

# OSC Trigger

User's guide

## Requirements

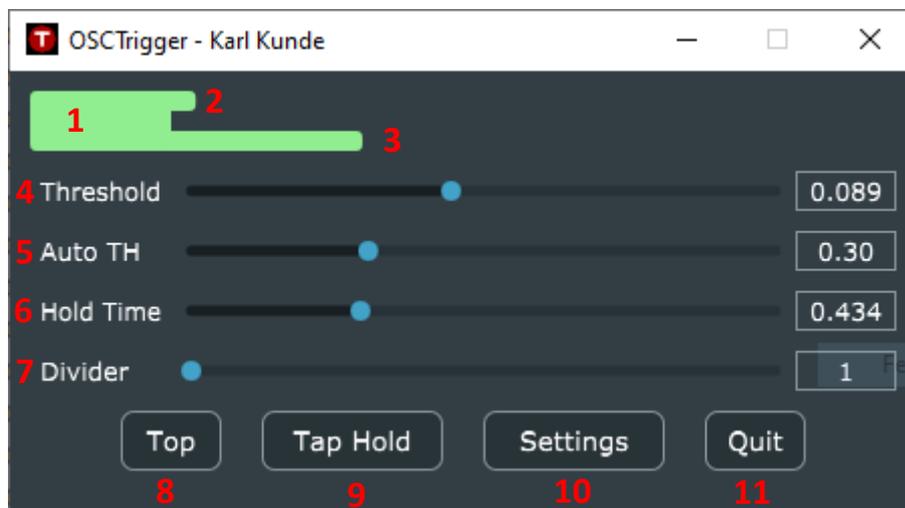
- PC with audio input (Line in)
- Target software (e.g. lighting software) that supports beat-tapping via OSC or keyboard shortcut
- Windows 7, 8 or 10
- Both, 32- and 64-bit OS are supported

## Installation

For installation, simply run "OSCTriggerSetup.exe". If Windows displays a security warning, just choose "Further Information" and "Run anyway" to run the installer. If necessary, the installer will install Microsoft Visual C++ Redistributables automatically. At the first program start-up you will be asked for a license file. You may obtain a demo or permanent license by [mailing](#) us. If you don't have a license file, you may try the software for 15 minutes before you have to restart it.

## The main window

This window contains all main functionality. You may close this window while the application keeps running as symbol in the system tray.

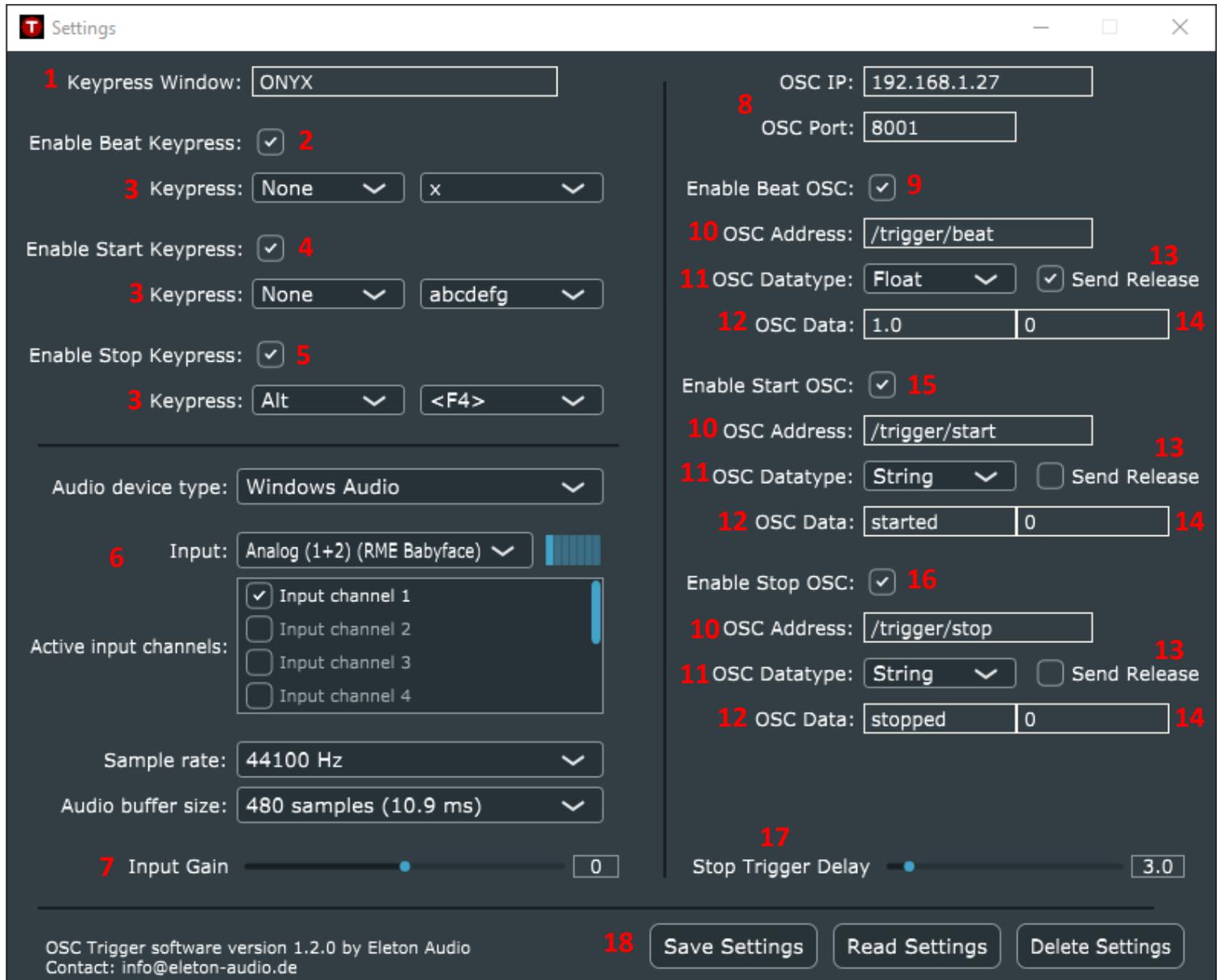


1. Trigger Indicator: Turns orange when a beat is detected and stays for the hold-time or violet when a beat was detected but suppressed (see (7)). Turns green when ready for next beat. Turns grey if "Stop Triggering" is enabled and no beats occurred for some time (see settings). Click here to trigger manually.
2. Input Meter: Show the (filtered) input level on which the detection is based.
3. Trigger Threshold: This is the actual Threshold above which a beat is detected. If the level of the input meter (2) exceeds this, a beat is detected.
4. Trigger Threshold Slider: Adjust the trigger threshold here.
5. Auto Threshold Slider: If moved above zero, this allows the threshold to adapt to the music signal (indicated by moving trigger threshold (3)). Increasing widens the allowed moving range, setting to zero fixes the threshold to the value set by the slider (4).

6. Hold Time: After a beat is detected, a hold time is applied. During this time, no further beat can be detected. Use this to avoid unwanted triggering between beats.
7. Beat Divider: Allows suppressing of all beats except every “n-th” detected beat. Like that it is e.g. possible to get half or quarter of the music tempo. Supressed beats are shown by violet colour of the indicator (1).
8. Keep On Top Button: Keep this window on top of all other windows, even if another one is selected.
9. Tap Hold Button: If you tap here, the hold time will be set to a value a little bit faster than you tapped. So, if you tap the actual beat, the hold time will be over shortly before the next beat.
10. Settings Button: Show settings window.
11. Quit Button: Quit program. Otherwise, if you hit the close button in the title bar, OSC Trigger will keep running and only minimize to the system tray.

## The settings window

Here the basic settings can be found. All changes are applied immediately, and the window simply can be closed. The settings are stored permanently when closing the application or by clicking the “Save Settings” button.



The basic principle consists of three different trigger types:

- A “Beat” trigger is generated at each musical beat of the audio signal. So, if the settings in the main window are reasonably matching the music material, there is a stream of triggers in tempo with the music. With this, e.g. a lighting cue that steps through a chaser with the beat of the music can be triggered.
- A “Stop” trigger is generated when the music stops for some (adjustable amount of) time, more exactly if there are no “Beat” triggers recognized for some time. With this, e.g. a lighting scene for music pauses can be triggered.
- A “Start” trigger is generated if the music (beat) starts again after a “Stop” trigger. With this trigger you could switch back from the pause lighting scene to a beat synchronized one.

All three triggers now can be sent via two different methods:

- OSC (Open Sound Control) is a network-based protocol supported by many devices and software products. This is the preferred transmission protocol as it is very reliable once it is up and running.
- For target software that does not support OSC, keypresses (from the PC keyboard) can be simulated. This works well but is limited by the operating system (Windows) in a way that keypresses only can be sent to the window that is currently in the foreground (e.g. last clicked in). To avoid distraction while working with other programs, OSC Trigger can suppress the generation of keypresses if your target software is not in the foreground.

The “Keypress” method, as an additional feature, can be used to randomize the triggered scenes. Therefore, if multiple printable character (e.g. “abcdefg123”) are entered in the Keypress field, a random one of them is chosen at every trigger.

Explanation of all the GUI elements:

1. Keypress Window: Here, the title of the target software window can be entered. If so, only if the currently “active” window (in the Windows OS) contains the supplied text in its title, a keypress will be simulated. E.g. if you enter “ONYX” keypresses will only be simulated if your last interaction was in the “ONYX” Window and not if, let’s say “Firefox” was last clicked in. Leave empty to disable this and always simulate keypresses.
2. Enable “Beat” Keypress: Here you can enable the simulation of a keypress when a beat is detected (also see explanation of trigger types above). Can be used together with “Enable Beat OSC”, but typically you will choose one of them.
3. Keypress: Choose the keypress to simulate. You may choose a modifier key and the key itself. For the second you may enter a printable character by clicking the textbox or choose a special key by clicking the arrow to show the options. If multiple printable characters (e.g. “abcdef”) are entered, a random one of the entered ones is chosen at each trigger event.
4. Enable “Start” Keypress: Here you can enable the simulation of a keypress when the music starts again after a pause (also see explanation of trigger types above). Can be used together with “Enable Start OSC”, but typically you will choose one of them.
5. Enable “Stop” Keypress: Here you can enable the simulation of a keypress when the music has stopped for some adjustable (see 17) time (also see explanation of trigger types above). Can be used together with “Enable Stop OSC”, but typically you will choose one of them.
6. Audio Setup: Select your audio device (and if needed some options) here. Usually you want to select your PC’s line input here.
7. Input Gain: If your input signal is too weak or too strong and you can’t adjust the level in hardware, you may adjust the input gain here (in dB).
8. OSC IP and Port: The IP address to which the trigger message should be sent. Set to “localhost” if the target is a software running on the same PC. And the Network port to which the OSC message should be sent. Find this out (or set this) in your receiving application.
9. Enable Beat OSC: Check this to transmit “beats” via OSC (also see explanation of trigger types above). Can be used together with “Enable Beat Keypress”, but typically you will choose one of them.
10. OSC Address: Every OSC message is targeted to an address, that specifies what this message contains. Find out in your receiving application which address to use for the beat trigger.

11. OSC Datatype: Additionally, every OSC message can contain different types of payload. OSC Trigger supports "Integer", "Float", "String" and "No Data". Common type can be "Float" but try to find out in your receiving application.
12. OSC Data: Enter the payload to use for the first (or the only) message here. This must match the datatype (11). E.g. "42" for integer, "42.0" for float, "forty-two" for string.
13. Send Release option: If enabled, for each trigger two OSC messages are sent: One immediately when a trigger is generated and another one a little bit later with a different payload. This can be used to simulate a "Tap" button that usually sends a "Press" and a "Release" message.
14. OSC Data for second message: Enter the payload for the second message (see 13) here. E.g. you may enter a "1" for the first message (12) and a "0" for the second message (here) if for your target software button "press" and "release" actions are needed.
15. Enable Start OSC: Check this to transmit an OSC message when the music starts after a pause (also see explanation of trigger types above). Can be used together with "Enable Start Keypress", but typically you will choose one of them.
16. Enable Stop OSC: Check this to transmit an OSC message when the music stops for some adjustable (see 17) time (also see explanation of trigger types above). Can be used together with "Enable Stop Keypress", but typically you will choose one of them.
17. Stop Trigger Delay: Here you can set the time (in seconds) of silence (no recognized beats) after that a "Stop" trigger is generated.
18. Save Settings: Stores the settings to a file, so they will be loaded at the next application start. If you don't hit this, the settings will be saved on application quit.  
 Read Settings: Reads the settings from the settings file and applies them. This may cause a short dropout because the audio device is reset.  
 Delete Settings: Deletes the settings file and restores factory defaults.

### Command line parameters

The following functions can be used by passing the specified parameters (without quotation marks) by command line:

- "hide" starts the program without showing the main window. To access the options simply double-click the symbol in the system tray.

All parameters also can be used while OSC Trigger already is running, simply starting it a second time will *not* create a second instance but apply the options to the already running one.

### Parameters in settings file

The settings are stored in a file in your user directory. You may find this file under "C:\Users\USERNAME\AppData\Roaming\OSCTrigger\OSCTrigger.stgs". You may open and edit this file with any text editor. If you can't find the file, make sure you have clicked "Save Settings" in the settings window of OSC Trigger at least once. The following options are possible in extension to the parameters editable from the settings window:

Parameter	Description	Default
osc-releasetime	Time after which the second OSC message ("Button-release") will be sent, in milliseconds.	100
filter-lp-f	Frequency of lowpass component of detection filter, in Hertz	180
filter-lp-q	Quality of lowpass component of detection filter	2

filter-bp-f	Frequency of bandpass component of detection filter, in Hertz	45
filter-bp-q	Quality of bandpass component of detection filter	0.7
filter-env-f	Frequency of envelope component of detection filter, in Hertz	5
filter-env-q	Quality of envelope component of detection filter	0.9

The easiest way to add such an option is by copying and pasting an existing entry. Optionally you can add a line yourself between the *PROPERTIES* tags, e.g. `<VALUE name="osc-releasetime" val="100"/>`