

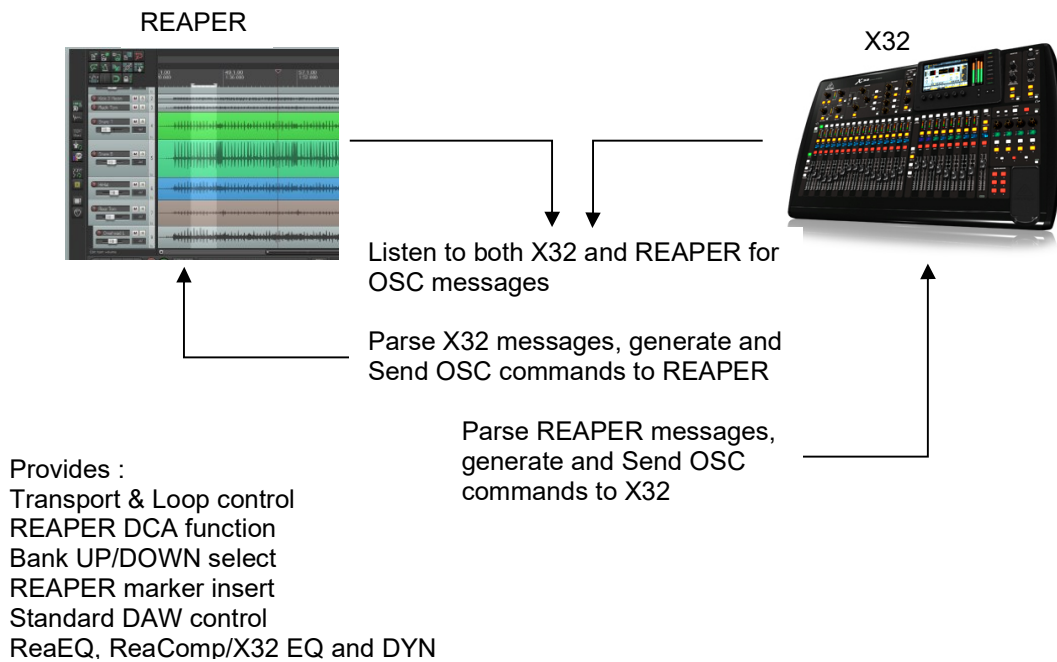
# X32Reaper

V2.84

A 2-way communication & parser to enable OSC messages to control one from the other.

## Principle:

The program opens two UDP communication streams and listens for UDP messages from both systems. X32Reaper acts as a proxy; upon receiving a message from one device, specific parsing & formatting takes place to generate OSC commands the other device can understand.



There are several steps to be carefully installed before X32↔REAPER controls can take place, so make sure you read this documentation first. You will also want to choose between the terminal and Windows-GUI versions of the program:

- The terminal version is aimed at OSX, Linux and Windows and runs as a command line program. Program parameters are setup from a resource file located in the program directory. The parameters are shown at startup but cannot be interactively modified. Also, the connections with REAPER and X32 are immediately established. To terminate the program, just hit Ctrl-C.
- The Windows-GUI version reads and displays parameters from a resource file. The user can modify these interactively and save them for future use. Parameter values are read at startup or when the connection is established with X32, and saved back to the resource file when the programs quits. Users can enable/disable communication between X32 and REAPER at will by pressing on the "Connect" button.

## X32Reaper

In both versions, the necessary parameters are saved in a resource file. The decision was made to mimic the REAPER configuration panel to ease setting values for X32Reaper. Below an example of resource file:

```
387 350 0 0 0 -1 -1 1
192.168.1.62
192.168.1.70
10025
10027
0 0 0 0 1 1 1 2
1 32 22 40 41 48 49 64 65 72 1
73 75
0 0
76 76
0 0
77 82
83 84
0 0
0 0
9 10 12 0 16
```

Detailed explanations on numbers above are provided in Appendix at the end of this document.

One thing to remember is that for REAPER, the X32Reaper program is the “Device” and therefore the IP of the system running X32Reaper should be used as HOST IP parameter in REAPER.

### REAPER setup:

Before anything, make sure you have installed the latest version of REAPER and completed your installation with the latest version of REAPER SWS extension package (<http://www.sws-extension.org/>).

A REAPER OSC configuration file must be installed:

In REAPER: chose **Options->Preferences**. In the Preferences panel, go to **Control Surfaces**. In the Control Surfaces panel on the right, click on “Add” to add a new Control surface and chose **OSC (Open Sound Control)**. In the Control Surfaces Settings panel, give a name to your device e.g. X32, and set a Pattern config matching a file name you will create, containing OSC REAPER methods, which corresponds to your OSC control file, for example “X32.ReaperOSC” in the REAPER OSC config directory.

X32.ReaperOSC is a subset of the Default.ReaperOSC file which has a lot more capabilities than the current X32Reaper implementation. An example of configuration setup and OSC config file are given below.

REAPER tracks corresponding to Input, Auxin, and FxRtn should include a ReaEQ effect as REAPER FX#1 (default value, changeable through the .inifile) in order to provide REAPER and X32 EQ controls, and a ReaCMP effect as REAPER FX#2 (default value, changeable through the .inifile) in order to provide REAPER and X32 COMP controls,. Also, it is a good idea to setup the default values of FX#1 and FX#2 as a preset that corresponds to what X32 can support by default: 3 band EQ, and a High Shelf control. This will ensure correct and comprehensive control of EQ between both platforms.

## X32Reaper

### OSC Patterns:

The figure next page displays the file you must install as the X32 OSC pattern file in REAPER.

Follow the instructions in REAPER to do so. In the REAPER→Options→Preferences→Control Surfaces panel, select ADD and in the new panel that opens, select OSC. A new Control Surface Settings panel appears, in which you can enter the different options. Give a name to your configuration and select the “open config directory” option in the Pattern Config drop menu. You can then create a new file with the information listed below.

Make sure you adjust DEVICE\_TRACK\_COUNT to the number of tracks your REAPER session actually use; a lower value will prevent some REAPER tracks to control their respective X32 Channel equivalent, and a higher value will unnecessarily use computing and REAPER resources.

```
#
# Behringer X32 Reaper/OSC pattern config file, Automation or Surface Control file
#
# © Patrick-Gilles Maillot, 2015
# Use with X32Reaper version 1.4 or higher #
# Reaper-> X32:
#
REAPER_TRACK_FOLLOWS REAPER
DEVICE_TRACK_FOLLOWS DEVICE
DEVICE_FX_FOLLOWS DEVICE

# X32 has 32 inputs, 6 aux, 2 USB, 8FXrtn, 16 buses, 8 DCA, 1 Master
DEVICE_TRACK_COUNT 88
DEVICE_SEND_COUNT 16
DEVICE_RECEIVE_COUNT 16
# General:
# Channel 1-32
TRACK_NAME s/track/@/name
TRACK_VOLUME n/track/@/volume
TRACK_PAN n/track/@/pan
TRACK_MUTE b/track/@/mute
TRACK_SOLO b/track/@/solo
TRACK_SELECT b/track/@/select
# Main Stereo (mute, solo and select are not supported by REAPER OSC)
MASTER_VOLUME n/master/volume
MASTER_PAN n/master/pan
# Bus Sends 1-16
TRACK_SEND_NAME s/track/@/send/@/name
TRACK_SEND_VOLUME n/track/@/send/@/volume
TRACK_SEND_PAN n/track/@/send/@/pan
# Transport
RECORD t/record
REPEAT t/repeat
STOP t/stop
PLAY t/play
PAUSE t/pause
REWIND b/rewind
FORWARD b/forward
REWIND_FORWARD_BYMARKER t/bymarker
ACTION i/action t/action/@
# Fx
FX_OPEN_UI b/track/@/fx/@/openui
FX_BYPASS b/track/@/fx/@/bypass
FX_NAME s/track/@/fx/@/name
FX_NUMBER s/track/@/fx/@/number/str
FX_PRESET s/track/@/fx/@/preset
FX_PARAM_VALUE n/track/@/fx/@/fxparam/@/value
FX_PARAM_NAME s/fx/@/fxparam/@/name
```

#### X32.ReaperOSC:

DEVICE\_TRACK\_COUNT should match the actual number of REAPER tracks of your project

## X32Reaper

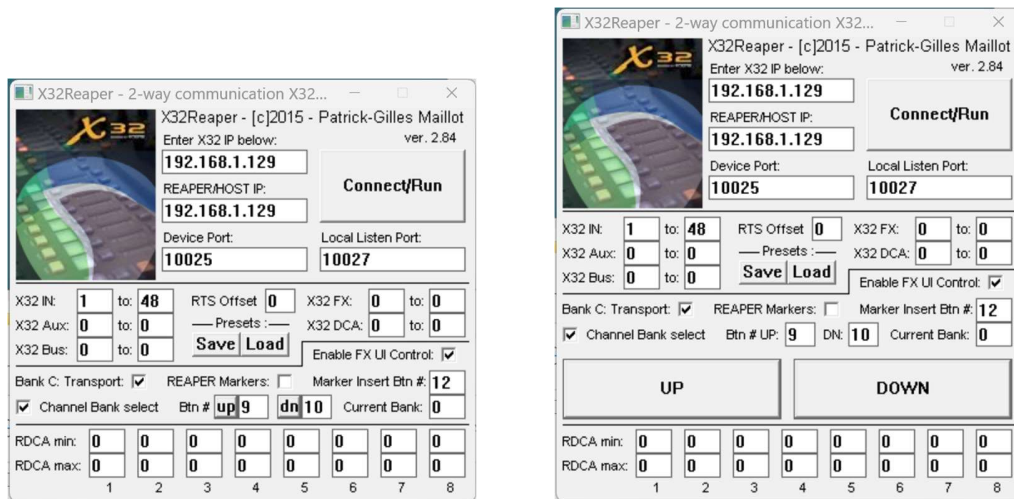
### Launching the program.

The Windows-GUI version of X32Reaper opens a window and displays a panel with parameters used for initializing the program and communicating with REAPER and X32.

*HOST IP*, *REAPER Device Port* and *Local Listen Port* can be modified if needed. The *X32 IP* address should match the address used by your X32 system. Port 10023 is reserved for X32 communications.

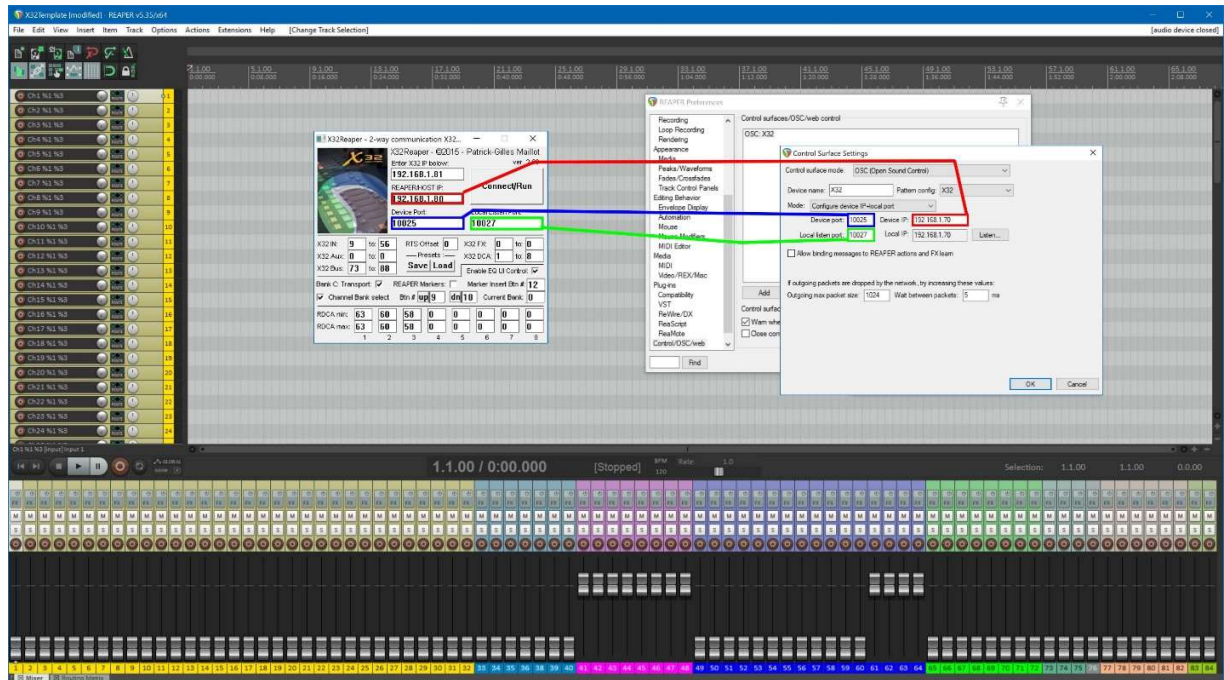
The program initializes and runs as soon as the **“Connect/Run”** button is engaged and displays **“Running...”** if the X32 has been found and communication with it is established, and the program was able to bind to REAPER.

There are two options for the UI as shown below, depending on the value of a variable in the .X32Reaper.ini configuration file (see later in this document)



At that point, and with everything correctly configured on you system and your X32, you will be able to play your REAPER files through the X32 channels + main/ST, while controlling REAPER faders from the X32.

# X32Reaper

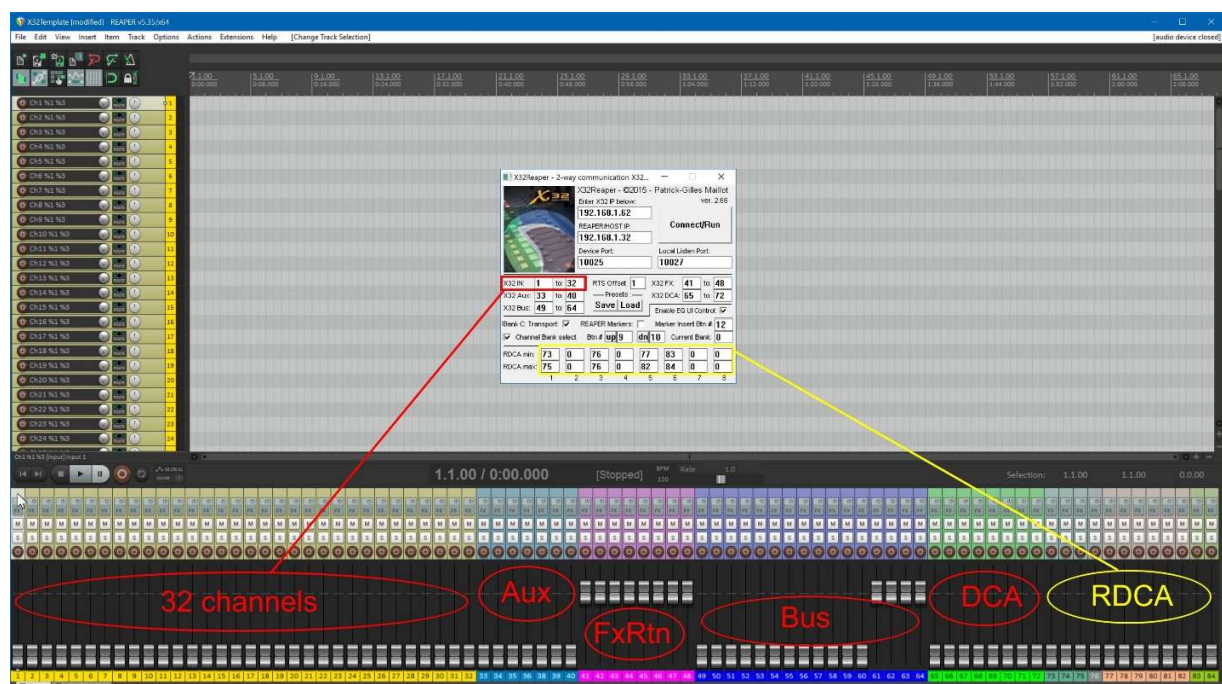


## X32Reaper

In the REAPER Control Surface Settings panel above. The Outgoing max packet size should be left at 1024. I have found that a value of 5ms for the Wait time between packets parameter helped in getting all data to be correctly transferred with no dropped UDP packet, especially when using the X32Reaper **Bank Channel select** flag (see later in this document).

Unless X32Reaper runs on a different computer, Device IP and Host IP will be identical in the REAPER Control Surface Settings panel. X32Reaper acts as a proxy between the X32 and the host, therefore it is the “Device” for REAPER.

It is possible to restrict the control in one or the other direction by unselecting the respective Device Port or Local Listen Port box in the REAPER Control Surface Settings panel. When Device Port is selected, the data received by REAPER (after conversion of the X32 data) can be displayed while X32 controls are used. You may need to authorize your system’s firewall to open the communication ports for the program to work correctly.



In the X32Reaper window, REAPER Transport control from X32, using knobs and buttons from X32 User bank C is enabled or not by setting the option in the respective checkmark.

There are 5 sets of two variables to set in the X32Reaper panel in order to tell X32Reaper which REAPER tracks correspond to physically available X32 “channels”. The X32 supports up to 32 Input Channels, 8 Aux Inputs, 8 FX Returns, 16 Bus Sends, and 8 DCA Groups. For each section, two values set the first and last REAPER track assigned to the respective section.

For example, assuming your REAPER project uses 8 REAPER tracks (1 to 8) for MIDI, controlled only with REAPER, and you would like to have 8 more tracks for audio, controlled by REAPER and X32 (tracks 9 to 16). Assuming also you would need REAPER tracks for 4 buses (tracks 17 to 20). You would then set values 9 and 16 into the two boxes following X32 IN, and 17 and 20 in the two boxes following X32 Bus. The other boxes should be set to 0.

This enables X32Reaper to correctly and strictly map X32 channels to REAPER tracks needed by your project. It also enables you to use additional programs managing the X32 channels not needed by your REAPER project (such as lighting management using X32 faders).

## Keyboard Shortcuts

Ver 2.80 and beyond of X32Reaper comes with keyboard shortcuts for all button-like or check-mark GUI function; The following tables gives the respective assignments of 1-key keyboard shortcuts to these:

Connect	C
Load Preset	L
Save Preset	S
Bank C Transport	T
Bank C REAPER Markers	M
Enable FX UI Control	F
Channel Bank Select	B
Channel Bank UP	U
Channel Bank DOWN	D

You can therefore use your mouse and click on a button to activate a functionality, or (ensuring the X32Reaper WINDOW is the top Window or the one receiving Windows' notifications) use your keyboard to type in the 1-key shortcut to perform the same action; Note that you can also program an external device such as Stream Deck to launch the keyboard shortcuts, and as a result launch the respective actions.

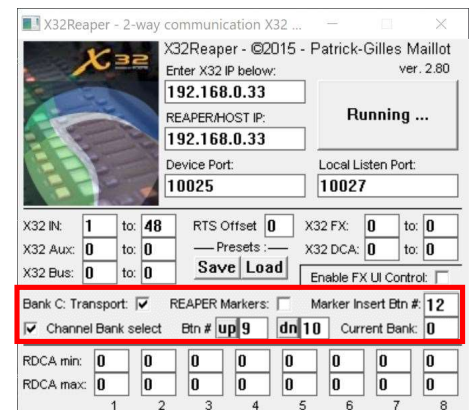
Along with shortcuts and user actions, the main GUI title bar will mention the current action taking place, helping programs such as NVDA to echo X32Reaper state changes.



## X32Reaper Features

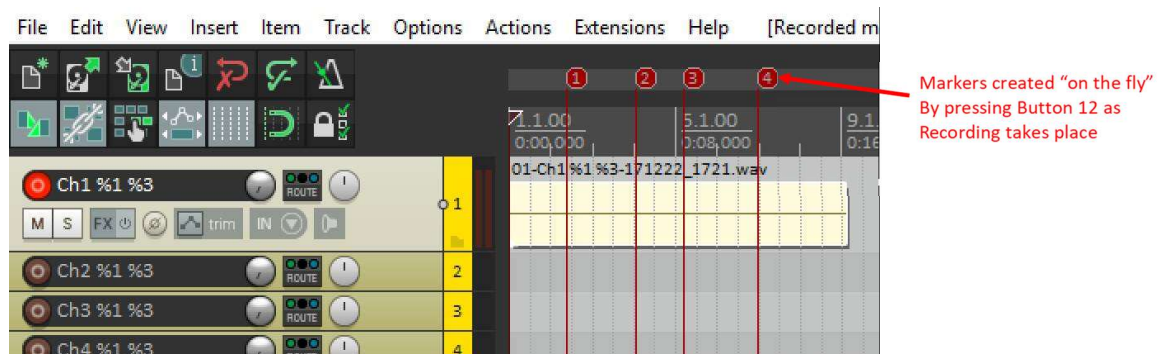
### Bank C flags

In the X32Reaper control panel, the “**Transport**” flag is used to enable or disable the use of the X32 **User Assign Bank C** dials and buttons to manage transport within REAPER. This is detailed later in this document. The “**REAPER Markers**” flag is exclusive to the “**Transport**” flag and enables inserting Markers in your REAPER recording or playback session on the fly, directly from a user selected bank C button; Finally, the “**Channel Bank select**” enables using two bank C buttons to move up and down in virtual channel banks, expanding the capabilities of the X32.



### REAPER Markers flag and Marker Insert Btn value

This flag enables the user to directly use one of the buttons of Bank C as an Insert Marker command for REAPER. This applies while recording or paly back. This flag is exclusive with Transport (i.e. you cannot set both simultaneously). The button value can be chosen by the user [values 5 to 12] but should be different from **Btn # UP** and **Btn # DN** used for **Channel Bank Select below**. Pushing the **Marker Insert Btn** will insert markers in your REAPER session (see below for an example, with 4 inserted markers):



### Channel Bank select flag, Current Bank value, and Bank size

The “**Channel Bank select**” flag is used to enable or disable the use of logical banks for selecting REAPER tracks in groups of 32, using the 32 input channel strips of the X32, which is then **ONLY** used as a control surface.

Indeed, if you have more than 32 REAPER input tracks (not taking into account any Aux, FxRtn, Bus, DCA track and assuming you don't make use of REAPER DCA – see later in this document), mixing should be done by the REAPER software. You can then use two of the transport section buttons to move up and down banks of inputs in your REAPER tracks layout, the current bank number being echoed in the “**Current Bank**” value. Additionally, there are two small buttons in the Window UI that enable you to select bank up or bank down that can be used instead of the X32 designated buttons. This can be useful for X32Rack users. The size of a bank is set at init time, by reading the respective value in the .X32Reaper.ini file. It is recommended (for ease of use) to set a bank size to a multiple of 8, and up to 32 (physical limit set by X32 channels) to optimize the use of X32 routing capabilities. Values of 8 and 16 will be the easier to use as the will not require you to also fiddle with the X32 physical channel bank select at the left of the console.

When switching up or down from one REAPER bank to the other, the number of X32 channel strips affected by the **Bank Size** will be updated to reflect the settings they have at the time of the switch,



either from a saved state if no REAPER change took place, or from their most recent REAPER settings. This applies to **fader, pan, solo, mute, scribble text, icon** and **color**, and the **mixbus sends**. The X32 **selected** channel strip is also reflected to match the respective REAPER track, in the current bank.

Using a **Bank Size** smaller than the actual number of X32 channels offers a large palette of routing options between what portion of the X32 desk will be used as a control surface for REAPER and what part of the X32 will be used to simultaneously manage audio (if that's your choice) using X32 routing to set what audio is managed by X32. Note that **Bank Size** also limits the number of X32 active channel strips; i.e. only the X32 channels less or equal to **Bank Size** are capable of interacting with X32Reaper, thus affecting REAPER tracks (directly or via banks).

If **"Transport"** is also set, the values for the two X32 buttons used this function are set as shown in the Transport panels depicted at the end of this document. If the **"Transport"** is not set, the two values can be chosen by the user between 5 to 12.

### X32 Input Channels [01-32]

With proper settings of values for the first and last REAPER track matching **"X32 IN"** in the dedicated box in the X32Reaper panel, X32Reaper enables managing and controlling **volume, panoramic, mute, solo, select**, and **scribble name** for each channel. **Scribble icon** and **color** can be set from the REAPER track name (see Notes below).

The REAPER MASTER track matches the **MAIN/ST** of X32 for **volume, pan** and **select**. Mute and solo have no effect as they are not associated to a real X32 input channel and REAPER does not send OSC data for these. You can nevertheless use MAIN solo and MAIN mute in the X32→REAPER direction, but care should be taken not to end-up with multiple REAPER channels selected at once (which is possible in REAPER, but not for the X32).

### X32 AUX IN/USB [01-08], and FX RETURNS [01-08]

Similar to X32 Input channels, these will map the same functionality: **level, pan, mute, solo, select**, and **scribble (name, icon, color)**. Make sure the correct values are set for the first and last REAPER track matching AuxIN/USB or FX Returns in the dedicated boxes in the X32Reaper panel. X32Reaper enables controlling X32 AUX IN 01-08 and FX RETURNS 01-08.

### X32 Bus and Channel Bus Sends [01-16]

X32Reaper enables controlling X32 Bus 01-16. REAPER Track sends levels can be controlled from X32 by modifying the Bus Sends of the X32 Channel respective of the REAPER Track (from the Channel strip BUS SEND section, or from the main faders when "Sends on Faders" is active). Vice-versa, modifying a REAPER Track bus send level will update the corresponding X32 Bus Send for the respective X32 Channel.

Bus Masters are also controllable from REAPER and REAPER Bus Tracks can be controlled from X32: At present time, the *X32.ReaperOSC* file and the program enable managing and controlling **volume, panoramic, mute, solo** and **select** for each Bus Master channel. As for standard Channels, X32 Bus Channel **Scribbles (name, icon, color)** can be updated from REAPER by changing REAPER Bus track names.

### REAPER Track Send [RTS] Offset [0...n]

REAPER treats "all" tracks the same way or pretty much. The normal expectation is when setting Track Sends to given REAPER track numbers, they would correspond to their respective X32 tracks. When also assigning a HW output to the REAPER track, the REAPER Track Sends [RTS] get incremented by an offset. This setting enables to compensate for such situations.

X32Reaper considers that when using this feature, typical setups will have the same number of HW

## X32Reaper

outputs for all Tracks, resulting in a single offset value applicable to all Track Sends.

### X32Group DCA [1-8]

In order to use the X32 DCA faders, it is necessary to create 8 REAPER Tracks that will be dedicated to control and receive data from the X32 DCA. Since DCA does not carry any audio signal, there is no need for any audio capability to be associated with these tracks in REAPER; nevertheless DCA automation is made possible by enabling automation record in REAPER. As for all other tracks controlled via X32Reaper, **fader movements**, **mute** and **solo** are managed in both directions; **Scribbles** have the same restriction as for other tracks due to a REAPER limitation. **pan**, and **select** do not operate for DCA groups.

### X32Reaper Presets

X32Reaper enables saving and loading the values of the lower part of the X32Reaper panel as user "Presets" for saving and setting the low and high values of REAPER tracks associated with X32 channels and the chosen option for transport. Click on the appropriate button (Save or Load) to use preset features. User Presets can be loaded and saved at any time; no need to be in "Running" state.

Upon loading an X32Reaper Preset, the preset name will be reflected in the X32Reaper Window title bar.

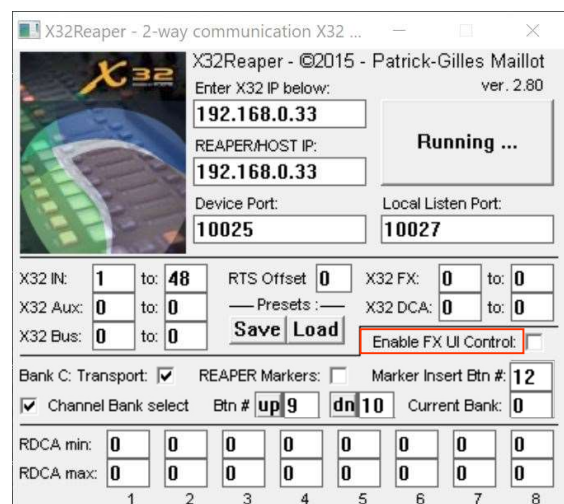
### RDCA Min/Max

X32Reaper offers a kind of DCA for managing large sets of REAPER tracks. The idea is that you have a number of REAPER tracks (typically beyond the X32 capacity) you have REAPER pre-mix to a REAPER mix bus or a set of REAPER tracks. This REAPER mix bus or set of tracks can be controlled by an X32 DCA strip (fader and MUTE). There are 8 of them enabling you to manage an arbitrary number of REAPER sub-mixes.

Note that it does not prevent you to still manage X32 internal DCA assigned channels, which will also be controlled by DCA changes you make. This affects the X32 DCA → REAPER direction; individual volume changes to any REAPER track assigned to a REAPER DCA track will affect the X32 DCA and its respective REAPER DCA track.

### FX Controls

Every X32 input (even virtual ones if banks are enabled/used), FxRtn, Auxin channel strip has the possibility to control its REAPER's respective track EQ. Bus and X32 channels can also control some of their respective Dynamics (compressor) parameters. Vice-versa, controlling the ReaEQ or a subset of ReaComp parameters for a REAPER track which has an equivalent mapped to X32 channels, will also change the X32 EQ or DYN settings for that respective track. If the X32Reaper "**Enable FX UI control**" flag is set, opening the FX UI of a REAPER track -if it is also the currently selected channel on X32- will bring up the X32 EQ or X32 DYN screen for that channel. Similarly, bringing up the FX screen of the current channel on X32 will open the REAPER respective UI on your DAW software.



X32Reaper expects \*all\* tracks to have the same effects present/loaded in REAPER tracks, because REAPER FX order and indexing (#1, #2...) is specified for all tracks in the project and can be set in the .X32Reaper.ini file. It is recommended to use REAPER templates to set REAPER tracks, so they are

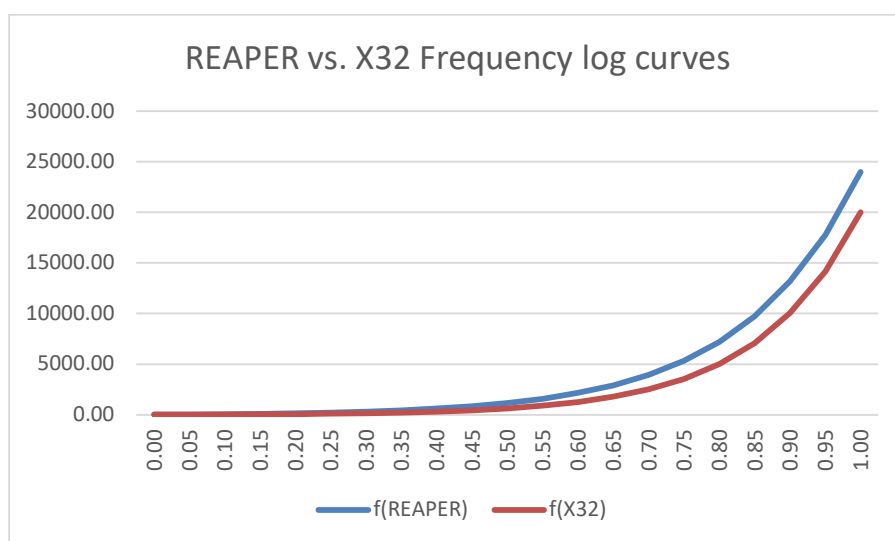
homogenous within your REAPER project. When using track associated to X32 bus channels, the EQ effect will not be taken into account, but the Compressor/Dyn one will.

### Constraints on REAPER templates and Settings

REAPER does not make any particular difference between input tracks, submix tracks or DCA tracks. Due to obvious hardware constraints, X32 does. To achieve proper translation from REAPER to X32, REAPER tracks must be grouped in disjointed groups but consecutive numbers, not going beyond the maximum number of respective channels supported by the X32 hardware (i.e. no more than 32 consecutive audio input, 8 consecutive Aux, 8 consecutive FX returns, 8 consecutive DCA, and 16 bus consecutive REAPER tracks). With the exception of REAPER tracks assigned to REAPER DCA tracks; all of these can be beyond X32 physical limitations, and managed through X32 DCA strips for REAPER mixing; would you want the mix of these tracks to be done within the X32, the sub mix tracks must then obviously appear within X32 physical channel value ranges.

In REAPER preferences, you should set in the “*Track Control Panels*” window the volume fader range to “-90dB to +10dB” (vs. the default -72dB to +12dB); this way REAPER sliders progression better match the pseudo log curve of X32 faders.

EQ curves and parameters mapping: X32 and REAPER have different frequency bandwidth spans, 20-20k for X32 and 20-24k for REAPER, furthermore the curves do not follow the same logarithm progressions slopes. X32Reaper adapts mapping from one to the other. Gain and Quality curves are left “as-is” with no conversion, so results can be slightly different. It is also advised to set REAPER ReaEQ parameters to match X32 band types and default values. This can easily be achieved through the use of FX presets and REAPER templates.



## Notes and Special cases

- You should first run X32Reaper, connect to X32, select/load an X32Reaper preset if applicable, and then load the REAPER OSC preferences or start REAPER; all REAPER settings for the project will automatically be synchronized with X32.
- While X32Reaper tests the different settings selected by the user, it is your responsibility to ensure the number of X32 channels does not exceed X32 capacity, or unexpected results may occur. Indeed, the program allocates dynamic memory depending on user settings; If these are wrong, X32 may receive bogus data and will most likely behave in unpredictable ways.
- X32 **MUTE GROUPS** will control multiple X32 and REAPER mutes at once, as expected.
- X32 **CLEAR SOLO** will cancel all solo on both X32 and REAPER, as expected.
- X32 **Linked** channels will also be linked in REAPER; vice-versa, grouped faders/tracks will be reflected on X32. When recording automation with REAPER, the linked channels should be managed from REAPER (see Track grouping) and not using the X32 LINK feature. Although the X32 LINK feature is great in live environments, it creates issues as no timestamp/bundle are used to notify multiple channels settings being changed at once, and this can create jaggy faders or catch 22 issues under certain conditions with automation being used in REAPER.
- X32 **Scribbles names** can be updated from REAPER by changing REAPER track names. The other way around does not work (although it should) but it's not very practical to change channel names from the X32. X32 **Scribbles icons** and **colors** can be updated from REAPER only, using the following syntax for REAPER track names: <name>[ %icon[ %color]]. (spaces are optional). For example, entering REAPER track name: "**IN 1 %21 %6**" will generate an X32 **cyan** scribble, with a **guitar** icon, and a scribble name "**IN 1**". The first line of the scribble will always reflect the X32 channel number (cannot be changed). The tables in appendix give a list of icons and colors available with the X32.
- REAPER **Master** Track:
  - Select, Mute and Solo on X32 MAIN will be reflected on REAPER Master track.
  - Select, Mute and Solo on REAPER Master track will not be reflected on X32. This is due to a limitation of REAPER OSC.
  - Muting the MAIN on X32 will automatically/force select X32 MAIN / REAPER Master track. This is due to a limitation of REAPER OSC.
- **Important note on assigning REAPER bus sends [RTS]:** when creating or assigning HW controls to manage REAPER bus sends, it seems that the bus numbers are assigned in logical order, rather than using the numerical value of the REAPER bus they are assigned to; for example, assigning as the first bus send ReaRoute #4 to track #1 will result in a send bus #1 being assigned to track #1. As this is not the expected behavior for the X32 console, it is necessary to assign bus numbers in increasing order, starting with bus #1, so the REAPER send bus numbers correspond to the X32 send bus values. A good practice is to create templates with all 16 send bus values systematically created, so they all correspond to X32 bus 01 to bus 16, as expected.
- Changing settings during a work session: The proper way to X32Reaper to take into account new settings is to disconnect/connect; this ensures all settings from the X32Reaper GUI panel are read and taken into account, re-allocating internal tables as needed.
- Disabling and re-enabling **Channel Bank select** while X32Reaper is running with the flag initially enabled will set all X32 channel scribbles to black, no name, no icon. A confirmation message will warn you of the possible blanking of scribbles. All respective faders and channel bus sends will be set to -oo, pans to center, unmute, unsolo. Channel 1 will be selected. It is expected that you load

## X32Reaper

or reload your REAPER session for channel banks to get proper data from REAPER. Also, bank size is initialized upon reading the resource file, and cannot be changed during a session.

- Navigating through banks using the **Bank Up** and **Bank Down** buttons can put a high stress on communications between X32Reaper and the X32. If needed, a delay of a few ms can be added on the OSC messages during Bank change requests. See the detailed description of the .X32Reaper.ini file in.
- Two special flags in the X32Reaper.ini file (see appendix at end of this document) enable the filtering of commands sent to X32 following REAPER messages parsing and vice-versa. For each flag, a single integer (default value = -1) sets the commands that will be sent to

X32, following a parsing of a REAPER OSC message:

The following bits are defined:

REAPER TRACK pan	0x0001	fits all X32 pan commands
REAPER TRACK volume	0x0002	fits all X32 fader commands
REAPER TRACK name	0x0004	fits all X32 scribble name, icon, color commands
REAPER TRACK mute	0x0008	fits all X32 mute commands
REAPER TRACK select	0x0010	fits all X32 select commands
REAPER TRACK send	0x0020	fits all X32 mix send commands
REAPER TRACK solo	0x0040	fits all X32 solo commands
REAPER TRACK FX	0x0080	fits X32 EQ/DYN commands
REAPER MASTER pan	0x0100	fits X32 L/R pan commands
REAPER MASTER volume	0x0200	fits X32 L/R fader commands

For example, a value of 5 instead of the default -1 will enable only REAPER track pan and name related commands to reach X32. The other REAPER commands will still be parsed but will not translate to X32.

REAPER, following a parsing of an X32 OSC message:

The following bits are defined:

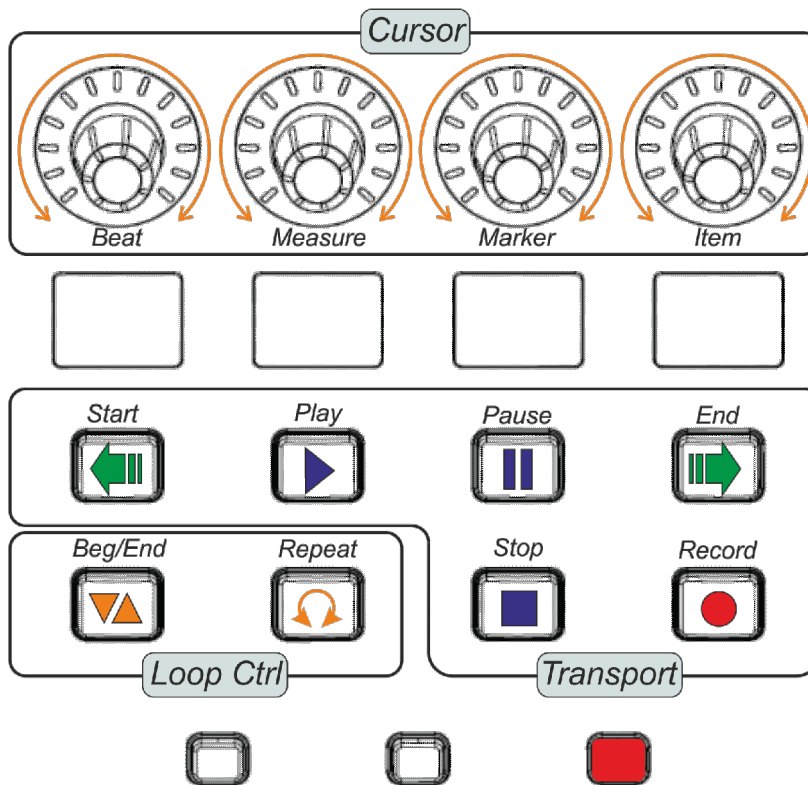
X32 pan	0x0001	fits REAPER tracks pan commands
X32 volume	0x0002	fits REAPER tracks fader commands
X32 name	0x0004	fits REAPER tracks scribble name, icon, color commands
X32 mute	0x0008	fits REAPER tracks mute commands
X32 select	0x0010	fits REAPER tracks select commands
X32 send	0x0020	fits REAPER tracks mix send commands
X32 solo	0x0040	fits REAPER tracks solo commands
X32 FX(EQ/DYN)	0x0080	fits REAPER tracks EQ/COMP commands
X32 Main pan	0x0100	fits REAPER master track pan commands
X32 Main volume	0x0200	fits REAPER master track fader commands

For example, a value of -65 instead of the default -1 will prevent X32 Solo buttons to affect REAPER. The X32 solo button will still be parsed, but the result will not be transmitted to REAPER. Similarly, a value of 15 will only transmit pan, volume, name and mute X32 actions to REAPER.

- By default, bank C (when used by the program) is set to color '0' (black). This can be changed in setting the respective flag in the X32Reaper.ini file (see at the end of this document) to the desired color index value.
- X32Reaper does not provide EQ control for REAPER tracks of type Bus/Busmix; this is because ReaEQ FX parameter OSC controls only support 16 parameters, and the 6 bands X32 supports for Busmix strips would require at least 18 parameters.

## Transport & Navigation

X32Reaper offers a **Transport & Navigation** capability for X32 standard console users, addressable with **User Controls Bank C**. Transport can be enabled or disabled by clicking on the respective checkbox. The picture below shows the implemented controls.



Navigation controls. Turn encoders right or left to move the REAPER cursor project accordingly. Movements can be made per Beat, Measure, Marker, or Item.

The scribbles will be set to black by default as no provision is offered by Behringer to program these.

The 8 buttons are assigned as shown. Loop Control and Transport buttons will light, depending on REAPER state.

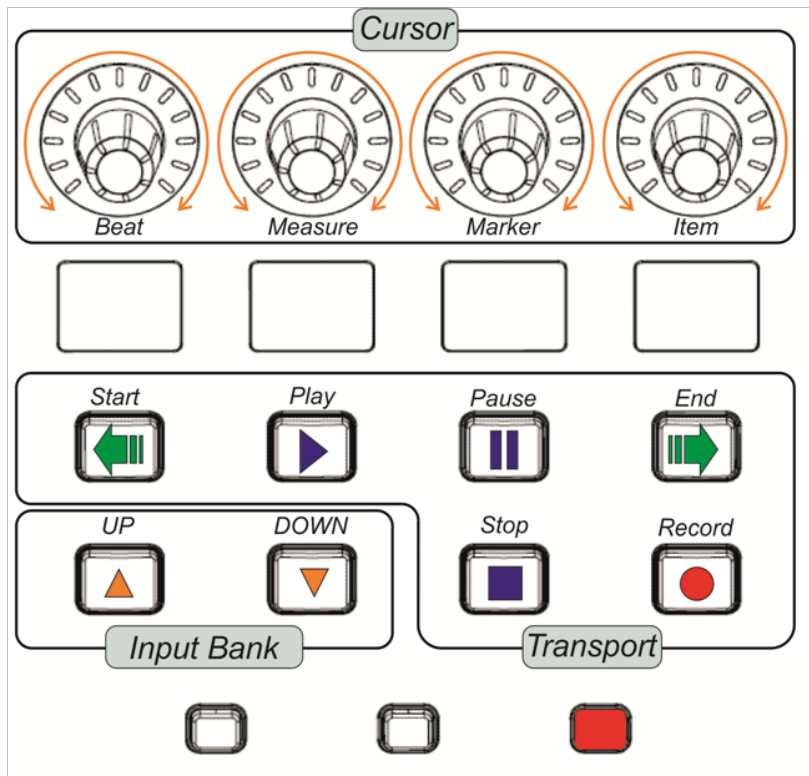
Make sure you have Bank C being selected!

Note, when in “Play” mode, encoders are still active, enabling a “jog style” navigation forward or backwards, by beat, measure, marker, or item.

If you selected the “**Channel Bank select**” flag in the X32Reaper control panel, the **User Controls Bank C**. Transport section is organized as shown below, with the “**Input Bank**” buttons assigned to manage REAPER tracks bank selection UP and DOWN, according to the bank size variable value set in the .X32Reaper.ini file. Pressing Bank UP and Bank DOWN will navigate through REAPER tracks as per bank changes requests, with the X32 respective channels reflecting the corresponding REAPER tracks. Again, it is a good idea to set bank size to a multiple of 8 in order to best user X32 routing capabilities; therefore, values of 8 and 16 are the best suited values for bank size; 24 and 32 are also possible values, but offers less ease of use as you’ll also have to manipulate X32 physical banks.

## X32Reaper

As you navigate through banks, the current bank number is displayed in the X32 control panel.



Navigation controls. Turn encoders right or left to move the REAPER cursor project accordingly. Movements can be made per Beat, Measure, Marker, or Item.

The scribbles will be set to black by default as no provision is offered by Behringer to program these.

The 8 buttons are assigned as shown. Transport buttons will light, depending on REAPER state.

Bank number is reflected in the X32Reaper control panel.

Make sure you have Bank C being selected!

### Arming REAPER tracks for recording:

X32Reaper supports/implement the REAPER "Toggle record arming for current (last touched) track" action, enabling arming or disarming a REAPER track. In order to do this, you press and release the Repeat (or DOWN) key while the Beg/End (or UP) key is pressed down.

### Enjoy!











































































Tested and validated with a Windows 10 based system running REAPER ver. 5.963 using Behringer USB/ASIO digital audio interface and a Behringer X32 Standard with FW 3.09.

I welcome feedback and ideas for additional features or controls which would make sense, without building too complex an environment. Musicians should spend time playing music, not programming! 😊



## Appendix – X32/M32 Icons

X32/M32 Icons are numbered 01...74 and shown in the table below.

					
01	02	03	04	05	06
					
07	08	09	10	11	12
					
13	14	15	16	17	18
					
19	20	21	22	23	24
					
25	26	27	28	29	30
					
31	32	33	34	35	36
					
37	38	39	40	41	42
					
43	44	45	46	47	48
					
49	50	51	52	53	54
					
55	56	57	58	59	60
					
61	62	63	64	65	66
					
67	68	69	70	71	72
					
73	74				

## Appendix – X32/M32 Colors

X32/M32 colors are numbered *0...15* and shown in the table below, (a trailing 'i' means "inverted"):

**OFF, RD, GN, YE, BL, MG, CY, WH, OFFi, RD*i*, GNi, YEi, BLi, MG*i*, CYi, WHi**

## Appendix – .X32Reaper.ini resource file

The **.X32Resource.ini** file contains data used in/by the program; it is read at program startup and is updated with the panel data and written when the program exists.

The file contains the following (values below are for example):

<b>397 339 0 0 0 -1 -1 0</b>	X32Reaper window size (X & Y)   verbose flag   ms between X32 bank up/down OSC messages   ms between generic X32 OSC messages   REAPER ->X32 command mask   X32->REAPER command mask   UP/DOWN UI Buttons size (0: tiny, 1: big)
<b>192.168.1.62</b>	X32 IP address
<b>192.168.1.32</b>	REAPER host IP address
<b>10025</b>	[REAPER] Send to Port
<b>10027</b>	[REAPER] Receive from Port
<b>0 0 0 0 1 1 1 2</b>	Transport   Channel Bank select   REAPER Markers   Bank C color   Enable FX UI control   Master_on (keep to 1)   ReaEQ index   ReaCMP index
<b>1 32 33 40 41 48 49 64 65 72 1</b>	Track   Aux   Fxrtn   Bus   DCA (min/max)   Bus sends offset
<b>73 75</b>	RDCA 1 min/max
<b>0 0</b>	RDCA 2 min/max
<b>76 76</b>	RDCA 3 min/max
<b>0 0</b>	RDCA 4 min/max
<b>77 82</b>	RDCA 5 min/max
<b>83 84</b>	RDCA 6 min/max
<b>0 0</b>	RDCA 7 min/max
<b>0 0</b>	RDCA 8 min/max
<b>9 10 12 0 16</b>	Bank Up button   Bank Down button   Marker Insert Btn   value of Bank (keep to 0)   bank size

The file can be edited with notepad++ for example.

### LIMITED WARRANTY AND DISCLAIMER

THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ANY AND ALL OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED. IN NO EVENT WILL THE AUTHOR BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, INCIDENTAL OR SPECIAL DAMAGES, INCLUDING ANY LOST PROFITS OR LOST TIME, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY PARTY.

Pictures © of their respective brands when applicable