65535, def=10\]](#)

Minimum duration for note events (work around for very short percussion notes).

synth.overflow.age FLOAT [\[min=-10000, max=10000, def=1000\]](#)

Weighting (on overflow) for a voice's duration.

synth.overflow.percussion FLOAT [\[min=-10000, max=10000, def=4000\]](#)

Weighting (on overflow) for a voice being on the drum channel.

synth.overflow.released FLOAT [\[min=-10000, max=10000, def=-2000\]](#)

Weighting (on overflow) for a voice that has been released, i.e., note off and no sustain pedal.

synth.overflow.sustained FLOAT [\[min=-10000, max=10000, def=-1000\]](#)

Weighting (on overflow) for a voice that has been sustained, i.e., note off, but sustain pedal held down.

synth.overflow.volume FLOAT [\[min=-10000, max=10000, def=500\]](#)

Weighting (on overflow) for a voice's volume.

synth.parallel-render BOOL [\[def=True\]](#)

Enables low-latency audio rendering response, even if synth is otherwise busy. Should always to be true for usage by fluidsynth executable.

synth.polyphony INT [\[min=1, max=65535, def=256\]](#) REALTIME

Voice polyphony count (number of simultaneous voices allowed).

synth.reverb.active BOOL [\[def=True\]](#)

Reverb effect enable toggle.

synth.sample-rate FLOAT [\[min=22050.000, max=96000.000, def=44100.000\]](#)

Synthesizer sample rate.

synth.threadsafe-api BOOL [\[def=True\]](#)

Serializes access to the synth API. Must always to be true for usage by fluidsynth executable.

synth.verbose BOOL [\[def=False\]](#)

Print received MIDI events to stdout.

## GENERAL AUDIO

audio.driver STR

Audio driver to use. Default and valid options depend on available drivers.

audio.input-channels INT [\[min=0, max=2, def=0\]](#)

Not used currently? (DOCME).

audio.output-channels INT [\[min=2, max=32, def=2\]](#)

DOCME

audio.period-size INT [\[min=64, max=8192, def=64\]](#)

Period size for audio buffers. Used by many audio drivers.

audio.periods INT [\[min=2, max=64, def=16\]](#)

Count of audio buffers. Used by many audio drivers.

audio.realtime-prio INT [\[min=0, max=99, def=60\]](#)

Realtime priority to assign to audio thread or 0 to disable high priority scheduling. Only used by some audio drivers (currently 'alsa' and 'oss').

audio.sample-format STR [\[def='16bits' vals:'16bits','float'\]](#)

Audio output format, to select format for those drivers which support 16 bit or floating point.

## AUDIO DRIVER SPECIFIC

audio.alsa.device STR [\[def='default'\]](#)

ALSA audio driver output device.

audio.coreaudio.device STR [\[def='default'\]](#)

CoreAudio? driver output device. Valid options depend on system.

audio.dart.device STR [\[def='default'\]](#)

OS/2 Dart audio driver device.

audio.dsound.device STR [\[def='default'\]](#)

Device to use for DirectSound? driver. Valid options depend on system.

audio.file.endian STR [\[def='auto' vals:'auto','big','cpu','little'\]](#)

File renderer or file driver byte order selection. 'auto' selects the default for the selected file type. 'cpu' uses the CPU byte order. Limited to 'cpu' if no libsndfile support.

audio.file.format STR [\[def='s16' vals:'double','float','s16','s24','s32','s8','u8'\]](#)

File renderer or file driver audio format. Limited to 's16' if no libsndfile support.

audio.file.name STR [\[def='fluidsynth.wav'\]](#)

Output file name for file renderer or file driver.

audio.file.type STR [\[def='auto' vals:'aiff','au','auto','flac','oga','raw','wav'\]](#)

Output file type for file renderer or file driver. 'auto' attempts to determine type from file extension in audio.file.name. Limited to 'raw' if no libsndfile support. Actual options will vary depending on libsndfile library.

audio.jack.autoconnect BOOL [\[def=False\]](#)

If enabled, then [\[FluidSynth\]](#) is automatically connected to Jack system audio output ports.

audio.jack.id STR [\[def='fluidsynth'\]](#)

Client ID to use when connecting to Jack.

audio.jack.multi BOOL [\[def=False\]](#)

TRUE to enable multi-channel output.

audio.jack.server STR [\[def= \]](#)

Jack server name. Blank for default.

audio.oss.device STR [\[def='/dev/dsp'\]](#)

OSS driver output device.

audio.portaudio.device STR [\[def='PortAudio\]](#) Default'

[\[PortAudio\]](#) driver output device. Available options depends on system.

audio.pulseaudio.adjust-latency BOOL [\[def=True\]](#)

Increases the latency dynamically if PulseAudio? suggests so.

audio.pulseaudio.device STR [\[def='default'\]](#)

PulseAudio? driver output device.

audio.pulseaudio.media-role STR [\[def='music'\]](#)

PulseAudio? media role information.

audio.pulseaudio.server STR [\[def='default'\]](#)

PulseAudio? driver server.

## GENERAL MIDI

midi.driver STR

MIDI driver to use. Default and valid options depend on available drivers.

midi.realtime-prio INT [\[min=0, max=99, def=50\]](#)

Realtime priority to assign to MIDI thread or 0 to disable high priority scheduling. Only used by some MIDI drivers (currently 'alsa\_seq', 'alsa\_raw' and 'oss').

## MIDI DRIVER SPECIFIC

midi.alsa.device STR [\[def='default'\]](#)

ALSA raw MIDI driver device.

midi.alsa\_seq.device STR [\[def='default'\]](#)

ALSA sequencer MIDI driver device.

midi.alsa\_seq.id STR [\[def='pid'\]](#)

ALSA sequencer client ID. 'pid' will use process ID as part of the client name.

midi.coremidi.id STR [\[def='pid'\]](#)

Client ID to use for CoreMIDI driver. 'pid' will use process ID as port of the client name.

midi.jack.id STR [\[def='fluidsynth-midi'\]](#)

Jack MIDI driver client ID.

midi.jack.server STR [\[def= \]](#)

Jack MIDI driver server. Blank to use default.

midi.oss.device STR [\[def='/dev/midi'\]](#)

OSS MIDI driver device.

midi.portname STR [\[def= \]](#)

Port name used for CoreAudio? and ALSA sequencer drivers.

midi.winmidi.device STR [\[def='default'\]](#)

Device for Windows MIDI driver.

## MISCELLANEOUS

player.reset-synth BOOL [\[def=True\]](#)

TRUE to reset synthesizer MIDI state between MIDI songs.

player.timing-source STR [\[def='sample' vals:'sample','system'\]](#)

Selects timing source for MIDI sequencer. 'system' uses the system timer. 'sample' uses the sample clock (amount of audio output, events synchronized with audio).

shell.port INT [\[min=1, max=65535, def=9800\]](#)

Shell command server TCP/IP port number to use.

shell.prompt STR [\[def= \]](#)

Shell prompt string.

## SHELL COMMANDS

### GENERAL

help

Prints out a summary of the main commands

help help

Prints out list of other help topics (type "help <topic>")

quit

Quit the synthesizer

## SOUNDFONTS

load filename

Load a [\[SoundFont\]](#)

unload number

Unload a [\[SoundFont\]](#). The number is the index of the [\[SoundFont\]](#) on the stack.

fonts

Lists the current [SoundFonts](#) on the stack

inst number

Print out the available instruments for the [\[SoundFont\]](#).

## MIDI MESSAGES

noteon channel key velocity

Send a note-on event

noteoff channel key

Send a note-off event

cc channel ctrl value

Send a control change event

prog chan num

Send program-change message

select chan sfont bank prog

Combination of bank-select and program-change

channels

Print out the presets of all channels.

## AUDIO SYNTHESIS

gain value

Set the master gain ( $0 < \text{gain} < 5$ )



interp num

Choose interpolation method for all channels

interpc chan num

Choose interpolation method for one channel

## REVERB

reverb [\[0|1|on|off\]](#)

Turn the reverb on or off

rev\_preset num

Load preset num into the reverb unit

rev\_setroomsize num

Change reverb room size

rev\_setdamp num

Change reverb damping

rev\_setwidth num

Change reverb width

rev\_setlevel num

Change reverb level

## CHORUS

chorus [\[0|1|on|off\]](#)

Turn the chorus on or off

cho\_set\_nr n

Use n delay lines (default 3)

cho\_set\_level num

Set output level of each chorus line to num

cho\_set\_speed num

Set mod speed of chorus to num (Hz)

cho\_set\_depth num

Set chorus modulation depth to num (ms)

## MIDI ROUTER

router\_default

Reloads the default MIDI routing rules (input channels are mapped 1:1 to the synth)

router\_clear

Deletes all MIDI routing rules.

router\_begin [\[note|cc|prog|pbend|cpress|kpress\]](#)

Starts a new routing rule for events of the given type.

router\_chan min max mul add

Limits the rule for events on  $\text{min} \leq \text{chan} \leq \text{max}$ . If the channel falls into the window, it is multiplied by 'mul', then 'add' is added.

router\_par1 min max mul add

Limits parameter 1 (for example note number in a note events). Similar to router\_chan.

router\_par2 min max mul add

Limits parameter 2 (for example velocity in a note event). Similar to router\_chan.

router\_end

Finishes the current rule and adds it to the router.

### Router examples

router\_clear

router\_begin note

router\_chan 0 7 0 15

router\_end

Will accept only note events from the lower 8 MIDI channels. Regardless of the channel, the synthesizer plays the note on ch 15 ( $\text{synthchannel} = \text{midichannel} * 0 + 15$ ).

router\_begin cc

router\_chan 0 7 0 15

router\_par1 1 1 0 64

router\_end

Configures the modulation wheel to act as sustain pedal (transforms CC 1 to CC 64 on the lower 8 MIDI channels, routes to ch 15).

## LADSPA

LADSPA must be enabled for these commands to work (-o synth.ladspa.active=1)

ladspa\_clear

Resets LADSPA effect unit to bypass state

ladspa\_add lib plugin n1 <- p1 n2 -> p2 ...

Loads and connects LADSPA plugin

ladspa\_start

Starts LADSPA effect unit

ladspa\_declnode node value

Declares control node `node' with value 'value'

ladspa\_setnode node value

Assigns 'value' to 'node'

## AUTHORS

Peter Hanappe

Markus Nentwig

Antoine Schmitt

Joshua "Element" Green

Stephane Letz

Please check the AUTHORS and THANKS files for all credits

## DISCLAIMER

[SoundFont](#) is a registered trademark of E-mu Systems, Inc.