OSC Matrix

User‘s guide

Inhalt

Requirements

- Windows 7, 8, 10 or 11
- Both, 32- and 64-bit OS are supported

Installation

For installation, simply run “OSCMatrixSetup.exe”. If necessary, the installer will install Microsoft Visual C++ Redistributables automatically. At the first program start 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 must restart it.

Overview

The main feature of this application is the distribution and routing of OSC messages. OSC (Open Sound Control) is a network protocol which is used to send control functions from one device (soft- or hardware) to another. It can be seen as a "modern" alternative to MIDI and clearly does not carry any audio signal but only control commands. Usually, devices (software or hardware) allow to setup one network port to listen for OSC messages and one IP/Port combination to send outgoing messages to. Therefore, connecting three or more OSC devices can be difficult, in some cases even impossible.

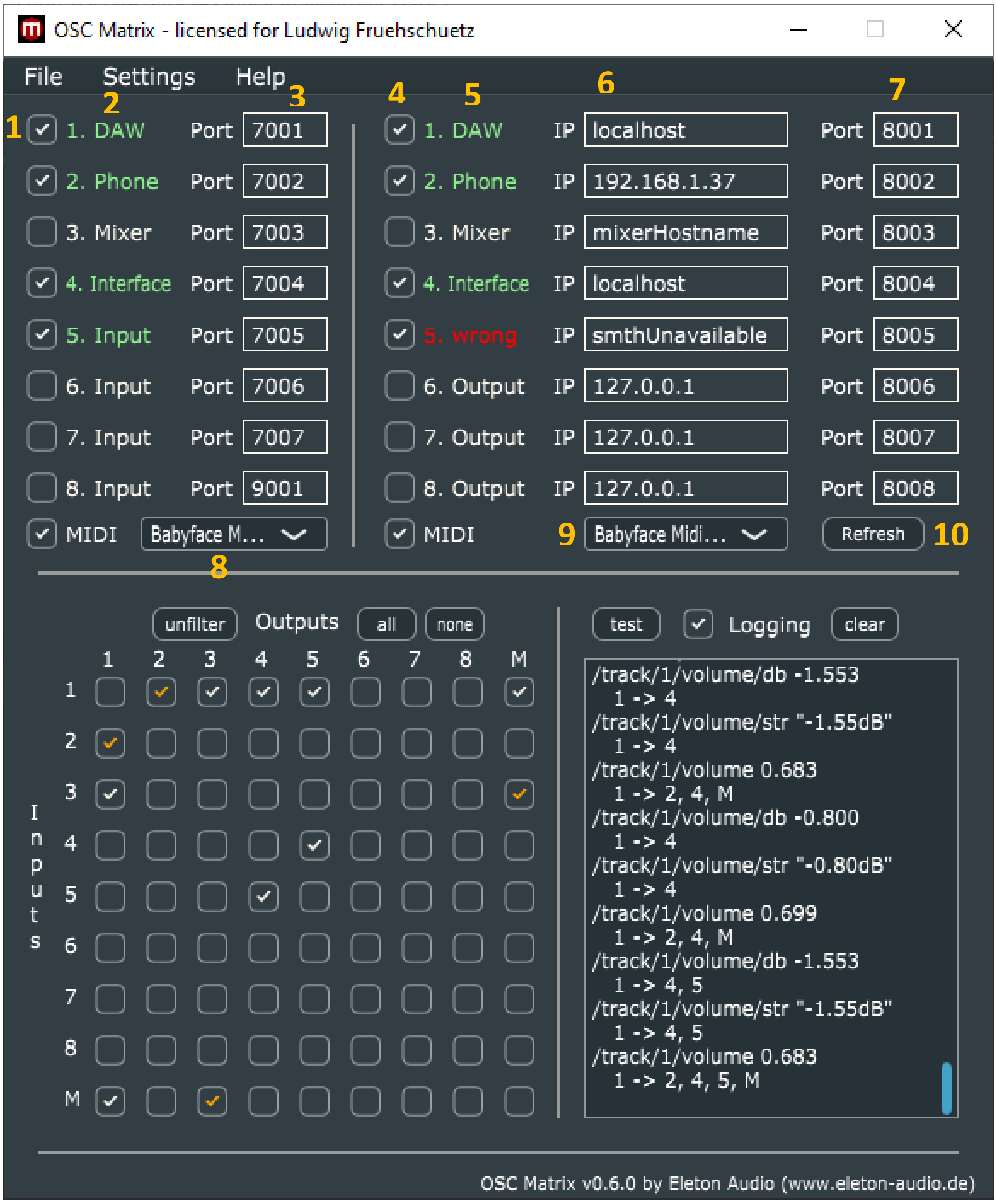
To solve this problem, OSC Matrix on its inputs takes the OSC messages from the outputs of up to 8 OSC enabled devices. On the other side it has up to 8 outputs that can send messages back to the same devices. In between there is a powerful routing matrix that allows flexible routing and filtering of messages.

An additional feature is the integration of one MIDI in- and output to the network of OSC devices. To connect the (otherwise incompatible) communication standards, two tables are used to translate OSC->MIDI and MIDI->OSC.

Features:

- 8 OSC in- and outputs
- 1 MIDI in- and output, translated via flexible translation tables
- Routing via routing matrix, including per-junction filtering
- Handy logging functionality
- Generation of user specific keepalive messages
- Support for "listen on outgoing" network ports
- Export and Import functionalities for all settings

Inputs and Outputs


OSC Matrix - licensed for Ludwig Fruehschuetz
File
Settings
Help
1
2
3
4
5
6
7
8
9
10
unfilter
Outputs
all
none
test
Logging
clear
Refresh
Inputs
OSC Matrix v0.6.0 by Eleton Audio (www.eleton-audio.de)

	Input		Port
☑	1. DAW	Port	7001
☑	2. Phone	Port	7002
□	3. Mixer	Port	7003
☑	4. Interface	Port	7004
☑	5. Input	Port	7005
□	6. Input	Port	7006
□	7. Input	Port	7007
□	8. Input	Port	9001
☑	MIDI		Babyface M...

	Output		IP		Port
☑	1. DAW	IP	localhost	Port	8001
☑	2. Phone	IP	192.168.1.37	Port	8002
□	3. Mixer	IP	mixerHostname	Port	8003
☑	4. Interface	IP	localhost	Port	8004
☑	5. wrong	IP	smthUnavailable	Port	8005
□	6. Output	IP	127.0.0.1	Port	8006
□	7. Output	IP	127.0.0.1	Port	8007
□	8. Output	IP	127.0.0.1	Port	8008
☑	MIDI		Babyface Midi...		

☑ Logging

	1	2	3	4	5	6	7	8	M
1	□	☑	☑	☑	☑	□	□	□	☑
2	☑	□	□	□	□	□	□	□	□
3	☑	□	□	□	□	□	□	□	☑
4	□	□	□	□	☑	□	□	□	□
5	□	□	□	☑	□	□	□	□	□
6	□	□	□	□	□	□	□	□	□
7	□	□	□	□	□	□	□	□	□
8	□	□	□	□	□	□	□	□	□
M	☑	□	☑	□	□	□	□	□	□

```
/track/1/volume/db -1.553
  1 -> 4
/track/1/volume/str "-1.55dB"
  1 -> 4
/track/1/volume 0.683
  1 -> 2, 4, M
/track/1/volume/db -0.800
  1 -> 4
/track/1/volume/str "-0.80dB"
  1 -> 4
/track/1/volume 0.699
  1 -> 2, 4, M
/track/1/volume/db -1.553
  1 -> 4, 5
/track/1/volume/str "-1.55dB"
  1 -> 4, 5
/track/1/volume 0.683
  1 -> 2, 4, 5, M
```

In the upper half of the main window, all the inputs and outputs are defined. Before you can use a device and dispatch messages to/from it, you have to setup one of OSC triggers inputs and outputs for it.

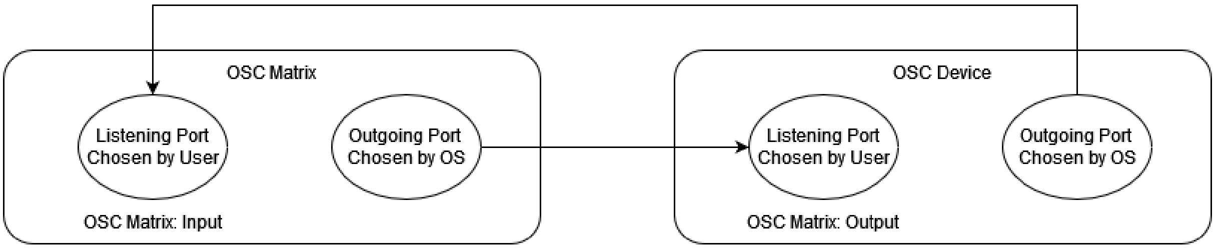
1. Input Enable Box: tick this box to enable the input
2. Input Number and Name: to change the name of the input, right-click the label
3. Input Port: enter the network port to which OSC Matrix should listen to (or enter 0 for automatic port selection, see below)
4. Output Enable Box: tick this box to enable the output
5. Output Number and Name: to change the name of the output, right-click the label
6. Output IP: enter the IP address to which outgoing OSC messages should be sent (“localhost” for software on the same PC)

7. Output Port: enter the network port to which outgoing OSC messages should be sent (the incoming port on your OSC receiver)
8. Choose the MIDI input device here
9. Choose the MIDI output device here
10. Refresh MIDI device list

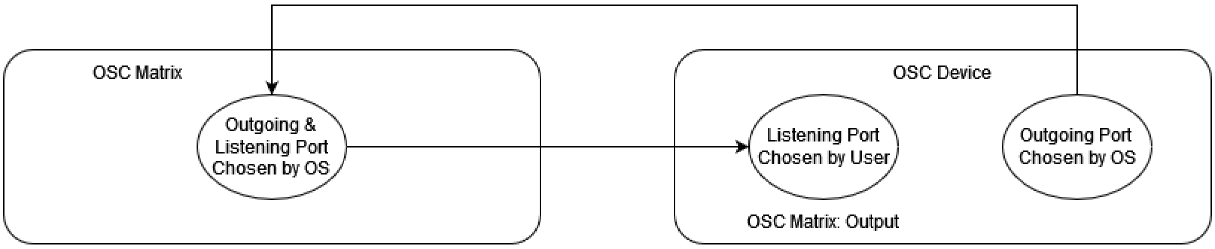
Usually, you want each of your devices to send its messages to one input of OSC Matrix and receive messages from one output of OSC Matrix. To keep things simple, use the same input and output number for one device. If e.g., you have a smartphone app sending to port 7002 of your PC and receiving messages on port 8002 and your smartphone has the IP 192.168.1.39 you would setup input and output number 2 like in the picture above.

Some OSC devices do not allow you to enter an “output” OSC port to which outgoing messages are sent but always send their messages to the port from which they receive the incoming messages. Please note that this “outgoing” port usually is not set up by the user but automatically chosen by the operating system (Windows). To be able to communicate with such devices, you can enter “0” as input port (Nr. 3 above) in OSC Matrix which reuses the outgoing port as input. Known devices that use this logic currently are Behringer X32, MIDAS M32 and Tangent Mapper software. To check the used port, point the mouse at the input label (Nr. 2) which shows the port number in the tooltip.

Fixed Listening Port


OSC Matrix
Listening Port Chosen by User
Outgoing Port Chosen by OS
OSC Matrix: Input
OSC Device
Listening Port Chosen by User
Outgoing Port Chosen by OS
OSC Matrix: Output

Listening Port Set To 0


OSC Matrix
Outgoing & Listening Port Chosen by OS
OSC Device
Listening Port Chosen by User
Outgoing Port Chosen by OS
OSC Matrix: Output

If OSC Matrix successfully opened a network port for sending or receiving messages, the name of the corresponding input or output turns green. But if then it fails to send messages for 2 times to an output (e.g. if the hostname you entered as a target cannot be resolved to an IP), the name turns red and the output is disabled. To retry sending messages to it, disable and re-enable the output.

The Matrix

In the matrix in the lower left of the main window, all the message routing is done. The rows correspond to the inputs defined in the input/output section, the columns show the outputs.

If you e.g. want to route all messages from input 3 to output 4, tick the box in line 3, column 4. To help you, if you hover over one box with the mouse for a short moment, it will show you the exact routing this box represents. Next to the 8 OSC inputs and outputs also the MIDI connection is shown in the matrix, in line (and column) “M”.

To untick all boxes, you may click the button “none” above the matrix. Clicking button “all” ticks all boxes.

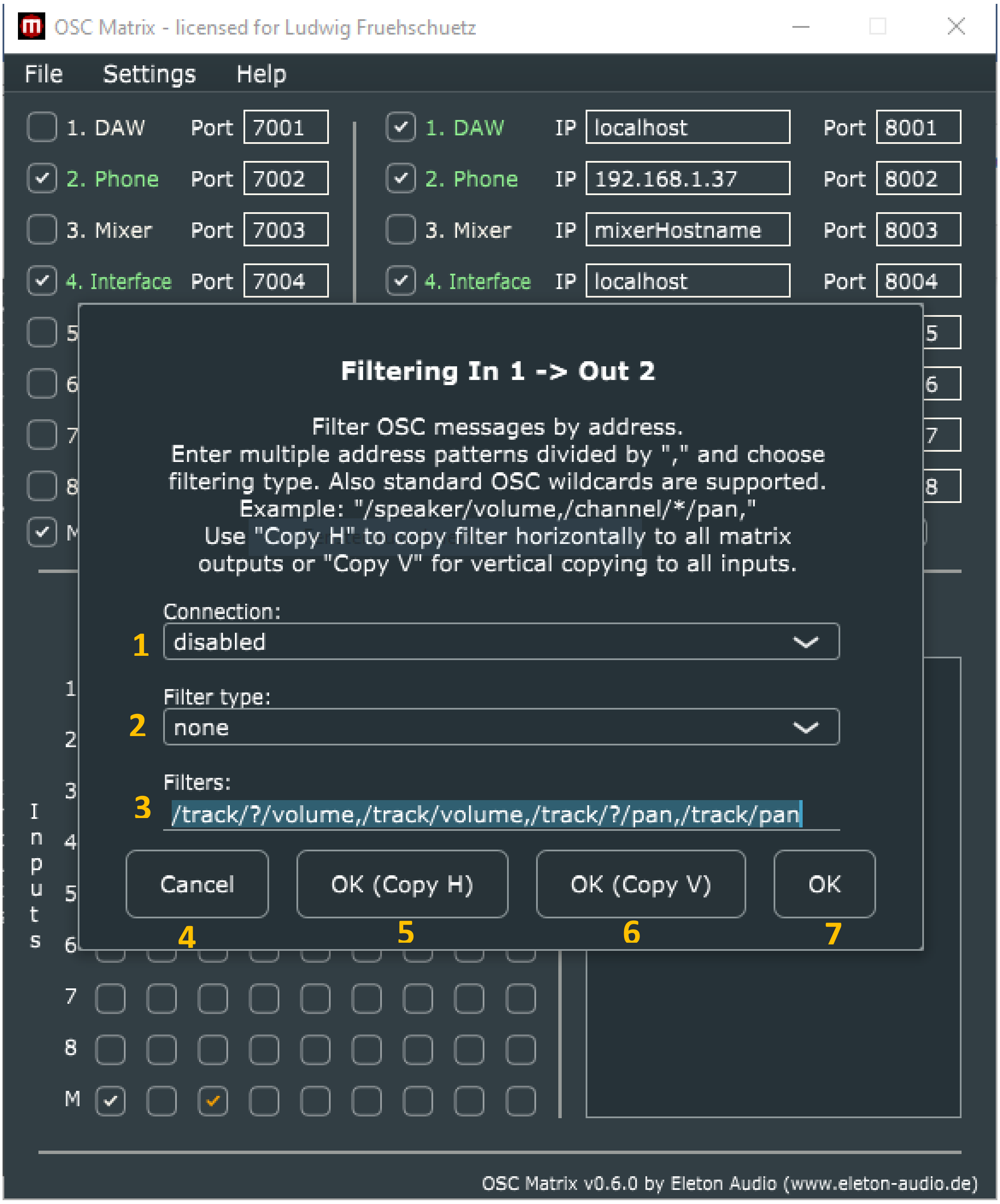
Filtering

To extend the routing options, every routing crosspoint has filtering options. Only if the filter lets one specific message pass, it will be sent to the output. If some filtering is set for a crosspoint, the tick is displayed in orange. To access the filter settings, right-click the crosspoint. To reset all filters, hit the button “Unfilter” above the matrix.

There are two filter types. A whitelist filter which only lets specific messages pass or a blacklist filter which filters out specific messages. To specify the list of messages for the filter you may enter multiple OSC addresses divided by a comma (“,”). An OSC address must begin with a “/” and may contain multiple layers, like “subfolders”, e.g. “/volume/track/42”. Because there is no standard that tells which address to use for a specific function, please refer to the manual of your OSC enabled device or try it and use the logging function to view used addresses.

Additional to a plain list of addresses, standard OSC wildcards can be used for the filter string. A complete explanation on OSC (and its wildcards) can be found on http://opensoundcontrol.org/.

Wildcard	Matching
?	One character or number
*	String of one or multiple characters (excluding “/”)
[…]	One of the characters inside the brackets. [a-c] or [abc] matches a,b or c. [!abc] matches everything but a,b or c.
{one,two,…}	One of the strings inside the brackets


OSC Matrix - licensed for Ludwig Fruehschuetz
File
Settings
Help
Inputs
OSC Matrix v0.6.0 by Eleton Audio (www.eleton-audio.de)

☐ 1. DAW Port 7001 | ☑ 1. DAW IP localhost Port 8001

☑ 2. Phone Port 7002 | ☑ 2. Phone IP 192.168.1.37 Port 8002

☐ 3. Mixer Port 7003 | ☐ 3. Mixer IP mixerHostname Port 8003

☑ 4. Interface Port 7004 | ☑ 4. Interface IP localhost Port 8004

Filtering In 1 -> Out 2

Filter OSC messages by address.
Enter multiple address patterns divided by "," and choose filtering type. Also standard OSC wildcards are supported.
Example: "/speaker/volume,/channel/*/pan,"
Use "Copy H" to copy filter horizontally to all matrix outputs or "Copy V" for vertical copying to all inputs.

Connection: disabled

Filter type: none

Filters: /track/?/volume,/track/volume,/track/?/pan,/track/pan

1. Connection: enables the connection from the corresponding input to the output. This is the same as the tick box in the matrix
2. Filter Type: Here you can select the type of filter which should be applied to the crosspoint
   a. None: Messages are not filtered, but just passed to the output
   b. Whitelist: Only specific messages are passed to the output, all others are filtered
   c. Blacklist: Specific messages are not being passed to the output but filtered out
3. Filter: List of messages to filter (corresponding to filter type)
4. Cancel: Don’t change settings and close the window
5. OK (Copy H): Apply the settings and copy them to all outputs, thus horizontally in the matrix
6. OK (Copy V): Apply the settings and copy them to all inputs, thus vertically in the matrix
7. OK: Apply the settings to this crosspoint.

MIDI channel filtering

For the crosspoints in the MIDI row there is the option to also filter for a specific MIDI channel. If a channel is selected, only MIDI messages with this channel are passed to the OSC output. “All channels” lets all MIDI messages pass. Additional to the channel filter, also the OSC address filter is applied to the MIDI messages (after they are translated to OSC messages).

For the crosspoints in the MIDI column there is also the option to set a MIDI channel. If a channel is selected, all MIDI messages from the corresponding input are sent to this channel on the MIDI output. This setting overrides the MIDI channel which is set by the translation from OSC to MIDI via the translation table. To not override this setting, choose “Use channel from translation table” here.

Logging

In the logging section in the main window, all incoming messages can be displayed. To turn logging on, tick the box over the “Logging” view. Logging will be disabled at program start to increase performance. Also, when the program is minimized to the system tray, logging will be off. To clear the logging view, click the “clear button above.

A log entry contains the OSC address, the data, the source and the destinations of the message. The data is formatted like “42” for datatype Integer, “42.0” for float and “”42”” (with quotes) for String. Only active outputs are shown as destination for a message.

Also, there is the possibility to send a test message to all active outputs. If you click “Test” above the logging box, a OSC message with address “/test” and data 1.234 (float) will be sent. Note that a MIDI message only will be sent if a valid translation is produced by the translation table.

MIDI Integration

All MIDI messages that sent through the routing matrix are translated to OSC messages by user configurable translation tables. There are two tables, one converts incoming MIDI messages to OSC messages (MIDI->OSC translation), the second one converts outgoing OSC messages that are routed to the MIDI output to MIDI (OSC->MIDI translation).

To edit one of the translation tables, go to the settings menu and choose which table to edit.

Technical background

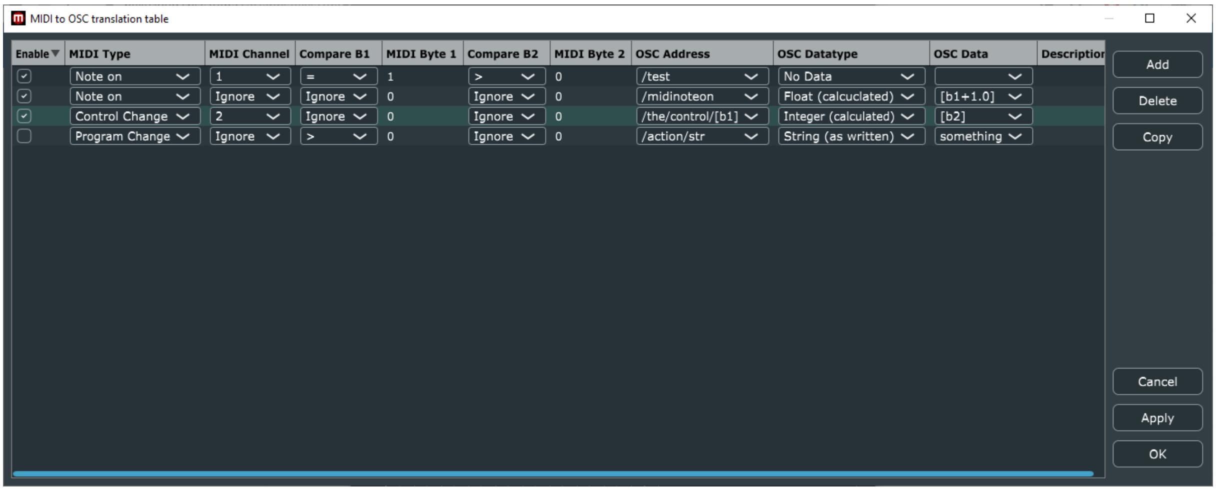
For a better understanding, it may be helpful to know the basic principles of OSC and MIDI. A very brief overview, do not consider this as complete:

Every OSC message contains an “address” that tells the receiver what the following data is about. The address looks like a hierarchical directory structure, e.g. to change the volume of input channel 1 one could use an OSC address like “/input/1/volume”, but there is no “standard” about which address is used for which data. Every device may define its own set of OSC addresses. Additionally, every OSC message may contain some payload which may be of the type “integer”, “float” or “string”. Further information can be found here: http://opensoundcontrol.org/.

Every MIDI message (ok, most of them) contains 3 “bytes”. The first byte always tells the receiver the MIDI “channel” and the “message type”, again what the following data is about. Byte 2 and Byte 3 contain two values between 0 and 127 that have a meaning determined by the specified message type. There are several message types, the MIDI Association has a good overview over them and the use of the data bytes: https://www.midi.org/specifications-old/item/table-1-summary-of-midi-message

MIDI->OSC translations

Every row of the table contains one translation pair. If an incoming MIDI message matches the (MIDI) criteria defined in one of the rows, an OSC message is constructed and sent to the output defined by the matrix. There also can be multiple rows in the table matching one specific message, resulting in multiple translated OSC messages being sent. If no row matches an incoming message, no OSC message is generated.



Enable	MIDI Type	MIDI Channel	Compare B1	MIDI Byte 1	Compare B2	MIDI Byte 2	OSC Address	OSC Datatype	OSC Data	Descriptior
☑	Note on	1	=	1	>	0	/test	No Data		
☑	Note on	Ignore	Ignore	0	Ignore	0	/midinoteon	Float (calcuclated)	[b1+1.0]	
☑	Control Change	2	Ignore	0	Ignore	0	/the/control/[b1]	Integer (calculated)	[b2]	
☐	Program Change	Ignore	>	0	Ignore	0	/action/str	String (as written)	something	

In order not to have to create a translation row for every possible data set of an incoming message, the generated OSC message may contain data from the incoming midi message. For example, it is easily possible to create a translation row matching every incoming “Note on” MIDI message that generates an OSC message with address “/midi/noteon” that carries the specified note number as data. In addition, basic calculations (like +, -, *, /) are supported while defining output messages.

Enable

If this box is unticked, the translation row is disabled and not used for translations. Like that you can keep settings that are currently not in use.

MIDI Type

Choose the MIDI message type(s) that match the translation row. The selection “Ignore Type” makes the row match every incoming MIDI message type.

MIDI Channel

Selects the MIDI channel(s) that match the translation row. Select “Ignore Channel” to match MIDI messages on all channels.

Compare Byte 1,2

Define a compare mode with that the MIDI data bytes are checked for a match. The compare function compares the byte from the incoming message to the data entered in the “Byte 1, 2” field.

Ignore	Don’t compare the data bytes, but always match bytes
=	Match bytes that are equal to the data entered in the table
!=	Match bytes that are not equal to the data entered in the table
>	Match bytes that are greater than the data entered in the table
<	Match bytes that are smaller than the data entered in the table
>=	Match bytes that are greater or equal to the data entered in the table
<=	Match bytes that are smaller or equal to the data entered in the table
Range incl. Range excl.	Match bytes that are in the range entered as compare data. E.g. enter “5~11” (without quotes) to match the range between 5 and 11, either including or excluding the limits.

OSC Address

The address of the generated OSC message. This can be either be a static OSC address (e.g. “/some/osc/address”) or contain parts from the incoming MIDI message. Examples can also be found in the dropdown menu of the table.

[b1]	Is replaced by the data in MIDI byte 1. E.g. /midi/[b1]/incoming
[b2]	Is replaced by the data in MIDI byte 2. E.g. /midi/[b2]/incoming
[c]	Is replaced by the data in MIDI channel. E.g. /midi/[c]/incoming
+	Addition of one of previous values with a constant value, e.g. /midi/[b1+42.1]/incoming
-	Subtraction of a constant value from one of previous values, e.g. /midi/[b1-42.1]/incoming
*	Multiplication of one of previous values with a constant value, e.g. /midi/[b1*42.1]/incoming
/	Division of one of previous values by a constant value, e.g. /midi/[b1/42.1]/incoming
#	Map a range of input values to a range of output values. E.g. use /midinote/[b1#0-127#0.0-1.27]/on to map incoming MIDI “byte1” values between 0 and 127 to a OSC float value between 0.0 and 1.27

OSC Datatype

The type of data which will be added to the generated OSC message. Next to the standard OSC datatypes, you can also select to calculate the data from the incoming MIDI message. To do so, you must select one of the options with “calculated” in the name and enter the calculation rule in the OSC Data field.

OSC Data

The actual data to send with the OSC message. If you have chosen to send static data with the message (and chosen one of the datatype options without “calculated”), just enter the data here (e.g. “42” for an Integer, “42.1” for a float, “forty-two” for a string). If you want to calculate the data from the incoming MIDI message, use these calculation symbols:

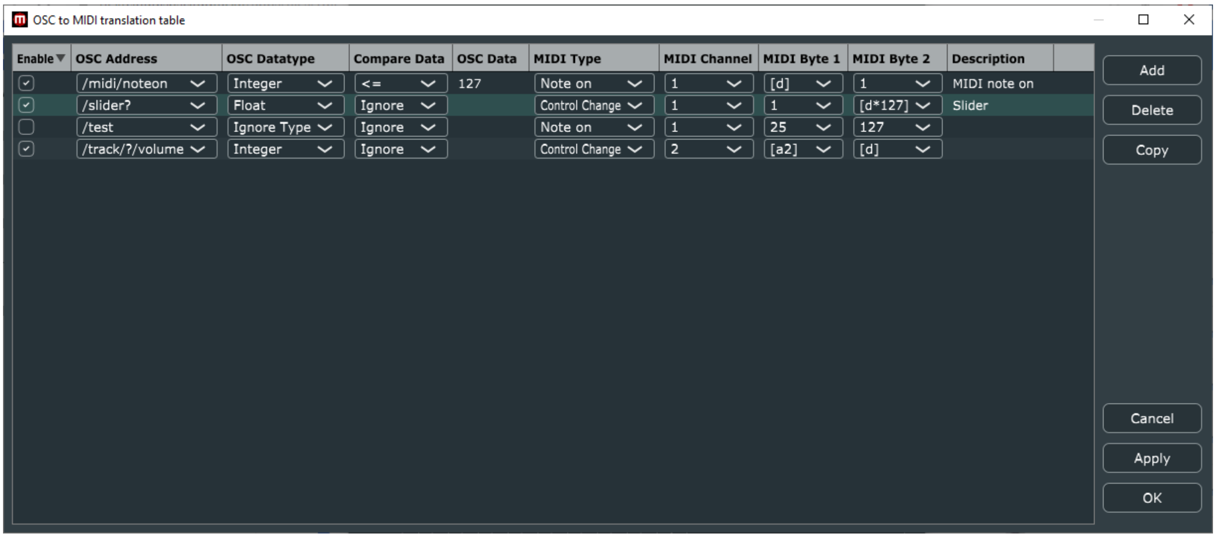
[b1]	Is replaced by the data in MIDI byte 1.
[b2]	Is replaced by the data in MIDI byte 2.
[c]	Is replaced by the data in MIDI channel.
+	Addition of one of previous values with a constant value, e.g. [b1+42.1]
-	Subtraction of a constant value from one of previous values, e.g. [b1-42.1]
*	Multiplication of one of previous values with a constant value, e.g. [b1*42.1]
/	Division of one of previous values by a constant value, e.g. [b1/42.1]
#	Map a range of input values to a range of output values. E.g., use [b1#0-127#0.0-1.27] to map incoming MIDI “byte1” values between 0 and 127 to a OSC float value between 0.0 and 1.27

Description

A user description of the row. E.g. what this type of message does in your application.

OSC->MIDI translations

Like in the previous chapter, here every table row contains one translation pair. An incoming OSC message that matches one (or several) OSC criteria in the rows will fire the corresponding midi message.



OSC to MIDI translation table

Enable	OSC Address	OSC Datatype	Compare Data	OSC Data	MIDI Type	MIDI Channel	MIDI Byte 1	MIDI Byte 2	Description
☑	/midi/noteon	Integer	<=	127	Note on	1	[d]	1	MIDI note on
☑	/slider?	Float	Ignore		Control Change	1	1	[d*127]	Slider
☐	/test	Ignore Type	Ignore		Note on	1	25	127	
☑	/track/?/volume	Integer	Ignore		Control Change	2	[a2]	[d]	

The generated MIDI message also can be constructed from the incoming OSC message (its specific address or the payload). Again, like in the previous chapter for example every “/midi/noteon” message can fire a MIDI “note on” message with the note number specified by the OSC payload data. Also, the calculations modes are available here.

Enable

If this box is unticked, the translation row is disabled and not used for translations. Like that you can keep settings that are currently not in use.

OSC Address

The address pattern against which incoming OSC messages are compared. This may contain standard OSC wildcards (also see http://opensoundcontrol.org):

Wildcard	**Matching**
?	One character or number
*	String of one or multiple characters (excluding “/”)
[…]	One of the characters inside the brackets. [a-c] or [abc] matches a,b or c. [!abc] matches everything but a,b or c.
{one,two,…}	One of the strings inside the brackets

OSC Datatype

If anything else than “Ignore Datatype” is selected, only OSC messages with payload of the specified type will match this row.

Compare Data

Define a compare mode with that the OSC payload is checked for a match. The compare function compares the data from the incoming message to the data entered in the “OSC Data” field.

Ignore	Don’t compare the data bytes, but always match bytes
=	Match values that are equal to the data entered in the table
!=	Match values that are not equal to the data entered in the table
>	Match values that are greater than the data entered in the table
<	Match values that are smaller than the data entered in the table
>=	Match values that are greater or equal to the data entered in the table
<=	Match values that are smaller or equal to the data entered in the table
Range incl. Range excl.	Match bytes that are in the range entered as compare data. E.g. enter “5.3~11.7” (without quotes) to match the range between 5.3 and 11.7, either including or excluding the limits.

MIDI Type

Choose the MIDI message type to be send on a match.

MIDI Channel

Choose the MIDI channel for the message that will be sent on a match. Attention: this setting can be overwritten by each matrix crosspoint individually when something else than “Use channel from translation table” is selected in the filtering view.

MIDI Byte 1 / MIDI Byte 2

Specifies the two payload bytes for the MIDI message that will be sent. This can either be a directly entered value (between 0 and 127) or a value calculated from the incoming OSC message. For a calculation, the following symbols may be used:

[a1]	Is replaced with the first part of the OSC address (which is split to /[a1]/[a2]/[a3]/[a4]/…)
[d]	Is replaced with the data of the OSC message
+	Addition of one of previous values with a constant value, e.g. [a1+42.1]
-	Subtraction of a constant value from one of previous values, e.g. [a2-42.1]
*	Multiplication of one of previous values with a constant value, e.g. [d*42.1]
/	Division of one of previous values by a constant value, e.g. [d/42.1]
#	Map a range of input values to a range of output values. E.g. use [d#0.0-1.0#0-100] to map incoming OSC data from a value between 0 and 1 to a MIDI byte between 0 and 100

Description

A user description of the row. E.g. what this type of message does in your application.

Advanced translations: SysEx

To some extent, also System Exclusive MIDI messages can be processed. These SysEx messages consist of more than three bytes and are quite specific to the manufacturer of the MIDI device.

Every SysEx message contains one or two bytes (well, technically three bytes) as “manufacturer ID”. OSC Matrix can filter for this ID in the MIDI->OSC translation table as follows:

Compare B1	**Compare B2**	**Matching Manufacturer ID**
== 2	== 3	ID: 02 03; ex. message: 0xf0 0x00 0x02 0x03 […data…] 0xf7
== 2	“Ignore”	ID: 02; ex. message: 0xf0 0x02 […data…] 0xf7
“Ignore”	“Ignore”	All incoming messages

The ID for a specific device manufacturer can be found here: https://www.midi.org/specifications-old/item/manufacturer-id-numbers

The received data from a SysEx message can be sent to an outgoing OSC String message. Therefore select “String (as calculated)” as data type and all data from the SysEx message (excluding the manufacturer ID) will be interpreted as ASCII text and sent in the OSC message. But you also may enter a calculation rule into the OSC data field:

OSC data field	**OSC String being sent for SysEx message “testString”**
empty	testString
[b5..]	String
[b1..4]	test
[b2]	e

It also is possible to translate an OSC message to a SysEx message. To do so you should specify the manufacturer ID you wish at the OSC->MIDI table “byte 1” (no extended manufacturer ID functionality here). At “byte 2” you may select a fixed value (one byte payload only then) or choose from the usual [d] and [a] options. One example use case could be creating a row that receives only OSC string messages of a type and transmit the received string as SysEx payload.

Other Options

File Menu

“Import Settings” will load all settings (like input, outputs, matrix and filtering) from a file. “Export Settings” will save such a file to the disk. “Reset Settings” restores all these settings to defaults.

“Import MIDI Translations” will load all OSC-MIDI and MIDI-OSC translations from a file. “Export Settings” will save such a file to the disk. “Reset Settings” resets the translation tables.

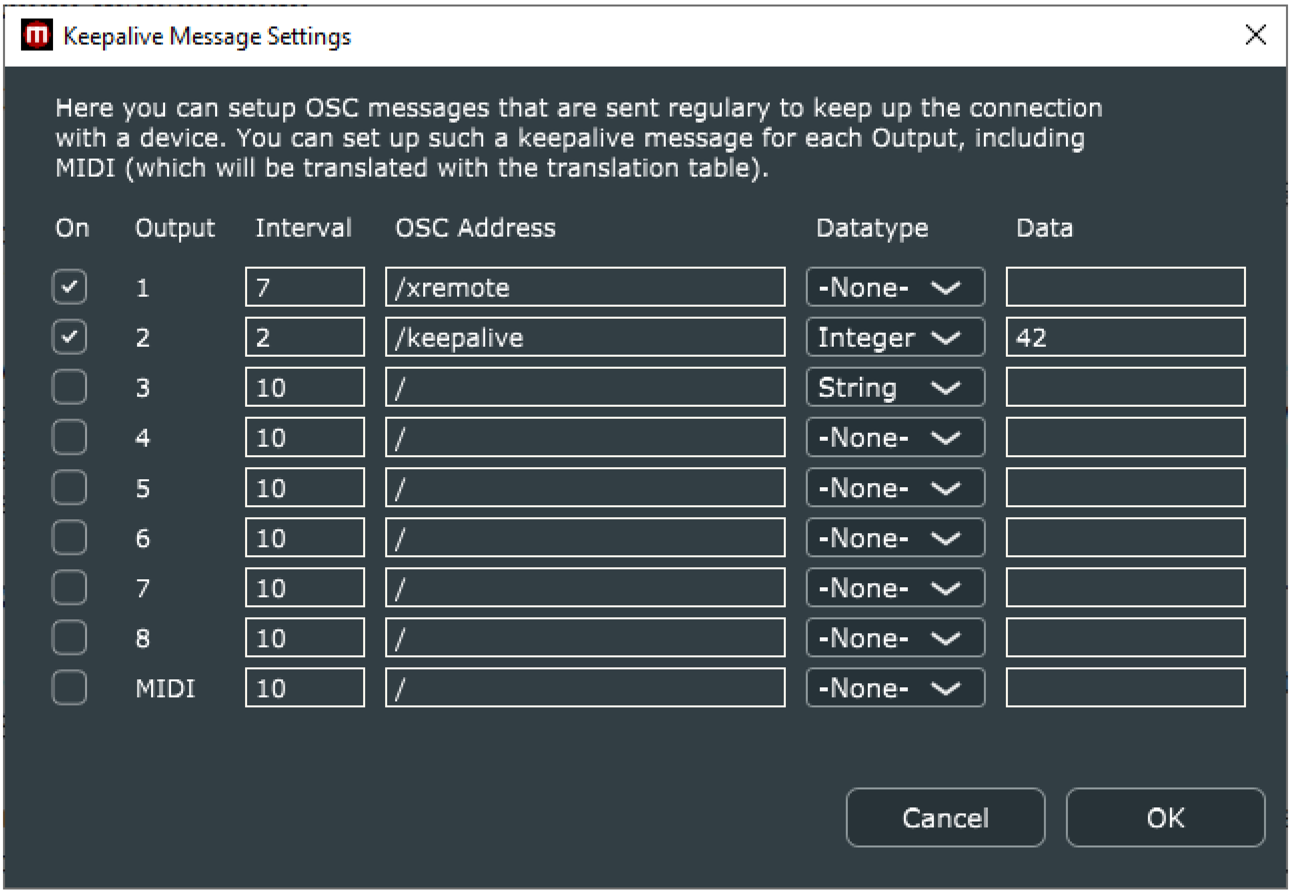
“Hide in Tray” will hide the program in the system tray but keep it running in the background.

Settings Menu

“Show my IP Address” will show a list of local IP addresses assigned to your computers.

Keepalive Messaging

Some OSC devices require periodic messages sent to them, to keep up the OSC connection, so called “keepalive” messages. To support such devices, OSC Matrix can generate individual keepalive messages for each output. You can find the accompanying settings in the settings menu.



Here you can setup OSC messages that are sent regulary to keep up the connection with a device. You can set up such a keepalive message for each Output, including MIDI (which will be translated with the translation table).

On	Output	Interval	OSC Address	Datatype	Data
☑	1	7	/xremote	-None-	
☑	2	2	/keepalive	Integer	42
□	3	10	/	String	
□	4	10	/	-None-	
□	5	10	/	-None-	
□	6	10	/	-None-	
□	7	10	/	-None-	
□	8	10	/	-None-	
□	MIDI	10	/	-None-	

For each output such a message can be activated and setup in terms of transmission interval (in seconds), OSC address and payload. The MIDI keepalive message again is translated from OSC via the OSC->MIDI translation table.

Command line Options

Some options can be passed to the software via command line on start:

Command	Description
hide	Hides the program in system tray by default

Tips

MIDI Loopback

If you are looking to send MIDI from a software on your PC to OSC Matrix on the same PC (or the other way around), you should have a look at “loopMidi” by Tobias Erichsen, which is a “virtual MIDI cable” that is free to use: http://www.tobias-erichsen.de/software/loopmidi.html

Questions?

Feel free to ask! Just write a mail to info@eleton-audio.de