# x32midiosc

X32midiosc is a “terminal/command line” application which send and receive OSC messages from a network port via UDP and send and receive MIDI packets. The data in both directions is converted from OSC to MIDI using Mackie Logic Control definitions.

This is done, by emulating six devices of Mackie Logic control on six midi channels.

Those six devices providing 8+8+8+8+8+9 control channels.

There is no bank switching implemented right now. So your project in the DAW has to match the X32 channel Layout (32 Input 16 Aux 1 Master)

### Installation

Just copy the provided binary “x32midiosc” to a location of your choice.

### Starting

### OSX

* Open a terminal on OSX.
* Change directory to the installation directory using “cd *pathtox32midioscdir*”
* Start the application by “./x32midiosc *ownport x32port x32ipaddress* ”  
  Example: ./x32midosc 10000 10023 192.168.1.2
* Move a fader on X32 and see if messages running trough the debug screen.
* If nothing happens. Check the Addresses and the firewall settings.

### Windows

* Make sure you have MIDI port Mapper like LoopBe30 with at least 12 Ports
* Open a DOS Box (the black screen) in Windows.
* Change directory to the directory where you extracted the ZIP file using   
  “cd *pathtox32midioscdir*”
* Get help by typing:  
   “x32midiosc –help”  
    
  x32midiosc Version 0.02 help

Invoking "x32midiosc":

x32midiosc [local port] [X32 port] [X32 IP] (use virtual ports; OSX only)

x32midiosc --list (Gives a list of available midi ports)

x32midiosc [local port] [X32 port] [X32 IP] [in1] .. [in6] [out1] .. [out6]

[inX] and [outx] MIDI port number x32midiosc read from and DAW writes to.

according to the 'x32midiosc --list' output

[in1] receiving MIDI data on this port will be mapped to CH01-08 on X32

[in2] receiving MIDI data on this port will be mapped to CH09-16 on X32

[in3] receiving MIDI data on this port will be mapped to CH17-24 on X32

[in4] receiving MIDI data on this port will be mapped to CH25-32 on X32

[in5] receiving MIDI data on this port will be mapped to Bus01-08 on X32

[in6] receiving MIDI data on this port will be mapped to Bus09-16 on X32

plus Master Fader on X32

[out1] CH01-08 on X32 will be sending MIDI data to this port

[out2] CH09-16 on X32 will be sending MIDI data to this port

[out3] CH17-24 on X32 will be sending MIDI data to this port

[out4] CH25-32 on X32 will be sending MIDI data to this port

[out5] BUS01-08 on X32 will be sending MIDI data to this port

[out6] BUS09-16 on X32 and Master Fader will be sending

MIDI data to this port

Sample: x32midiosc 10000 10023 192.168.1.2 0 1 2 3 4 5 6 7 8 9 10 11

* Get a MIDI interface list by typing:  
   “x32midiosc –list”

There are 16 MIDI input sources available.

Input Port 0 : '01. Internal MIDI'

Input Port 1 : '02. Internal MIDI'

Input Port 2 : '03. Internal MIDI'

Input Port 3 : '04. Internal MIDI'

Input Port 4 : '05. Internal MIDI'

Input Port 5 : '06. Internal MIDI'

Input Port 6 : '07. Internal MIDI'

Input Port 7 : '08. Internal MIDI'

Input Port 8 : '09. Internal MIDI'

Input Port 9 : '10. Internal MIDI'

Input Port 10 : '11. Internal MIDI'

Input Port 11 : '12. Internal MIDI'

Input Port 12 : '13. Internal MIDI'

Input Port 13 : '14. Internal MIDI'

Input Port 14 : '15. Internal MIDI'

Input Port 15 : '16. Internal MIDI'

There are 17 MIDI output ports available.

Output Port 0 : 'Microsoft GS Wavetable Synth'

Output Port 1 : '01. Internal MIDI'

Output Port 2 : '02. Internal MIDI'

Output Port 3 : '03. Internal MIDI'

Output Port 4 : '04. Internal MIDI'

Output Port 5 : '05. Internal MIDI'

Output Port 6 : '06. Internal MIDI'

Output Port 7 : '07. Internal MIDI'

Output Port 8 : '08. Internal MIDI'

Output Port 9 : '09. Internal MIDI'

Output Port 10 : '10. Internal MIDI'

Output Port 11 : '11. Internal MIDI'

Output Port 12 : '12. Internal MIDI'

Output Port 13 : '13. Internal MIDI'

Output Port 14 : '14. Internal MIDI'

Output Port 15 : '15. Internal MIDI'

Output Port 16 : '16. Internal MIDI'

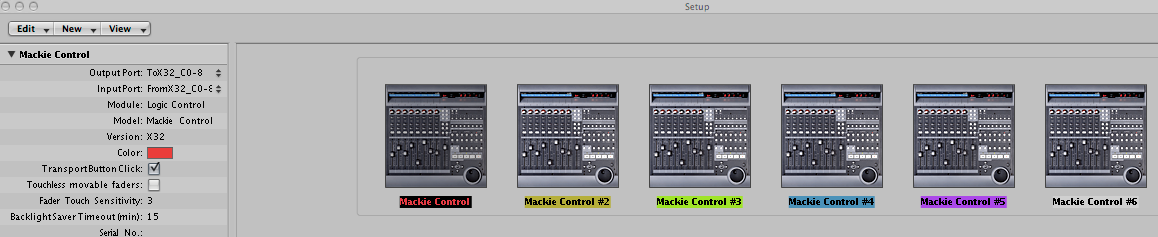
* Start the application by typing:  
   x32midiosc *ownport x32port x32ipaddress 0 1 2 3 4 5 6 7 8 9 10 11 12* ”  
  Example: ./x32midosc 10000 10023 192.168.1.2 6 7 8 9 10 11 1 2 3 4 5 6
* Move a fader on X32 and see if messages running through the debug screen.
* If nothing happens. Check the Addresses and the firewall settings.

### Ableton Live V8

* Press COMMAND key + “,“ and bring up midi setup.
* Assign all six Devices:  
    
  
* Prepare your Project on the X32 (32 Input 16 Aux 1 Master)
* I did not find a way to assign Control Surface 1 to channel 1-8 and so on. So you must define a “send a initialization sequence key” on the X32.  
    
  1. Select “set C” in user assignable section on the X32  
  2. Assign “Button 5; Midi Push; Channel 01; 0”  
  3. Press the Button ☺. This should adjust the controller banks.
* Have fun.

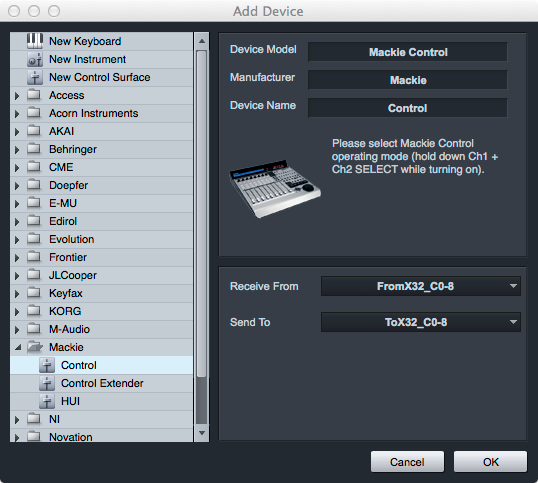
### Logic Pro 9

* Select from Menu: Preferences-Control Surfaces->Setup
* Select from Dialog Menu: New->Automatic Installation
* Start x32midiosc in the tree parameter version:  
   **./x32midiosc 10000 10023 192.168.1.2**
* Six Mackie Control devices show up automatic.



* Check the Output Port and Input Port assignments for every of the six added controllers.
* Have fun

### Presonus Studio One 2 setup

* Select Preferences -> external Devices
* Select “Add”
* Mackie -> Control Extender  
    
  
* Repeat 4 times by using the next midiosc interface for in and out
* Add Mackie-Control
* In the resulting list select each of the interfaces press “edit” and change the device name to 1-6.
* Select “Placement” and drag all devices in correct order “1-6” to “Group 1”.
* Have fun.

### Reaper OSX (nearly the same under Windows)

* Select from Menu: Preferences
* Scroll down to Plug-ins->Control Surfaces
* Click “Add”
* Select “Mackie Comtrol Universal”
* Select Midi Inputs and Outputs
* Adjust “Surface offset” (0,8,16,24,32,40)
* Press OK
* Do this six times and do not forget increasing the “Surface Offset” by 8
* Have fun

