

I-CUBE X



Max plug-ins v2.xx

I-CubeX Reference

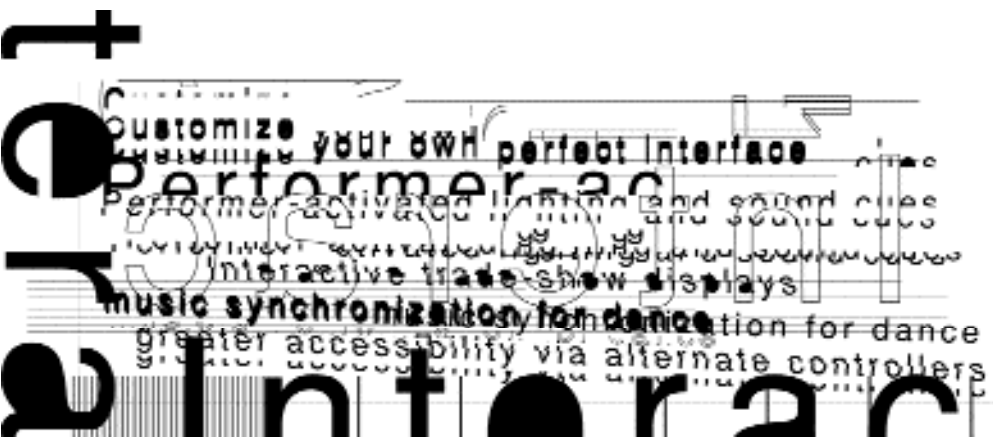
I-CubeX: The ultimate MIDI controller!

Infusion Systems Ltd.
2033 Avenue de Vendôme
Montreal, QC
Canada H4A 3M4

Tel: (514) 484 5850
Fax: (514) 484 5852

Email: info@infusionsystems.com
<http://www.infusionsystems.com>

16 January 2004
© 1997-2004 Infusion Systems



Max plug-ins v2.0x

for Max on MacOS and Windows

Host mode gives maximum control and functionality over the I-CubeX. Host mode's drawback is the fact that the Digitizer does not retain its configurations upon powerdown (they can be stored within a program like Max). In addition, the Digitizer MUST be connected to the computer at all times in order to configure it or obtain sensor data. The following text applies only to Max running on MacOS.

Max objects

I-CubeX ships with two Max objects - iCube and oCube objects (version 2.0). These objects support both PPC and 68k versions of Max 3.5 ad Max 4. With Max, the I-CubeX can be pushed to its sampling limit without slowdowns! These objects will also work with older versions of Max – tested with Max 2.52.

To learn how to use the iCube and oCube objects see the iCube.help / oCube.help patches, as well as the downloadable Max tutorials (available at the www.infusionsystems.com page). Use this section for reference.

Installing iCube and oCube objects

Copy the iCube and oCube objects (found in the I-CubeX objects folder) into your max-startup folder (see the Opcode Max manual for more info). Next, copy the iCube.help and oCube.help objects into your max-help folder. The iCube and oCube objects will be loaded the next time you run Max. When creating new objects, the iCube and oCube object will be located in the Control category.

Using the iCube (input) object

The iCube object is used to bring sensor control into the Max environment.

Inlets

The command list inlet (leftmost inlet) is used to program the I-Cube. See the iCube object command summary (as well as iCube.help) for a description of available commands. The MIDI inlet (rightmost inlet) MUST be connected to the midiin object.

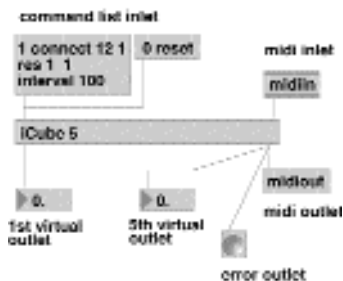


Fig. H-1: The iCube object in all its glory

Outlets

The number of outlets depends on how many virtual outputs have been allocated (see iCube x y command below). If 8 outputs have been allocated, then the first 8 outlets act as the first 8 virtual outputs. The 9th outlet is the Error indicator (which flashes whenever the system encounters an error, usually due to errors in the command list). The 10th (rightmost) outlet MUST be connected to a midiout object.

iCube Object Command Summary :

iCube x y where x specifies the number of virtual outlets to create and y indicates the Device ID of the Digitizer to be used. These settings are entered when a new iCube object is created but can be changed at any time. The range of x is 1 to 32. The range of y is 0 to 127. If only one parameter is passed, the DeviceID is assumed to be 0. If no parameters are passed, the number of outlets is 32, and the DeviceID is 0.

0 reset x resets the Digitizer's memory. The Digitizer acknowledges with "I-Cube Reset(x)". X indicates the Digitizer's Device ID. A reset command resets all commands which have been sent to the Digitizer. They have to be re-"banged".

x connect y connects sensor input x to virtual output y. You may connect several sensor inputs to virtual outputs by typing 1-4 connect 4 - 1 (sensor input 1 is mapped to virtual outlet 2, sensor input 2 is mapped to virtual outlet 3, and so on) or 20 - 29 connect 1 - 10 (connects sensor input 20 to virtual outlet 1, sensor input 21 to virtual outlet 2, and so on). Warning: typing 10 - 12 connect 3 - 4 yields an error (sensor input 12 can not connect to a corresponding outlet since input 10 is connected to outlet 3 & input 11 is connected to outlet 4).

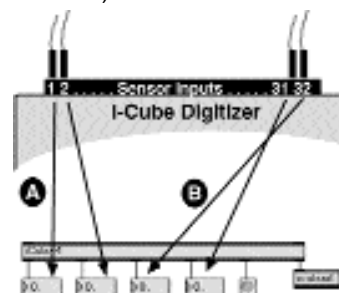


Fig. H-2: Two examples of the connect command: (A) 1-2 connect 1-2, (B) 3-4 connect 32-31

x res 0 (or 1) this command sets the sampling resolution for one or more virtual outputs. A setting of 0 sets the resolution to 7bit mode (lo-res) (input range of 0 to 127). A setting of 1 sets the resolution to 12bit mode (hi-res) (input range of 0 to 4095). This command will affect the inmin and inmax settings.

x on turns one or more virtual output(s) on. Example: You can turn virtual output 4 on by typing 4 on. You can turn sensors 4 through 10 on using only one command - 4 - 10 on. You will get an error if you try to turn a virtual output which has not been connected to a sensor (using the connect command above)

x off turns one or more virtual outputs off. Example: You can turn virtual output 4 off by typing 4 off. You can turn sensors 4 through 10 off using only one command - 4 - 10 off. (you can type 10 - 4 off, or 4 off 5 off 6 off ... 10 off to do the same thing). You will get an error if you try to turn an inactive virtual output off (one that has not been turned on with the on command above).

x inmin y this command acts as sensor input threshold for one or more virtual outputs (ex: 1 - 10 inmin 30). Input values below y will output as 0 (or the value set by the min command below). Use this command with sensors which exhibit a noticeable noise floor (where output values fluctuate). You can also use this command to zoom into a sensor input region of interest. You will get an error if you try to set inmin on a virtual output which has not been connected to a sensor (using the connect command). The inmin range is 0 to 127 (if the virtual output is in 7bit mode) or 4096 (if the virtual output is in 12 bit mode).

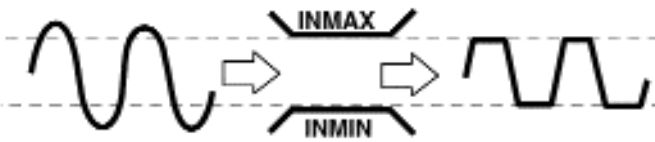


Fig. H-3: An example of limiting the sensor input signal using *inmin* and *inmax*

x inmax y this command acts as sensor input ceiling for one or more virtual outputs (ex: 1 - 10 inmax 99). Input values above y will output as 127 (or the value set by the max command - see below). Use this command with sensors which tend not to span the full input range (like the heat sensor). You can also use this command to zoom into a sensor input region of interest. You will get an error if you try to set inmax on a virtual output which has not been connected to a sensor (using the connect command) The inmax range is 0 to 127

(if the virtual output is in 7bit mode) or 4096 (if the virtual output is in 12 bit mode).

x min y this command sets the output range minimum of one or more virtual outputs (ex: 1 - 10 min 0). The virtual output will range from y to 127 (or the value set by the max command - see below). You will get an error if you try to set min on a virtual output which has not been connected to a sensor (using the connect command). The min range is +/- 100000.

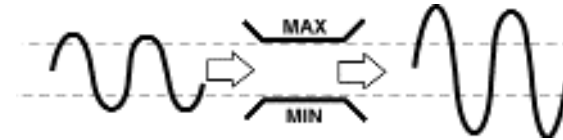


Fig. H-4: An example of amplifying the sensor input signal using *max* and *min*

x max y this command sets the output range maximum of one or more virtual outputs (ex: 1 - 10 max 1). The virtual output will range from 0 (or the value set by the min command - see below) to y. You will get an error if you try to set max on a virtual output which has not been connected to a sensor (using the connect command). The max range is +/- 100000.

x step y this command sets the number of virtual output steps (similar to sample and hold). You will get an error if you try to set step on a virtual output which has not been connected to a sensor (using the connect command). You can set the step for multiple virtual outputs by typing 1 - 10 step 3 (which sets the steps for virtual outputs 1 through 10 to 3 steps) for example.

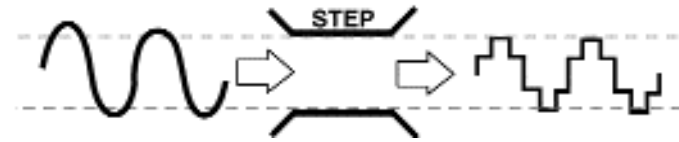


Fig. H-5: An example of changing the sensor input signal from continuous to stepped using *step*

x sample this command takes an instant snapshot of the sensor connected to virtual output x. You can sample more than one virtual output by typing 1 - 10 sample (which sets the steps for virtual outputs 1 through 10 to 3 steps) for example. Sample does not work on active virtual outputs (ones that have been turned on with the on command). You will get an error if you try to set sample on a virtual output which has not been connected to a sensor (using the connect command).

0 mute this command mutes and un-mutes all active virtual outputs.

0 print this command prints out the Digitizer's current configuration (the accumulation of all of the above commands).

0 interval x this command sets the Digitizer's sampling interval which affects all virtual outputs. Use this command very carefully. Setting a very small interval may cause system slowdowns in multi-sensor setups. The sampling interval range is 4ms to 16sec.

0 setdigid x sets the Digitizer's Sysex Device ID. X ranges from 0 to 126. This means you could use up to 126 Digitizers at once! This gives you a whopping 4064 sensor inputs (with 1016 digital outputs). [240, 125, <DeviceID>, 92, <new DeviceID>, 247]

0 hostmode switches the I-CubeX mode to Host mode (this can also be changed from the I-CubeX editors on both the Mac and PC). [240, 125, <DeviceID>, 90, 0, 247]

0 standalone switches the I-CubeX mode to Stand Alone mode (this can also be changed from the I-CubeX editors on both the Mac and PC). [240, 125, <DeviceID>, 90, 1, 247]

oCube object

The oCube object allows easy control of the Digitizer's 8 digital outputs.

Inlets

The 8 inlets are mapped to control the 8 digital outputs (banging a value of 0 turns the output off, while a positive, non-zero number turns the output on).

Outlets

The leftmost outlet acts as an error indicator (you may connect a bang object to it to indicate errors). The rightmost outlet MUST be connected to a midiout object.

See the oCube.help file for an interactive tutorial.

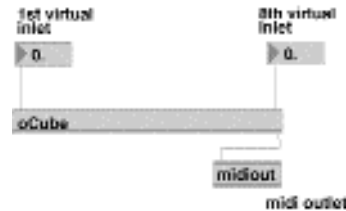


Fig. H-6: The oCube Object in its natural splendor



On PowerPC macs with Max and I-CubeX, warning do not switch between Max windows (and other windows) when running your iCube object at a sampling interval of 4ms with more than 10 sensor input channels active. You can also tweak the Configuration setting under the Options menu in Max.

iCube object error messages

The iCube object dedicates an outlet to error messages. In addition, a text description of the error is displayed in the Max window. Every time an error condition is encountered, the error outlet sends a bang, and an error message is displayed. If the same error occurs more than once, the error message is not displayed again.

iCube error() Can't create negative number of outlets when creating a new iCube object, you've specified a negative number as the number of virtual outlets.

iCube error (outlet#) Sensor ID out of range (1..32)
You are attempting to connect a virtual outlet to a non-existent Digitizer input.

iCube error(outlet#) Output port ID out of range
You are attempting to access a non-existent virtual outlet (ex. you are banging [6 connect 1] to an iCube object with only 5 outlets).

iCube error (outlet#) Bad resolution setting (0=lo, 1=hi)
You've specified an incorrect resolution setting. Valid setting is 0 for 7 bit, connections and 1 for 12 bit connections.

iCube error() Bad sampling interval (4..16000ms)
You've specified an invalid sampling interval for the system.

iCube error () Invalid number of steps
You've set the number of steps to more than 127(7bit) or more than 4096(12bit)

iCube error (outlet#) Float given
You've specified a number with a decimal point (ex 1.22). The iCube object accepts integers only.

iCube error (Outlet#) Number supplied where name expected
You specified a number instead of a command (ex. 1 1 on)

iCube error (outlet#) Bad command name
The command name given is not in the command list.

iCube error (outlet#) Parameter value missing
You haven't specified a value for the given command (ex. 1 connect)

iCube error (outlet#) Name supplied where number expected
You supplied a name before supplying a virtual outlet number (for example, when typing one on instead of 1 on)

iCube error (outlet#) Outlet not connected
You're attempting to access an outlet which has not been connected to a digitizer input. (ex.1 on, before 1 connect 20)

iCube Error (92) MIDI Protocol Error

The MIDI bytes received by the Digitizer do not comply with the MIDI protocol.

I-Cube Error (94) MIDI byte Scrambled

The reception of MIDI bytes by the Digitizer has malfunctioned.

I-Cube Error (95) MIDI Buffer Full

The MIDI-in buffer in the Digitizer has been filled and the received command bytes may have been lost.

Disclaimer:

Waiver of Liability

Infusion Systems Ltd. offers no warranty, limited or otherwise, on any of its products. Infusion Systems Ltd. advises any user to immediately notify us on the discovery of any major failure in the functionality of any piece of equipment or software purchased from us.

UNDER NO CIRCUMSTANCES WILL INFUSION SYSTEMS LTD. BE LIABLE FOR ANY LOST PROFITS, LOST SAVINGS, ANY INCIDENTAL DAMAGES, OR ANY CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PRODUCT, EVEN IF INFUSION SYSTEMS LTD. HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.