

# I-CUBE X

## Editor v2.2x I-CubeX Reference

*I-CubeX: The ultimate MIDI controller!*

Infusion Systems Ltd.  
2033 Avenue de Vendome  
Montreal, QC  
Canada H4A 3M4  
Tel: (514) 484 5850  
Fax: (514) 484 5852  
Email: [info@infusionsystems.com](mailto:info@infusionsystems.com)  
<http://www.infusionsystems.com>



15 August 2003  
© 1997-2003 Infusion Systems

Interac

Customize your own perfect interface  
Performer-activated lighting and sound cues  
Interactive trade show displays  
music synchronization for dance  
greater accessibility via alternate controllers

## The I-CubeX environment

The I-CubeX is a powerful analog (control voltage) to MIDI translator with 32 inputs, of which 8 can double as actuator outputs. The I-CubeX is the missing link between you and your MIDI equipment and is at home in both studio and live performance settings. It will allow you to control (and be controlled by) your MIDI gear in ways you never thought possible (or sensible, for that matter).

This help file contains a wealth of information, hints, tips and tricks to tune your sensor setup to your needs, be it in the sound studio, on stage, in the gallery, the exhibit hall, simply in your living, or even outside. The I-CubeX can be taken anywhere. Rest assured that we'll make an official announcement of the first arrival of the I-CubeX on the moon.

Back down on earth, we really hope you will find the I-CubeX a useful tool, something that enables and inspires you to make great art, deep scientific inquiries or insanely funny installations for your friends (or just your cat). Check out the Cube Culture section on our web site to see what others have done with the I-CubeX so far - we want to make the I-CubeX the best tool for these explorations.

Any technical questions, please contact us at [Tech@InfusionSystems.com](mailto:Tech@InfusionSystems.com) or send a message to [Cube-Culture@InfusionSystems.com](mailto:Cube-Culture@InfusionSystems.com) to ponder collectively with your fellow I-Cubists. To place an order, please contact [Sales@InfusionSystems.com](mailto:Sales@InfusionSystems.com). For any other questions contact [CustomerService@InfusionSystems.com](mailto:CustomerService@InfusionSystems.com) or call toll-free 1 877 4 ICUBEX (+1 604 983 3640). Find us on the web at <http://InfusionSystems.com>.

I-Cubically,

Infusion Systems Ltd.

## Important safety instructions

When using electric products, basic cautions should always be followed, including the following:

1. Read all safety and operating instructions before using this product.
2. All safety and operating instructions should be retained for future reference.
3. This product should not be used near water, ie. a bathtub, sink, swimming pool, wet basement, etc..
4. This product should not be placed near a source of heat such as a stove, radiator, a heat producing amplifier, etc..
5. Connect only to a power supply of the type marked on the top facing label of the Digitizer.
6. The cord to the power supply should always be handled carefully. Never walk or place equipment on the power supply cords. Periodically check the cords for cuts or signs of stress, especially near the plug and the at the point where the cable exits the unit.
7. The power supply should be unplugged whenever the Digitizer remains unused for any period longer than two hours.
8. Water can enter the Digitizer very easily, and not only potentially harm it, but also cause a safety hazard. Do not clean the Digitizer with a damp cloth.
9. Care should be taken so that objects do not fall and liquids are not spilled onto the unit.
10. Infusion Systems Ltd. should be contacted immediately if any problems occur with the regular functioning of any part of the system.
11. The user should not attempt to service this equipment. All service work should be performed by Infusion Systems Ltd..
12. Do not maintain contact between the sensors and the skin of any person. Wearing sensors on the body without the proper insulation could be hazardous.

The Digitizer and the sensors provided by Infusion Systems Ltd. are delicate electrical instruments, and are easily damaged. To ensure that you do not unnecessarily harm them in any way we recommend the following:

1. Do not drop the Digitizer or sensors.
2. Be careful not to step on the Digitizer or sensors.
3. Do not connect any devices to an input or outputs of the Digitizer that draw more than 30 mA per input or output.
4. Do not short the power and ground pins of any of the input plugs on the Digitizer.
5. Do not wet the Digitizer or sensors.
6. Do not use the power supply with voltages other than specified on the power supply unit.

### 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.

## Connecting the cables

Having installed the I-CubeX editing software, you are now ready to setup the I-CubeX hardware – the Digitizer.



*Make sure you are using a Digitizer with firmware version 4.1. Any other firmware is not compatible with editor v2.2 and you will receive an error message at startup.*

This section will help you to make the correct connections as well as diagnose any setup problems you may encounter. Use the section about the [Digitizer hardware](#) for reference.

### Making connections

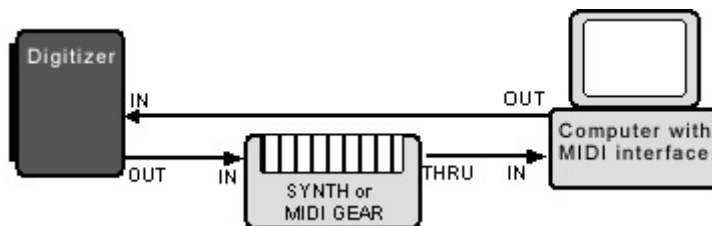


*Before connecting and plugging MIDI cables into your Digitizer, make sure the Digitizer's power is disconnected ! Hot (un)plugging (plugging or unplugging cables while the power is on and/or the editor is running) is not guaranteed to work in the I-CubeX environment. If the power or any of the MIDI cables is disconnected while using the editor, the editor may crash or hang.*

In order to program the Digitizer using the I-CubeX editor, you will need to connect the Digitizer's MIDI output to your MIDI interface's MIDI input, and the Digitizer's MIDI input to your MIDI interface's MIDI output.



*Fig. 1-1: Stand alone mode basic MIDI signal path between the Digitizer and computer.*

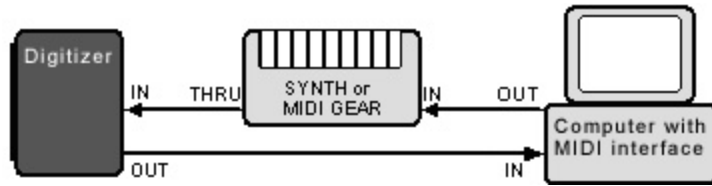


*Fig. 1-2: Alternate stand alone mode MIDI signal path including a synthesizer or other MIDI gear.*

You may chain other MIDI devices (thru) between the computer and the Digitizer (although some older devices may not be able to keep up with the sheer amount of MIDI messages).

For host mode only, we suggest that you plug your MIDI devices between the PC's MIDI output

and the Digitizer's MIDI input.



*Fig. 1-3: Host mode MIDI signal path including a synthesizer or other MIDI gear.*

Next, make sure that your MIDI interface's MIDI thru has been disabled (this passes MIDI messages from the MIDI input port directly out to the MIDI output port). This parameter can be adjusted using the driver or setup programs that came with your MIDI interface. On multi-port interfaces use the supplied MIDI patch bay software to unplug any links between each port's MIDI In and it's MIDI output. This prevents MIDI feedback loops.

Connect the supplied power adapter to the Digitizer's power supply socket. Insert the adapter plug in a suitable wall socket or power bar. As the Digitizer has no power switch, it will turn on immediately. To turn the Digitizer off, simply unplug the power adapter from the Digitizer's power supply socket.

Having connected the hardware, you are now ready to configure the I-CubeX software to work with your Digitizer.

## Setting up MIDI communications

1) Go to Configure | MIDI Connections. Select the MIDI input and output drivers for the MIDI interface connected with the I-CubeX Digitizer. This MIDI link is vital - it is the only way for the I-CubeX editor to communicate with the Digitizer.

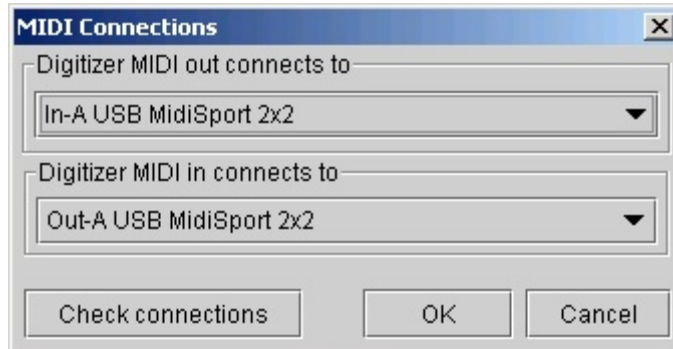


Fig. 1-4: MIDI connections dialog window.

2) Test the MIDI connection by clicking "Check connections". If the test wasn't successful it is possible that the Digitizer is too busy sending out MIDI messages (if your Digitizer was never used before, ie. you just bought it, this shouldn't be the case) or receiving MIDI messages. If it is too busy sending out MIDI messages (that were not forwarded from the MIDI input using the Digitizer MIDI thru option), increasing the sample interval (lowering the sample rate) may resolve this. If the Digitizer is too busy receiving MIDI messages check which other MIDI device may be sending these messages.

3) Go to Configure and make sure MIDI Thru is off (unchecked).

4) Go to Configure | Reset. Select sysex ID to make sure the system exclusive ID of the Digitizer is set to zero. Normally this step is not necessary because at startup the Digitizer's sysex ID is set to zero.



Fig. 1-5: Using the Configure menu to reset the Sysex ID to 0.

5) Go to Configure | Mode. Select stand alone mode. You should see the Digitizer's yellow MIDI input LED flash briefly. If this doesn't happen, you have not chosen the right MIDI drivers (repeat step 1). Normally this step and the following is not necessary because at startup the Digitizer is set to stand alone mode.

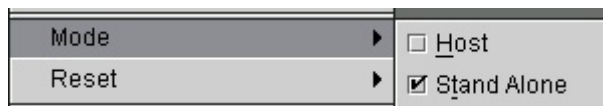
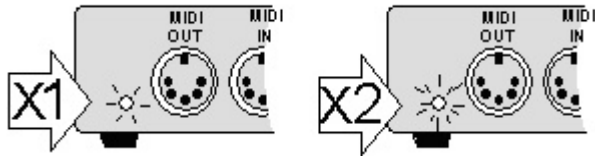


Fig. 1-6: Using the Configure menu for setting the mode of operation.

6) Power off and then power up the Digitizer. Note the number of the green MIDI output LED blinks at power up. Ensure that your Digitizer is switched to stand alone mode.



*Fig. 1-7: Number of flashes of the Digitizer MIDI out LED on power-up indicates the mode. Host mode is indicated by one green flash during power up. Stand alone mode is indicated by two green flashes during power up.*

## Testing the I-CubeX environment

Having set up your hardware (Digitizer) and software (I-CubeX editor) and its connections, you are now ready to make some creative connections – those that connect your I-CubeX sensors to your MIDI gear.

This section will demonstrate how to connect sensors to the Digitizer, as well as how to use them with your MIDI equipment.

### Getting sensor input

- 1) Find the Turn sensor included with your I-CubeX package.
- 2) Connect the Turn sensor to the Digitizer's first sensor input (labeled '1') by inserting the 3-pin sensor plug (see [Fig. 3-3](#)) vertically (with the red wire pin on top) to sensor input 1.
- 3) Start the I-CubeX editor.
- 4) Clear the configuration by selecting File | New.
- 5) Turn the first sensor on. Use Configure | Add sensor, then click the “On” button.
- 6) Map the first sensor to a volume controller on MIDI channel 1. Set the MIDI Channel to 1, select “Control-change” from the *MIDI map* pull down menu, and type 7 in the Control field.
- 7) Turn the knob and observe the green monitor bar at the bottom of the sensor window for activity.
- 8) Congratulations! Your turn sensor is controlling MIDI volume on channel 1. Let's test this with your MIDI gear.
- 9) Connect the Digitizer's MIDI Out to a MIDI keyboard's MIDI In. Set your MIDI keyboard to receive on MIDI channel 1. Also turn your keyboard's local control on. Select a patch on your keyboard (on MIDI channel 1). Play and hold a few notes.
- 10) Now tweak the Turn sensor. You are now controlling MIDI volume.
- 11) You may save your configuration by selecting File | Save.
- 12) Leave the I-CubeX editor on.
- 13) Let's continue by utilizing your See actuator!

### Sending actuator output

- 14) Find the See actuator (LED) included with your I-CubeX package.
- 15) Connect the See actuator to the Digitizer's first output (labeled 'Outputs 1' / Sensor input 25) by inserting the 3-pin connector (see [Fig. 3-3](#)) vertically (with the red wire pin on top) to output 1.
- 16) Start the I-CubeX editor (unless it's already running).



17) Set the actuator outputs MIDI Map channel to 1 (go the outputs window and set the remote MIDI control channel to 1).

18) Set the outputs MIDI Map base note to 60 (middle C) - go the actuators window and set the MIDI Map base note to 60.

19) Set the first actuator output to trigger mode (click on the first mode button until it displays the up-down arrow). This actuator output will turn the See actuator on when note 60 (middle C) is pressed down, and will turn the actuator off when note 60 (middle C) is released.

20) Save your configuration by selecting File | Save.

21) Connect your MIDI keyboard's MIDI output to your Digitizer's MIDI input.

22) Set up your keyboard to play on MIDI channel 1. Play middle C (or note 60). The See LED will turn on. Release the note and the LED will turn off.

23) Start turning the Turn sensor to control volume, play middle C to control the See LED.

Congratulations, you have created your very first I-CubeX sensor environment.

## A MIDI mixer example

For this example, we will be using 16 Turn sensors. You can still do this tutorial if you have fewer Turn sensors by allocating fewer sensor inputs (in step 6).

1. Label each one with a number (1 through 16). You may wish to mount your sensors on a panel by drilling a 1/4" diameter hole for each Turn sensor. Remove the sensor knob by loosening the small screw on the side of the knob. Remove the washer. Pass the sensor through the hole. Put the washer back on the sensor. Tighten the nut so the sensor stays firmly in place. Put the knob back onto the sensor, and tighten the screw on the side of the knob).
2. Connect each Turn sensor to the I-CubeX Digitizer's inputs (starting from input 1) by inserting the 3-pin of each sensor connector vertically (with the red wire pin on top) starting with input 1.
3. Connect the Digitizer to your MIDI interface (see "[Cabling](#)" section).
4. Start the I-CubeX editor (the I-CubeX editor can be started directly from the Windows start menu under programs).
5. Clear the configuration by selecting File | New.
6. Create 16 sensor windows (or fewer if you're using fewer Turn sensors) by selecting Configure | Add sensor. Note that the new sensor windows use sensor inputs 1 through 16.
7. Find the first sensor window (which has been assigned sensor input 1). Go to the *MIDI Map* section and select "*Control-change*" from the list box. Set the channel field to 1. Set the *Control* field to 7 (ie. volume).
8. Repeat step 7 for the second sensor window (assigned to sensor input 2). This time set the *MIDI Map Channel* field to 2, but set the *Control* field again to 7 (ie. volume).
9. Repeat the above step for each consecutive sensor window (up to 16, or less if you're using fewer turn sensors) each time incrementing the *MIDI Map Channel* field only.
10. Set all 16 (or fewer sensor windows) on by clicking the "on" button for each sensor window.
11. Select each sensor window and start turning the corresponding knob (The input monitor will only monitor the MIDI activity of the selected sensor). If you see no activity, make sure that you're turning the correct knob.
12. Save your masterpiece by selecting File | Save (or File | Save As..). Name it eg. "My MIDI Mixer".
13. Set the Digitizer's MIDI output to "thru" mode by selecting the Configure menu and putting a check mark by MIDI thru. This will allow the Digitizer to pass MIDI from your keyboard controller.
14. Congratulations, you created your very own MIDI mixer!



## **Configure it !**

The basic element of the I-CubeX editor is the configuration. Each configuration represents up to 32 sensors (using up to 32 sensor windows), 8 actuators (using the actuators window) and a few system variables (using the system window and the configure menu). Each configuration can be sent to and retrieved from the Digitizer, as well as stored and loaded from your hard drive. In effect this turns the I-CubeX editor into a librarian. Now that you know how to configure your I-CubeX environment, go wild and make some interactive connections with your MIDI gear !

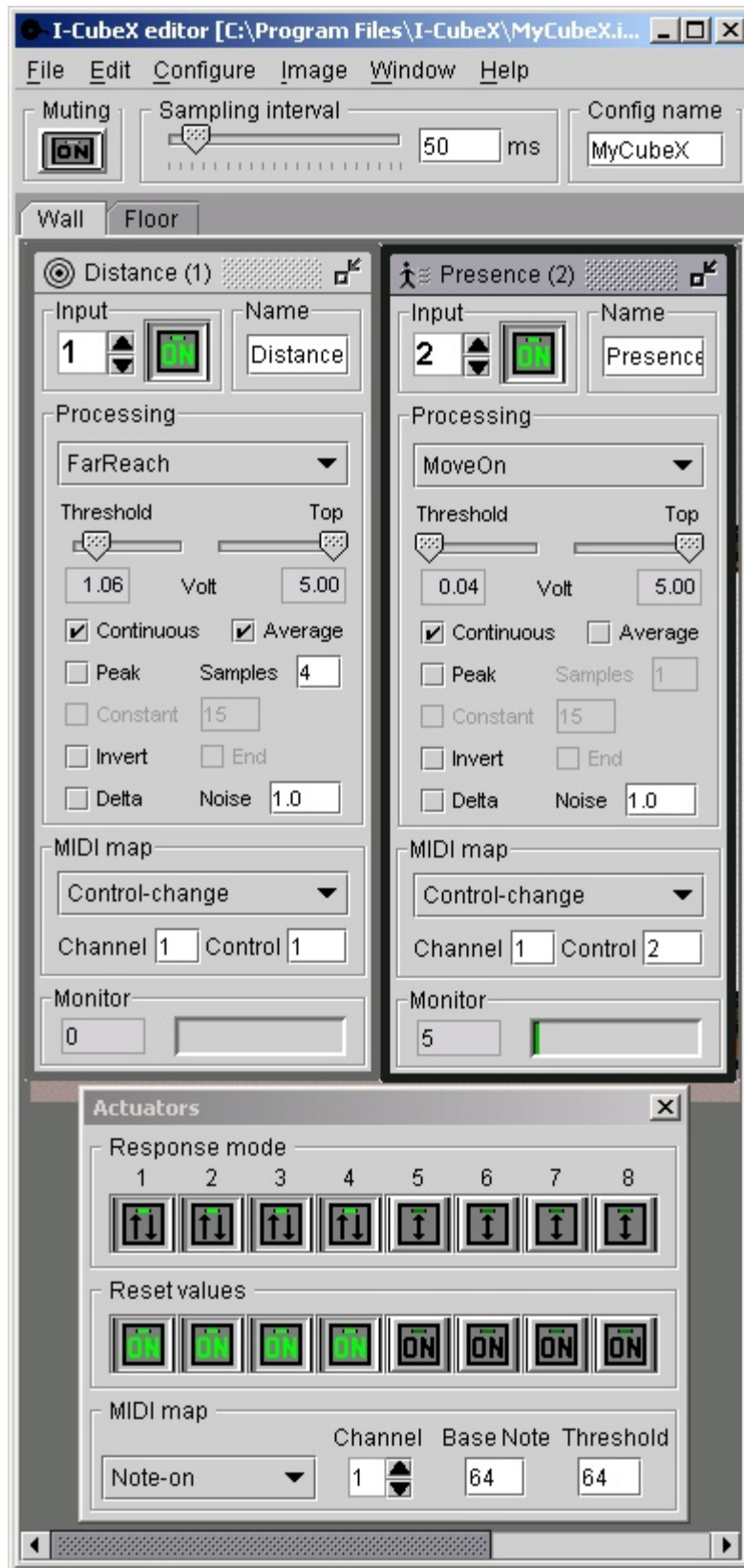


Fig 1-8: The I-CubeX editor with two sensor windows, actuators and system window. See [fig](#).

*2-7 to see the background image. To explore the shown configuration, load MyCubeX into the editor. It can be found in the I-CubeX program directory (shown on the top bar of the application window).*



*Configurations saved in I-CubeX editor version 2.1 or 2.2 can only be loaded in version 2.1 or 2.2 and cannot be loaded in editor version 2.0 or 1.5 nor vice versa. To convert a configuration file from an older editor to a newer one, load the configuration with the appropriate editor into the Digitizer and then obtain the configuration from the Digitizer using the newer editor.*

## The file menu

This menu manages the storage and retrieval of your sensor configurations.

### **New (Ctrl+N)**

Creates a new configuration (and clears the Digitizer's memory if auto update is on). A new configuration contains the actuators window, the system window, but no sensor windows. If you were editing a configuration, you will be asked whether you wish to clear that configuration or not. This gives you an opportunity to save your work.



*When using File | New when auto update is turned on (see the Configure window) the action will clear the Digitizer's memory.*

### **Open (Ctrl+O)**

Opens a saved configuration (and sends it to the Digitizer if auto update is on). Again, if you were editing a configuration, you will be asked whether you wish to clear that configuration or not. This gives you an opportunity to save your work.

### **Save (Ctrl+S)**

Saves your configuration. If the configuration has never been saved, you will be prompted for a file name. See the section on file formats for information which features of the configuration are saved in the file.

### **Record (Ctrl+E)**

Record all channel voice MIDI data received by the MIDI input device specified in the MIDI connections window to a file. This function enables the recording of MIDI message values as they are sent by the Digitizer. All active sensors are recorded simultaneously. Each value is, together with a time stamp in milliseconds, written to a file. Selecting the Record function will bring up a window to select/create a file to save the MIDI message values into. The values are saved as a text file (ie. ASCII data) in the following format (using only integers): "time,sensor input number,MIDI channel,MIDI control number or note number or -1 (in the case of program-change, after-touch or pitch-bend),MIDI value" eg.:

```
"  
0,1,2,1,55  
277,2,3,64,60  
300,3,1,-1,1069  
"
```

This sequence of data represents a MIDI control or note number 1, controller or velocity value 55 computed from sensor input number 1, sent on MIDI channel 2 at the start of the recording, then a MIDI control or note number 64, controller or velocity value 60 computed from sensor input number 2 sent on MIDI channel 3 at 277 ms after starting recording, then a MIDI pitch bend value 1069 computed from sensor input number 3 sent on MIDI channel 1 at 300 ms after starting recording.



*Note that the values are written asynchronously (ie. the interval between time stamps varies)*

*because the Digitizer only sends out a MIDI message when the sensor value has changed more than the noisegate value.*

***Exit (Ctrl+Q)***

Exits the I-CubeX editor. Make sure to save your edited configuration.



## The edit menu

### **Copy (Ctrl+C)**

### **Cut (Ctrl+U)**

### **Paste (Ctrl+V)**

Standard cut, copy and paste functions. You can cut, copy or paste any field which you select (sensor name, etc...). You cannot cut, copy or paste an entire sensor window.

### **Tabs (Ctrl+T)**

Tabs are windows that allow you to subdivide your sensors into different groups with separate tasks so that it's easier to keep track of all the sensors in your art installation. Each tab can hold a different background image in which each sensor window can be minimized and placed to provide a visualization of your setup. New tabs can be added, deleted and renamed.

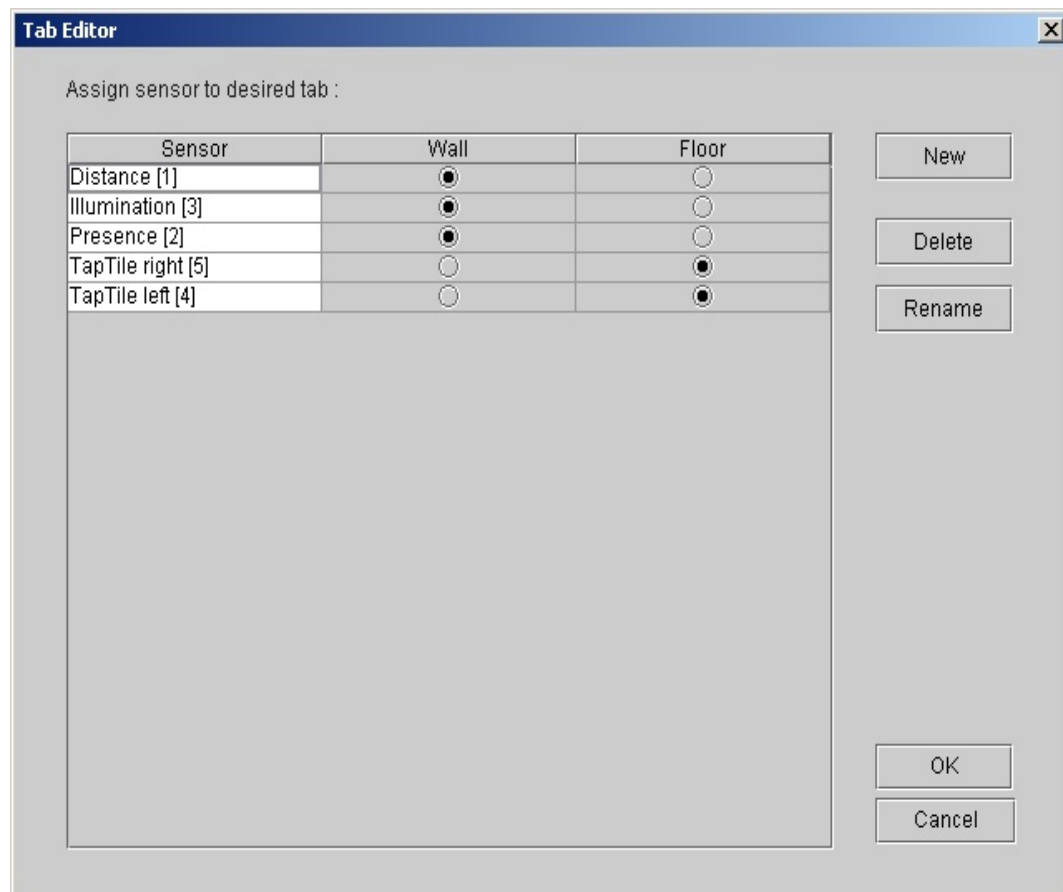


Fig. 2-2: The tab editor.

## The configure menu

<u>A</u> dd sensor	Ctrl-Insert
<u>D</u> elete sensor	Ctrl-Delete
<input type="checkbox"/> Auto update	
<u>R</u> eceive	Ctrl-R
<u>S</u> end	Ctrl-D
Mode	▶
Reset	▶
<input type="checkbox"/> MIDI thru	
<u>M</u> IDI connections	

Fig. 2-3-1: The configure menu.

### **Add/Delete sensor (Ctrl+Ins/Ctrl+Del)**

Inserts a new sensor window or delete an existing sensor window. You can add up to 32 sensor windows.

### **Auto update**

If checked, all configuration parameter changes are sent to the Digitizer as soon as they are changed in the editor. If unchecked, parameter changes are not stored in the Digitizer - use the Configure | Send command to store the changes. [default: checked]



*When auto update is turned on, the MIDI connections are never checked before sending updates of Digitizer parameters (when using "Send" and "Receive" the MIDI connections are checked). If for some reason your MIDI connection is unreliable (test it with the "Check connections" button in Configure | MIDI connections), the Digitizer may miss a command to change a parameter sent by the editor and no warnings will be displayed.*



*If the editor was started with auto update turned off and turned on later, you need to either Send or Receive a configuration to/from the Digitizer to make sure the editor displays the same configuration as stored in the Digitizer.*

### **Send (Ctrl+D)**

Sends the current configuration to the Digitizer (regardless whether auto update is checked). A transfer progress bar indicates the state of the MIDI transfer.

### **Receive (Ctrl+R)**

Requests and receives the configuration from the Digitizer's memory (regardless whether auto update is checked). A transfer progress bar indicates the state of the MIDI transfer.



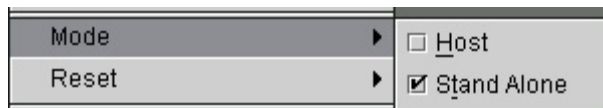
*Configure | Receive will only obtain the configuration of sensor inputs that are turned on, ie. are sampling sensor signals and subsequently sending these out as MIDI messages.*



*The MIDI connections are checked when executing Send or Receive - if no Digitizer is found there will be a warning displayed.*

### **Mode**

Switch the Digitizer between host or stand alone mode. Note that upon startup of the editor, the Digitizer will be set to stand alone mode. Another way to find out the Digitizer's mode of operation is to power off and then power up the Digitizer and count the number blinks of the green MIDI output LED at power up. Host mode is indicated by one green flash during power up (see Fig. 1-6). Stand alone mode is indicated by two green flashes during power up (see Fig. 1-6). Do not set the Digitizer to host mode unless you are planning to communicate with the Digitizer in MIDI system exclusive mode using some other program. [default: stand alone]



*Fig. 2-3-2: Using the Configure menu for setting the mode of operation.*

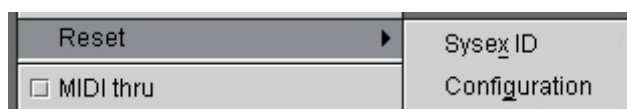


*The Digitizer must be in stand alone mode to be able to use all of the editor's functions.*

### **Reset**

Reset sysex ID: In order to configure the Digitizer, the editor communicates with the Digitizer using a proprietary MIDI system exclusive (sysex) protocol. For host mode operation, the Digitizer can be set to have any device ID between 0 and 127. The device ID 0 is used by the editor for all stand alone mode configuration programming. The "Reset system exclusive device ID" option gives the user the possibility to reset the device ID to 0 in case it was set to another value during a host mode session. Note that upon startup of the editor the Digitizer's sysex ID is reset to 0.

Reset configuration: reset the Digitizer configuration to the default factory settings. This is the same operation as executing *File | New*.



*Fig. 2-3-3: Using the Configure menu to reset the Sysex ID to 0.*

***MIDI thru***

Determines whether the Digitizer's MIDI output port acts as a MIDI thru or not. With MIDI thru active, the Digitizer will pass on every message that is not intended for the Digitizer to the next MIDI device in the MIDI chain. [default: off]

***MIDI connections***

Enables you to select the appropriate MIDI interface (MIDI input and output) to connect to the I-CubeX Digitizer.

## The configure menu - MIDI Connections item

### **MIDI connections**

Enables you to select the appropriate MIDI interface (MIDI input and output) to connect to the I-CubeX Digitizer.

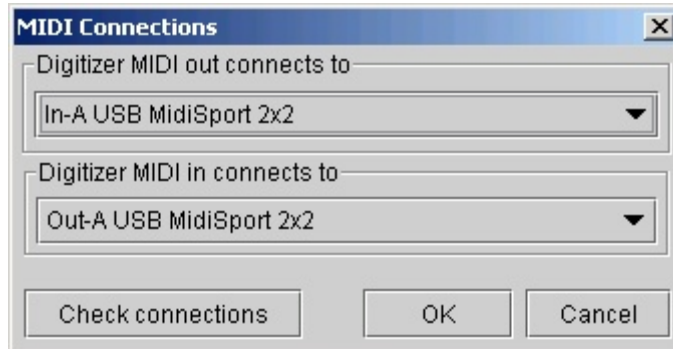


Fig. 2-3-4: MIDI connections dialog window.

The *Check connections* button enables you to test the reliability of the MIDI connections. Under normal circumstances all tests should be successful.

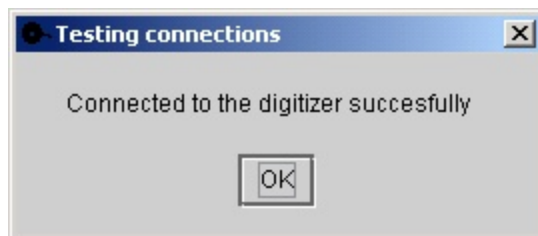


Fig 2-3-5: Check connections dialog.



*If not all of the tests were successful after pressing Check connections it is possible that the Digitizer is too busy sending out or receiving MIDI messages. If it is too busy sending out MIDI messages (that were not forwarded from the MIDI input using the Digitizer MIDI thru option), increasing the sample interval (lowering the sample rate) temporarily may resolve this. If the Digitizer is too busy receiving MIDI messages check which other MIDI device may be sending these messages.*



*A host mode Reset message is sent to the Digitizer each time the MIDI connections are checked. The Reset message may change the status of the actuator outputs (ie. power supplies of inputs 25-32).*



## The sensor window (Ctrl+Ins)

A sensor window maps one sensor input to a standard MIDI message (a note-off, note-on, key-pressure, control-change, program-change, after-touch, or pitch-bend message). It is created in Configure | Add sensor. Up to 32 sensor windows can be created, one for each sensor input, per configuration. A sensor window is composed of the following user interface components:

- Input section: number selector, *on/off* button
- Name section: *Name* field
- Input processing section: Preset pull down menu, *Threshold* and *Top* sliders and number displays, *Continuous*, *Peak/Dip*, *Average*, *Constant*, *End* and *Delta* check boxes, *Noise*, *Constant* and *Samples* number fields
- MIDI map section: Message pull down menu, *Channel* selector, 1st data field (*Control*, *Note*, or *n/a*)
- Monitor section: Monitor slider and number display

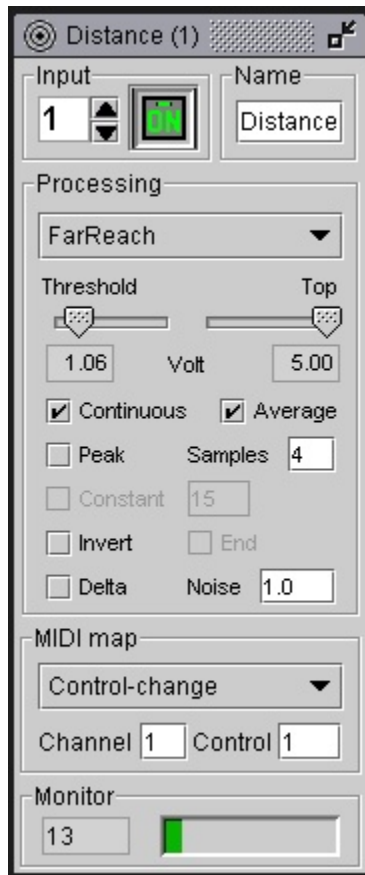


Fig. 2-4-1: A sensor window.

## The sensor window - input and name sections



Fig. 2-4-2: The input and name sections of a sensor window.

**Input number selector:** Selects the sensor input number (ie. the input of the Digitizer used for connecting the sensor to the Digitizer) to be used with the sensor window. You can scroll through all unused sensor input channels (channels which are not used by any other sensor input objects). Each sensor input object must use a different sensor input. [default: first sensor input number that is not used by any other open sensor window]

**On/off button:** This button turns the sensor input on and off (ie. if the sensor input is turned on the Digitizer will send out MIDI messages in response to sensor value changes). [default: off]



*Leaving an input on without a sensor plugged in will cause signal noise to be translated to MIDI messages and may cause large amounts of MIDI data to be output, very often overloading your MIDI interface or other MIDI gear! Make sure to turn unused inputs off.*

**Name field:** Allows you to name the sensor configured with this sensor window. This label can indicate the type of sensor used, the location of the sensor, the function of the sensor, or anything else. The name is updated in the window title bar. By keeping this label short, the entire label can be visible when the object is minimized (great if you are using all 32 sensor input objects). [default: "Sensor"]



*The sensor name is not stored in the Digitizer and hence is entered as "Sensor" when retrieving a configuration from the Digitizer.*



## The sensor window - processing section

The processing section enables you to maximize the sensitivity and response of the sensor.

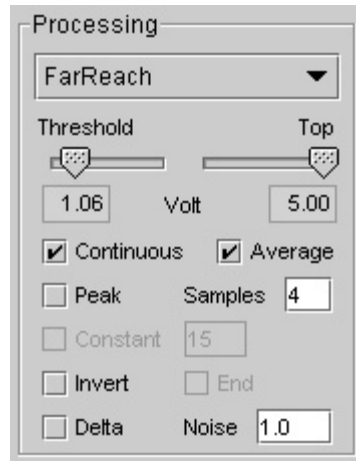


Fig. 2-4-3: The processing section of a sensor window with the FarReach preset selected.

**Preset pull down menu:** To facilitate configuring the processing section of a sensor window, a setting can be selected from a list of presets for all of Infusion's sensors. You can add your own processing settings by editing the presets.ini file in the directory where the I-CubeX program is stored. See the [File Formats](#) section for information on the presets.ini file format. [default: "no processing"]



*The processing preset names are not stored in the Digitizer and hence are left blank when retrieving a configuration from the Digitizer. The processing settings are correctly retrieved, just the name is not stored in the Digitizer.*

**Threshold slider:** Sets the lower limit of the effective sensor input voltage range (any voltages applied to the sensor input below this point return 0). When *Peak* is checked this parameter is also used as the trigger for starting a peak search. Raise *Threshold* to something other than 0 - let's say 1.02 Volt. When the sensor value rises above 1.02 Volt and the number of samples specified in the *Samples* field have been obtained, it triggers a MIDI event. When it drops again below 1.02 Volt, and if *End* is checked, it triggers another MIDI message. This value cannot be set above the *Top* setting. [default: 0.00]

**Top slider:** Sets the upper limit of the effective sensor input voltage range (any voltages applied to the sensor input above this point return 127). When *Dip* is checked this parameter is also used as the trigger for starting a dip search. Lower *Top* to something other than 5 - let's say 3.98 Volt. When the sensor value drops below 3.98 Volt and the number of samples specified in the *Samples* field have been obtained, it triggers a MIDI event. When it rises again above 3.98 Volt, and if *End* is checked, it triggers another MIDI message. This value cannot be set below the *Threshold* setting. [default: 5.00]

**Continuous check box:** When checked, any new sensor values will be output. Use this feature when you are interested in all the sensor values. Checking *Continuous* will allow checking

*Average*. When a new sensor window is added, turning the sensor on sets *Continuous* to checked by default. [default: checked when the input is turned on]

**Peak/Dip check box:** When checked, peak/dip detection is active. By default the processing algorithm looks for a peak - if *Invert* is checked the algorithm looks for a dip in the sensor signal. When the sensor value raises above *Threshold* (in case of peak detection) or drops below *Top* (in case of dip detection) the peak/dip search starts. The number of samples taken before the peak or dip value is determined is set using the *Samples* field. The output value is scaled between *Threshold* and *Top*. Use this feature when you are interested in detecting sudden significant changes in the sensor values, ie. when they are occasionally jumping to a high or a low value. Checking *Peak/Dip* will also allow checking *Constant* and *End* detection. [default: unchecked]



Checking both *Continuous* and *Peak/Dip* will set the processing algorithm to first look for a peak/dip in the sensor signal according to the *Samples* value, and then output sensor values as key-pressure values until the sensor value drops below *Threshold* (in case of peak detection) or rises above *Top* (in case of dip detection).

**Average check box:** When checked, sensor values will be averaged. The number of samples to be taken for calculation of the average is set using the *Samples* field. Use this feature if the variation of the sensor values due to noise is almost as large or larger than the variation of the sensor values due to stimulus variation. The average operation is only available when *Continuous* is checked. [default: unchecked]

**Samples number field:** Use for entering the number of samples to be taken before the peak/dip value (*Peak/Dip* checked) and/or average value (*Average* checked) should be computed. If both *Continuous* and *Peak/Dip* are checked the *Samples* value will first be used for peak/dip detection and then for averaging. [default: 0]

**Delta check box:** When checked, only the difference between the current and the previous sensor value will be output. It is useful when you are interested in the changes of the sensor values only, not the sensor value itself. The delta operation is a simple form of obtaining the differentiated sensor value, better known as its velocity, speed or rate of change. The delta operation is applied after any averaging and/or peak/dip detection but before the noise gate operation. [default: unchecked]

**Noise number field:** Use for entering a noise gate value such that the MIDI mapped value of a sensor is only output if it changes at least by the number entered in the *Noise* field. The noise gate value depends on the MIDI map selected. When using 7-bit resolution MIDI messages such as note-on and control-change messages a noise gate value of 0.5 means that the MIDI message needs to change at least by one half up or down, ie. the sensor value needs to change at least 0.02 Volt (ie. 20 milliVolt) before a MIDI message is output. When using pitch-bend messages a noise gate value of 4 means that the pitch-bend value needs to change by at least 4 up or down, ie. the sensor value needs to change at least 0.0012 Volt (ie. 1.2 milliVolt) before a MIDI message is output. Use this feature if the variation of the sensor values due to noise is much less than the variation of the sensor values due to stimulus variation. The noise gate operation is applied after any averaging and/or peak/dip detection and/or delta operations. [default: 0.5]

**Constant check box:** When checked, no peak or dip value is computed but the *Constant* number field is used instead for immediately (ie. the number specified in the *Samples* field is irrelevant)

outputting the MIDI message specified in *MIDI map* when *Threshold* (in case of peak detection) or *Top* (in case of dip detection) are passed. Useful when you want to make sure the MIDI message is always the same when your sensor is activated. The constant operation is only available when *Peak/Dip* is checked. [default: unchecked]

**Continuous number field:** Use for entering the *Constant* value to use in the MIDI message when *Threshold* or *Top* are passed. Only available when *Constant* is checked. [default: 15 when *Constant* is checked]

**End (detection) check box:** When checked, a MIDI value zero will be output whenever the sensor value drops below the *Threshold* value (in case of peak detection) or above the *Top* value (in case of dip detection) selected. The end detection operation is only available when *Peak/Dip* is checked. [default: unchecked]

**Invert check box:** When checked the MIDI output value decreases with increasing sensor values. When *Peak* is checked the processing algorithm will look for a dip in the signal instead of a peak (and vice versa). [default: unchecked]

## The sensor window - MIDI map section

This section enables you to map the sensor value to three standard MIDI messages.



*Fig. 2-4-4: The MIDI map section of a sensor window with the sensor value mapped to a control-change number 1 message on MIDI channel number 1.*

**Message pull down menu:** This menu enables you to map the sensor value to all 7 standard MIDI messages - note-off, note-on, key-pressure, control-change, program-change, after-touch and pitch-bend. [default: control-change]



*The pitch-bend MIDI map maps a sensor signal to MIDI values in steps of 4 because the pitch-bend range is 14 bit (ie. 0-16383) while the sensor values are sampled with 12 bit resolution (ie. 0-4095).*

**Channel selector:** Sets the MIDI channel for the MIDI Message selected above. [default: 1]

**1st data field:** Sets the controller number (if Message is set to Control-change) or note number (if Message is set to Note-on, Note-off or Key-pressure) mapping. Disabled otherwise. [default: same number as sensor input number]

## The sensor window - monitor section

This feature displays the MIDI value as output by the Digitizer of the sensor configured with the current sensor window.



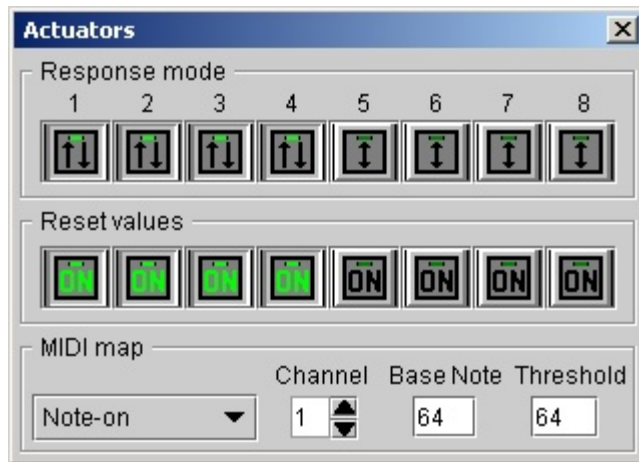
*Fig. 2-4-5: The monitor section of a sensor window.*

**Displays:** A moving bar (left to right) indicates the sensor's MIDI value. When mapped to a continuous controller message the bar is blank if the sensor value is 0 and full if the value is 127. In the case of a pitch-bend mapping, full bend up (MIDI value 16383) is indicated by a full bar, and full bend down (MIDI value 0) is indicated by the blank bar. In the case of a note event mapping, an empty bar represents a note off, while a full or semi full bar indicates a note on (the bar's level indicates the note velocity of the event).



*If the MIDI map is identical for two sensor inputs the monitor functions in the respective sensor windows in the editor will both show the sensor signals of the two sensor inputs, which may appear as a noisy signal.*

## The actuators window



*Fig. 2-5-1: Actuators window with outputs 1-4 set to toggle mode and on after reset with the remainder in trigger mode and off after a reset*

The actuators window configures all 8 actuator outputs (ie. power supplies of sensor inputs 25 - 32). The 8 actuator outputs are controlled by sending MIDI messages to the Digitizer's MIDI input port. The actuators window is composed of:

- Response mode section: 8 *Trigger/Toggle* buttons
- MIDI Map section: Message pull down menu, *Channel* selector, *Base Note/Control* number field, *Threshold* number field

## The actuators window - response mode section



Fig. 2-5-2: The response mode section of the actuators window with outputs 1-4 set to toggle mode.

**Trigger/Toggle buttons:** This strip of 8 buttons determines the response mode for each actuator output. [default: toggle]



**Trigger:** The two-sided (up and down) arrow setting indicates that the actuator output responds as a trigger to note-on events: a note-on event turns the output on, while a note-off event turns the output off.



**Toggle:** The other two-arrow (one up, and one down) setting indicates that the actuator output responds in toggle mode to note-on events: consecutive note-on events toggle the output on and off (the first note-on turns the output on, while the next note-on turns the output off).

## The actuators window - reset values section



*Fig. 2-5-3: The reset values section of the actuators window with all outputs set to turn on after a Digitizer reset.*

The reset values section of the actuators window enables you to control which state each actuator output will be in after resetting the Digitizer (e.g. after a power up). [default: ON]



## The actuators window - MIDI map section



Fig. 2-5-4: The MIDI map section of the actuators window.

This section enables you to configure the MIDI message that, if sent to the Digitizer, will activate the actuator connected to a Digitizer output.

**MIDI map channel:** This parameter sets the MIDI channel on which note-on/off messages will be received. [default: 1]

**MIDI map base note/control:** This parameter sets the note or control number that is mapped to activate the first actuator output. For example, If the base note is set to 60 and activates actuator output 1, a note on message with note number 67 will activate the 8th actuator output. The valid range of this parameter is from 0 to 120. [default: 64]

**MIDI map threshold:** Sets the value above which the actuator is activated. [default: 1]

## The system window

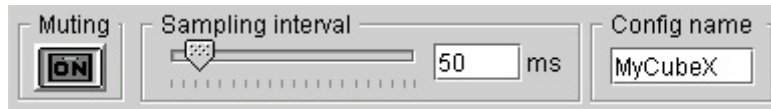


Fig. 2-6: The system window.

The system window controls the following parameters:

**Muting:** This button enables you to stop any sending out of MIDI messages by the Digitizer. [default: off]

**Config name:** The configuration name can be used to store a name with a configuration stored in the Digitizer, so that it is easier to recognize which configuration was last stored in the Digitizer. Maximum length is 8 characters. [default: Digitize]

**Sampling interval:** This parameter determines the Digitizer's sampling speed or rate and is the same for all inputs. The sampling interval ranges from 4 milliseconds (ie. a sampling rate of 250 times per second or 250 Hz) to 16.380 seconds, ie. 16380 milliseconds (ie. a sampling rate of 3.7 times per minute or 0.061 Hz). The slider can range up to 400 milliseconds. For larger sampling intervals, use the number box to the right of the slider. [default: 100 ms]



*Using a small sampling interval (20 ms or less) requires a lot of processing (and may cause a crash on slower computers). Using a small sampling interval with a large number of sensors increases this risk. To avoid this, disconnect the MIDI output cable from the Digitizer and increase the sampling interval.*



*Using a very large sampling interval (3000 ms or more) may make the Digitizer behave as if it was "dead" (ie. there doesn't seem to be any response coming from the Digitizer when changing a sensor value) since sensor data is sent infrequently.*



*All of the above parameters are instantly updated in the Digitizer's memory if auto update is checked in the Configure menu. Whether or not auto update is checked, all sensor windows, and the actuators window settings can be stored in the Digitizer using the Send (Ctrl+D) command found in the Configure menu.*

## **The image menu**

The image menu enables you to add background images to each tab. This feature aims to help organize your sensor setup by providing a visualization of the physical layout of the sensors.

### ***Load/Remove (Ctrl+L/Ctrl+M)***

Load or remove an image file as a background image in one of your tabs.

### ***Show/Hide (Ctrl+W/Ctrl+H)***

Make the selected image(s) visible or hide them.

### ***Record/Restore position(s)***

Record or restore the position of all the minimized sensor windows wrt. the background image. Make sure to record the positions in order to include them in the configuration when saving it as a file.

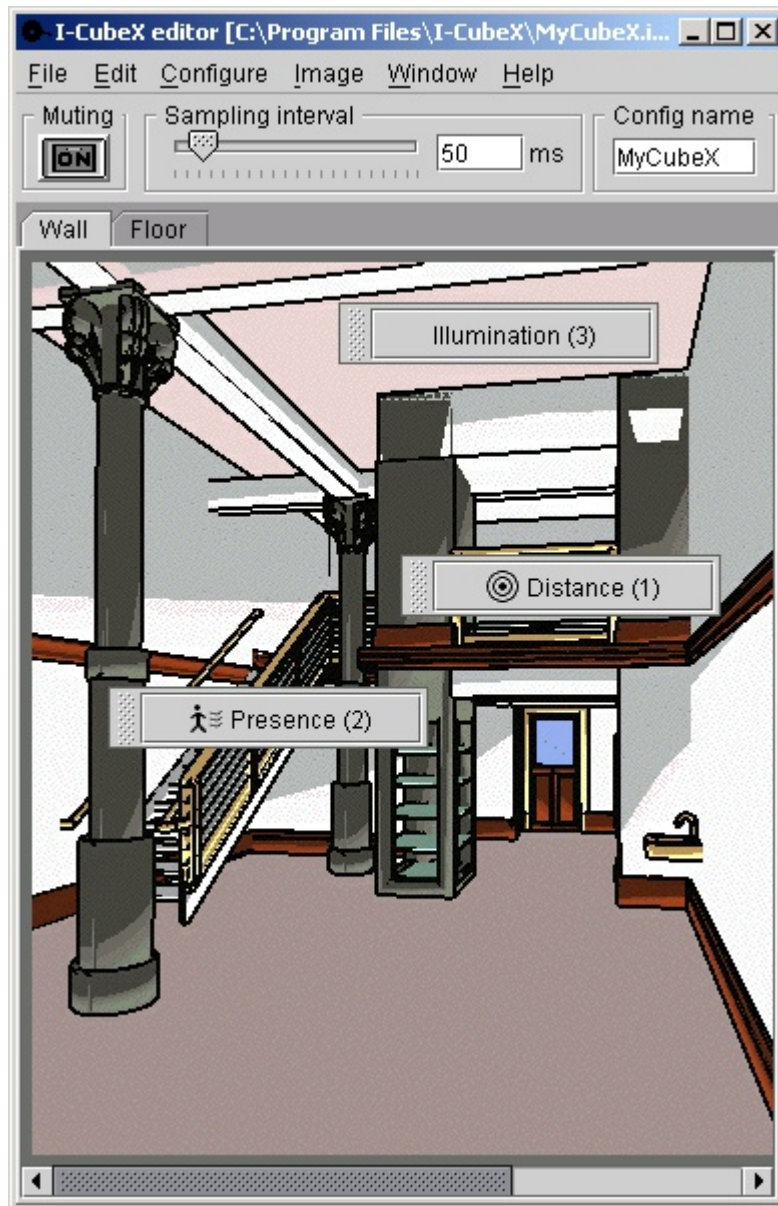


Fig. 2-7: An example of using an image as a background for your I-CubeX sensor setup. Refer to [fig. 1-8](#) to see the sensor windows opened.

## **The window menu**

### ***Tile***

Standard windows function to clean up the position of sensor windows.

### ***Actuators***

Brings the actuators window to the front. If the actuators window has been closed, this command will bring it back again. [default: present]

### ***System***

Brings the system window to the front. If the system window has been closed, this command will bring it back again. [default: present]

## The help menu

### **Help topic**

Brings up this help file in browser window.

### **About**

Brings up a window with the I-CubeX logo, website link and version information.



Fig. 2-9: The About window.

## **Error messages**

### ***Configuration setting error***

The Digitizer received a MIDI system exclusive message containing instructions for storing a configuration that do not make sense. For example setting the base note value higher than 120 does not make sense because the 8th actuator output would be mapped to note value 128 which is an illegal value.

### ***MIDI protocol error***

The Digitizer received a MIDI message that doesn't conform to the MIDI protocol. Check which device(s) is(are) sending MIDI to the Digitizer and that the MIDI data is not corrupted.

### ***Receive buffer is full***

The Digitizer cannot process all the MIDI messages it receives. Please check your MIDI connections and reduce the number of MIDI messages being sent to the Digitizer. This error may also occur when you change one of editor settings continuously for some time, while auto-update is on.

### ***MIDI byte has been scrambled***

The Digitizer cannot recognize the signal it receives through the MIDI cable as a MIDI byte. Check your MIDI connections and make sure the cables are in good order.

## Mode of operation

The I-CubeX Digitizer can function in one of two modes - host and stand alone. Unless you have software that can send and receive the MIDI system exclusive messages required for host mode (such as the iCube and oCube objects used within the Max programming environment), your Digitizer should remain in stand alone mode.

To check whether the I-CubeX is in host or stand alone mode, observe the green MIDI output LED (at the side of the Digitizer, to the left of the MIDI output port) when powering up.

**Host mode** is indicated by **one green flash** during power up (see [Fig. 1-6](#)).

**Stand alone mode** is indicated by **two green flashes** during power up (see [Fig. 1-6](#)).

### Stand alone mode

Stand alone mode allows you to use the I-CubeX Digitizer directly with any MIDI device (keyboard, module, sequencer, light controller, drum machine, micro wave oven). As the Digitizer has no front panel controls, all editing must be performed using the I-CubeX editor. An edited sensor configuration is stored in the Digitizer's memory and remains there even after power down. This allows you to edit the Digitizer, plug it directly into your MIDI equipment, and go!

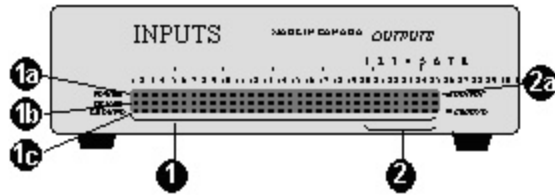
Stand alone mode allows the Digitizer to map any sensor input to a standard MIDI message (note-on/off, key-pressure, control-change, program-change, after-touch or pitch-bend). In addition, the 8 actuator outputs (ie. the power supplies of sensor inputs 25 to 32) can be turned on and off using MIDI note-on/off, key-pressure or control-change messages. While not quite as flexible as host mode, stand alone mode maximizes portability of your sensor setup.

Note that at startup of the editor the Digitizer is set to stand alone mode and the Digitizer's system exclusive ID is set to zero.



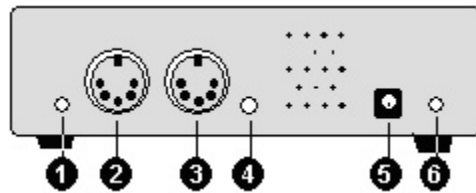
## The I-CubeX hardware - Digitizer v4.0x

**Fig. 3-1: Front View**



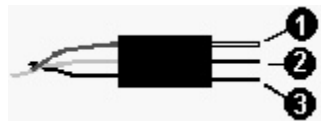
- (1) Sensor inputs (12 bit resolution)
- (1a) Sensor power supply (5VDC / 30mA for each sensor)
- (1b) Sensor signal input
- (1c) Sensor ground
- (2) Actuator outputs (power supplies of last 8 sensor inputs)
- (2a) Actuator signal output (5VDC / 30mA for each actuator)

**Fig. 3-2: Back View**



- (1) MIDI output indicator
- (2) MIDI output (also MIDI Thru) jack
- (3) MIDI input jack
- (4) MIDI input indicator
- (5) Power supply socket (7.5VDC / 1A)
- (6) Power indicator

**Fig. 3-3: Sensor Connector**



- (1) Power pin (+) - red wire
- (2) Signal pin (signal) - white or grey wire
- (3) Ground pin (-) - black wire

## I-CubeX Digitizer technical specifications

**Version** hardware v4.01, firmware v4.1

### **Sensor input**

**Number of inputs** 32 (power supplies of inputs 25-32 are also outputs)

**Range** 0 to 5 Volt

**Resolution** 12 bit or 7 bit (user selectable)

**Sampling interval** programmable from 4 ms (minimum) - 16380 ms (maximum) in steps of 4 ms (actual sampling rate below)

<b>Sampling rate</b>	<i>active inputs (signal processing set to continuous)</i>								
		1	1-2	1-4	1-8	1-16	1-24	1-32	
<i>stand alone mode</i>	<i>control</i>	244	225	154	95	48	32	24	Hz (max)
	<i>pitch</i>	240	192	125	74	37	n/a	n/a	Hz (max)
<i>host mode</i>	<i>7 bit</i>	244	244	244	238	138	93	83	Hz (max)
	<i>12 bit</i>	244	244	200	149	82	56	45	Hz (max)

**Sampling latency** 1.5 ms (min) - 4.0 ms (max) (one active input without signal processing)

### **Signal processing functions**

inversion

scaling (0.04 Volt minimum range, leaving 5-bit resolution)

threshold detection (0.04 Volt minimum step)

peak/dip detection (over 16 samples maximum)

averaging (over 16 samples maximum)

differentiation (between 2 consecutive samples only)

noise gate (5 Volt maximum range)

**Supply voltage** 5 Volt

**Supply current** 30 mA per sensor (typical)

### **Actuator output**

**Number of outputs** 8 (power supplies of inputs 25 - 32)

**Range** 0 to 5 Volt

**Resolution** 1 bit (off = 0 Volt, on = 5 Volt)

**Switching rate** 1540 Hz (maximum)

**Switching latency** 1.3 ms (minimum)

**Output current** 30 mA per actuator (maximum)

### ***Communications data format***

**Host mode** MIDI system exclusive protocol

#### **Standalone mode**

MIDI pitch-bend (12 bit resolution, up to 16 inputs),  
note-on/off, key-pressure, control-change,  
program-change or after-touch (7 bit resolution)  
messages for sensor data mapping  
MIDI note-on/off, key-pressure or control-change for  
actuator output control

### ***Connectors***

**Sensor / actuator** 3 pin row (2.54 mm / 0.1" spaced)

**MIDI** 5 pin DIN (input, output with software switch-able thru)

**Power** 2.1mm / 5.0mm (inner/outer diameter) coaxial plug  
with center positive

### ***Form***

**Dimensions** 121 mm x 94 mm x 34 mm (4.75" x 3.75" x 1.3")

**Weight** 0.180 kg (0.39 lbs.)

## **Known bugs of the I-CubeX environment**

For the most recent bug listing please check our website !

### **Windows editor version 2.20**

#### **Screen redraw error**

After adding a sensor and subsequently receiving the Digitizer configuration it may occur that the workspace is not properly redrawn.

Fix: Move the lower left window corner momentarily with the mouse to initiate a screen redraw.

#### **Add sensor while Digitizer disconnected**

The editor doesn't check whether the Digitizer is connected when adding a sensor. The new sensor window may be preset for configuring a Digitizer input for which a sensor window is already present. When deleting a sensor window using Delete sensor, the editor does check whether the Digitizer is connected.

Fix: Make sure the Digitizer remains connected at all times, and if in doubt use the Send option of the Configure menu to update the Digitizer configuration.

#### **Running two instances of the application**

When running two instances of the editor on Windows 98SE the second instance will halt and needs to be ended using the task manager. On Windows NT/XP it is possible to run two instances of the application but it is unknown whether it leads to any problems.

Fix: Verify whether the editor is already running before starting it again.

#### **Minimized sensor window position changes unexpectedly**

On Windows 2000 the position of the minimized sensor windows will change if the application window size is changed.

Fix: Use the record/restore positions function to maintain the correct sensor window positions relative to the background image.

#### **Exit dialog not canceled**

On Windows 98SE, when canceling the exit dialog the application will appear to quit instead, but the task manager will show that a task named "javaw.exe" is still running.

Fix: Use "End Task" in the task manager to end "javaw.exe" in order to really end the application.

### **MacOS 9/10 editor version 2.20**

#### **Screen redraw error**

After adding a sensor and subsequently receiving the Digitizer configuration it may occur that the workspace is not properly redrawn.

Fix: Move the lower left window corner momentarily with the mouse to initiate a screen redraw.

#### **No sensor icons in minimized window**

When minimizing a sensor window it doesn't show the sensor icon.

Fix: n/a

#### **Add sensor while Digitizer disconnected**

The editor doesn't check whether the Digitizer is connected when adding a sensor. The new

sensor window may be preset for configuring a Digitizer input for which a sensor window is already present. When deleting a sensor window using Delete sensor, the editor does check whether the Digitizer is connected.

Fix: Make sure the Digitizer remains connected at all times, and if in doubt use the Send option of the Configure menu to update the Digitizer configuration.

#### **Digitizer hardware version 4.0x, firmware version 4.1**

We are not aware of any bugs. Please notify us if you think you may have encountered a bug.

## I-CubeX editor file formats

The following details the formats of some important files used by the I-CubeX editor.

### ***\*.ICU***

Files that contain an I-CubeX configuration. Configurations saved in I-CubeX editor version 2.1 or 2.2 cannot be loaded in editor version 2.0 or lower and vice versa. To convert a configuration file from an older editor to a newer one, load the configuration with the appropriate editor into the Digitizer and then obtain the configuration from the Digitizer using the newer editor.

The format is as follows:

"

Editor software version number [float eg. 2.20]  
Configuration name [8-character text, eg. MyCubeX]  
Sampling Interval value [4-16383]  
MIDI Thru [true or false]  
# of Tabs [integer]

*For each Tab:*

Tab name [text]  
*End of For loop*

*For each given Tab:*

# sensors for the Tab [integer]  
*End of For loop*

*For each sensor in a Tab:*

Tab number the sensor is in [integer]  
Sensor name [text]  
Sensor number [0-31]  
Sensor on/off [true or false]  
Preset number [integer]

*If Preset number is equal to zero:*

Threshold value [0-127]  
Ceiling value [0-127]  
Invert on/off [true or false]  
Continuous on/off [true or false]  
Peak/Dip on/off [true or false]  
Average on/off [true or false]  
End detection on/off [true or false]  
Constant on/off [true or false]  
Delta on/off [true or false]  
Samples value [0-15]  
Constant value [0=off, 15-127 in steps of 16]  
Noise value [0-127]  
*End of If statement*

MIDI map name [0-6]  
MIDI channel number [0-15]  
Note/Ctrl number [0-127]

Window position X value [integer]  
Window position Y value [integer]  
Window position X value when minimized [integer]  
Window position Y value when minimized [integer]  
*End of For loop*

# of output channels [8]

*For each Output Channel:*  
Response Mode on/off [true or false]  
Reset State on/off [true or false]  
*End of For loop*

Threshold [0-127]  
MIDI map name [0-3]  
MIDI Channel number [0-15]  
Note/Ctrl number [0-127]

*For each Tab:*  
Background Image path [text, eg. C:\My Documents\My Pictures\  
*End of For loop*  
"

**\*.TXT (or any files with recorded sensor data)**

Files that hold recorded sensor values as MIDI values use the following format (using only integers):

"time,sensor input number,MIDI channel,MIDI control number or note number or -1 (in the case of program-change, after-touch or pitch-bend),MIDI value" eg.:

"  
0,1,2,1,55  
277,2,3,64,60  
300,3,1,-1,1069  
"

This sequence of data represents a MIDI control or note number 1, controller or velocity value 55 computed from sensor input number 1, sent on MIDI channel 2 at the start of the recording, then a MIDI control or note number 64, controller or velocity value 60 computed from sensor input number 2 sent on MIDI channel 3 at 277 ms after starting recording, then a MIDI pitch bend value 1069 computed from sensor input number 3 sent on MIDI channel 1 at 300 ms after starting recording.

**PRESETS.INI**

This file holds the processing settings used as presets in the sensor window and can be edited to incorporate your own settings. The format is as follows:

"  
*first line:*  
number of lines in file [integer]  
*for each preset listed on one line:*  
Preset name [text, may contain spaces etc. but no comma],  
Threshold [0 = 0.00V, 127 = 5.00V],  
Top [0 = 0.00V, 127 = 5.00V],

Invert [0 = normal, 1 = invert],  
Continuous on/off [0 = off, 1 = on],  
Peak/Dip on/off [0 = off, 1 = on],  
Average on/off [0 = off, 1 = on],  
Constant on/off [0 = off, 1 = on],  
Constant value [0=off, 15-127 in steps of 16],  
End detection on/off [0 = off, 1 = on],  
Delta on/off [0 = off, 1 = on],  
Samples value [0-15, 0 = 1 sample],  
Noise gate value [0-127, 0 = off]CR  
"

For example the line:

"BendMini,15,110,1,1,0,0,0,0,0,0,1"

represents the BendMini setting with threshold 15 and top 110 and invert and continuous activated and the noisegate set to 1 (displayed as 0.5).

### ***ICUBEX.CFG***

This file saves the following settings for the editor:

"  
Auto update [true or false]  
System window visible [true or false]  
MIDI input device [integer]  
not used [integer]  
MIDI output device [integer]  
Actuator window visible [true or false]  
Main window top position [integer]  
Main window left position [integer]  
Main window width [integer]  
Main window height [integer]  
Main window state [integer]  
Output window top position [integer]  
Output window left position [integer]  
System window top position [integer]  
System window left position [integer]  
"



## Building your own sensors

**Warning:** Infusion Systems Ltd is not liable for any personal , Digitizer, or sensor damage due to the use or mis-use of custom (user-)built sensors.

Before constructing or testing your custom sensor, read the following points.

Things you'll need:

1. a HackCable (see our catalogue) - custom sensor connector
2. a multimeter
3. a soldering iron
4. a basic understanding of electronics
5. your sensor

Important things to keep in mind:

1. Your custom sensor's voltage range must be between 0 to 5 Volt. Any output voltage above 5 Volt may result in permanent damage to the Digitizer.
2. Your sensor should not draw more than 30 mA of current.
3. Test your sensor for correct voltages and current drain with your multimeter.
4. Connect your sensor's power input (+) to the HackCable's red (top, power) wire.
5. Connect your sensor's ground (-) to the HackCable's black (bottom, ground) wire.
6. Connect your sensor's signal output to the HackCable's white (middle, signal) wire.
7. We strongly recommend against using your own power supply (to power your sensor).

## The Sensorium

### Which sensor should I use ?

At Infusion Systems we develop the coolest bizarre ways of controlling gear. The I-CubeX comes with a Turn sensor and a See actuator to get you started and check out the list below for more possibilities!

All of our sensors are continuous control but can also act as switches (note on/note off). They are 'tuned' and pre-wired to work with the I-CubeX - no additional electronics or soldering required. Please note that the sensors and actuators listed below and their specifications are subject to change. Please see our web site for the latest available sensors and actuators or for their technical specifications.

The sensor/actuator column indicates the sensor/actuator name. The Env. (or environmental) column indicates whether the sensor/actuator is influenced by / influences the environment around it or not – thus needing to be reconfigured in the editor whenever the sensor/actuator is moved. The Interaction column describes the form interaction with the sensor/actuator. The Applications column provides some suggested uses for the sensor/actuator.

<b>Interaction</b>	<b>Env.</b>	<b>Sensor</b>	<b>Applications</b>
Airpressure	Yes	Air	Blow your brains out and see the light come on. Measures from 15 to 115 kPa (150 to 1150 mBar).
Acceleration	No	GForce2D, GForce3D	Senses acceleration and deceleration in two or three dimensions simultaneously. Attach these to Whirling Dervishes, kinetic sculptures or Fido as he chases the neighborhood cat. A visually exciting sensor. Can be held in the palm of your hand. Bonus: the GForce2D is identical to the Tilt2D so can sense inclination too ! The GForce2D detects upto 2 G, the GForce3D upto 35G which makes it a good sensor for impact (rapid deceleration) measurement.
Contact position	No	Slide, SlideLong, SlideWide	Is your keyboard lacking a cool ribbon controller? Or maybe you want 5 more... The Slide is a position sensor perfect for adding to your keyboard or DJ setup. The Slide and SlideWide ranger upto 165 mm (6.49"), but note that 2 SlideWides can be stacked on top of each other to obtain an XY touch sensitive pad. The SlideLong ranges upto 388 mm (15.3").
Contact position	No	Push, Push2D	This is the classic fader, slider (but of course we had to name it differently). Use it to tweak tweak tweak those sounds. Create your own front panel display. Performance oriented. The Push has 100 mm (3.94") range. The Push2D senses in 2 directions and has $\pm 5$ mm ( $\pm 0.2$ ") range, and is spring-loaded (ie. will return to middle position when not touched)..

Contact pressure	No	Touch, TouchMini, TouchMiniO, TouchMicro, TouchStrip, TouchStrip On	A sensor no one should do without. Contact pressure provides continuous control. Can be easily worn or applied to a variety of flat surfaces. Trigger those samples! The Touch, TouchMini, TouchMicro and TouchStrip all sense from 1 N (0.1 Kg, 0.2 lb) all the way upto 100 N (10 Kg, 23 lb) respectively on a square surface of 37.5 mm (1.48") x 37.5 mm (1.48"), on a 13 mm (0.51") diameter disk, on a 5 mm (0.2") diameter disk and along a strip of 60.9 cm (24.0") length and 13 mm (0.50") width. The TouchMiniOn is the low-res version of the TouchMini. Triggers at 1 mm (0.04") travel or 2.0 N (0.20 Kg, 0.45 lb) weight. The TouchStripOn is the low-res, stomp version of the TouchStrip. Triggers at 1.5 mm (0.059") travel or 3 N (0.3 g, 0.7 lb) weight. Both low-res versions are suitable for control freaks who think button, knob, button, slider, foot switch, ...
Contact pressure	No	TouchGlove	A controller for drum machines at your fingertips. A very expressive and natural sensor. Includes 6 pressure sensors per glove - five on the fingertips and one on the palm, each 13 mm (0.51") in diameter and capable of sensing from 1 N (0.1 Kg, 0.2 lb) to greater than 100 N (10 Kg, 23 lb).
Distance	Yes	FarReach	Create a huge presence. Ultrasonically senses distances of objects, persons, walls from 0.15 to 12 m (0.49 to 39 ft.). Is invisible to the audience. Not suitable for low volume sound installations.
Flexion	No	BendShort, BendMini, Bend Micro	Bend your fingers and control your lights. Goes from 0° to 180° (larger flex angle possible). BendMicro, BendMini and BendShort have different lengths.
Humidity	Yes	Hot&Humid	Let's get steamy, huh ! Measures both temperature and humidity. Can you handle temperatures from 0° upto 60° C (32° to 140° F) and a relative humidity from 10 % all the way to 95 % ?
Illumination	Yes	Flash	Detects rapid light changes or light flashes, including infrared, within a narrow field of view. Create a light beam switch using the SeeLaser or SeeIR that triggers when you cross it or use it as a continuous controller to sweep your filters. Detects upto 1000 Lux (5.0 mW/cm <sup>2</sup> ) within a 20°-wide cone field of view.
Illumination	Yes	Light	Responds to white light from a wide viewing angle. Put fish in a bowl with lights below and let the fish shadows play (fish not included). Detects upto 10,000 Lux (50 mW/cm <sup>2</sup> ) within a 180°-wide cone field of view.
Inclination	No	Tilt2D, TiltOn	The Tilt2D senses inclination in two dimensions simultaneously. Attach this sensor to your head to find out if you still can walk straight Saturday night at 4 am (or whether you're actually balancing on your head). Bonus: it is identical to the GForce2D so can sense acceleration and deceleration too ! Tilt2D senses a full 360° in two tilt axes, TiltOn only detects whether something is tilted or not (ie. on/off detection). It is a low-res tilt sensor. Triggers at 3° tilt (in one direction).
Magnetism	Yes	Magnetic	This sensor will show you the mysteries of magnetism. Range more than ± 1000 Gauss.

Movement	No	MoveOn	Detects movement within a range of 3 m. Be there not or be I-CubeXed.
Proximity	Yes	Close	A touchless sensor. Re-construct the front panel of Star Trek's USS Enterprise. Works from 25 mm (1.0") to 175 mm (6.9"). Sensitive to ambient lighting conditions !
Proximity	No	ReachClose	A sensor for waving at your gear. Leave your fans in awe of your magic majestic gestures. Reaches up to 80 cm (31") distance. Insensitive to ambient lighting conditions.
Proximity	Yes	Reach	One of our most popular sensors which work along the principle of the Theremin. Put your hands into its electromagnetic field while mapping it to Pitch Bend and create spooky movie music. Ranges upto 15 cm (5.9").
Rotation	No	Orient	Where do you want to go ? No, this is not some Microsoft gadget. Find east, west, south, north with this sensor.
Rotation	No	Turn	Tweak tweak tweak those sounds. Comes with funky black knobs. Perfect for your own front panel display. Performance oriented. Turn from 0° to 300°.
Temperature	Yes	Hot, Hot&Humid	Our "Mood Ring" sensor. Let your own body heat play your sequences or be adventurous and let the number of people in a room increase the heat. Warning: does not change color! Detects from -40° to 100° C (-40° to 212° F). The Hot&Humid also measures humidity.
Weight	No	TapTile, TapOn	For sensing walking, jumping and yes.. tapping. Can be covered to hide the tile. Put it on the dance floor or underneath your door mat. Senses from 45 to 1000 N (4.5 to 100 Kg, 10 to 227 lbs). Tapon just detects whether pressure is applied (ie. on/off detection), while the TapTile measures weight. It is the low-res, compact version of the TapTile. Suitable for control freaks who think knob, button, slider, foot switch, ... Requires 7 mm (0.3") travel or 20 N (2 Kg, 4.54 lb) weight to trigger.

<b>Interaction</b>	<b>Env.</b>	<b>Actuator</b>	<b>Applications</b>
Illumination	No	See, SeeIR, SeeLaser	An LED which provides visual feedback. Use it to monitor your MIDI messages or as a hi-tech Christmas tree decoration. Available in a green as well as (invisible) infrared. The SeeLaser is a laser LED actuator which works excellently in conjunction with the Flash sensor to make trigger beams that control your interactive music piece. Possibly the most unlikely way a diamond thief will find out about the captive powers of the I-CubeX. The SeeIR can also be used in this way with the Flash to make invisible beams (of shorter length though than the SeeLaser).
Motion	No	Switch	A low power control signal switcher. Advance your slide projector with each MIDI signal. Turn on your fan, your lights, your appliances, your fog machine, your micro wave. Handles at most 200 V DC at 1 A.

The above list of sensors and actuators is subject to change without warning. For information on our latest sensors and other products, contact us directly or check our web site at

<http://InfusionSystems.com>

## Index

wire 27

.

.ICU 30

+

+1 604 983 3640 1

### A

About 24

Actuator 20, 19, 18, 17, 27, 28

actuators 17, 23

Add sensor 29, 12

analog 1

Average 30, 14, 12

### B

Base Note 20, 17

bit 28

### C

cable 3

Check connections 11, 4

Close 32

Config 21

configuration 7, 10, 25, 30, 21

connection 3

Constant 30, 29, 12

Contact position 32

Continuous 30, 14, 12

control voltage 1

Copy 9

Ctrl+D 10

Ctrl+E 8

Ctrl+N 8

Ctrl+Q 8

Ctrl+S 8

.TXT 30

Acceleration 32

actuator output 5

actuators window 20, 19, 18, 17

after-touch 15, 12, 28

Auto update 10

averaging 28

Bend 32

Channel 20, 12

CLEAR CONFIG 29

Communication 28

Config name 21

configure menu 10

Connector 27, 28

Constant 14

Contact pressure 32

Continuous 14

control-change 15, 12, 28

Ctrl+C 9

Ctrl+Del 10

Ctrl+Ins 10, 12

Ctrl+O 8

Ctrl+R 10

Ctrl+T 9

Ctrl+U 9  
Cube-Culture@InfusionSystems.com 1  
CustomerService@InfusionSystems.com 1

Ctrl+V 9  
custom 31  
Cut 9

## D

Delete sensor 10  
difference 14  
Digitizer 3, 28  
Dip 30, 29, 14, 12, 28

Delta 30, 14, 12  
differentiation 28  
Dimension 28  
Distance 32

## E

edit menu 9  
End 30, 12  
Error 25  
Exit 8, 29

electronics 31  
End 14  
Error message 25

## F

FarReach 32  
firmware 29, 28  
Flexion 32

file format 30, 8  
Flash 32

## G

GForce2D 32

ground 27

## H

HackCable 31  
help 24  
Help topic 24  
host 10, 26  
Hot 32  
Hz 28

hardware 27, 29, 28  
help menu 24  
Hide 22  
host mode 28  
<http://InfusionSystems.com> 1

## I

icon 29  
I-CubeX 1  
I-CUBEX.CFG 30  
image 22  
Inclination 32  
input 27, 13, 28  
Invert 30, 29, 14

iCube 26  
ICUBEX.CFG 30  
Illumination 32  
image menu 22  
Infusion 1  
inversion 28

## K

key-pressure 15, 12, 28

## L

Light 32

## M

MacOS 29

message 17

MIDI byte 25

MIDI channel 30

MIDI Connections 10, 11, 4

MIDI feedback loop 3

MIDI input 3, 11, 27

MIDI map 20, 17, 30, 15, 12

MIDI mixer 6

MIDI patch bay 3

MIDI thru 3, 10, 4

millisecond 21

minimized 22

Monitor 16, 12

Motion 32

MoveOn 32

Max 26

MIDI 28

MIDI cable 3

MIDI communication 4

MIDI driver 4

MIDI gear 5

MIDI interface 3, 10, 11, 28

MIDI message 20, 11, 4, 26

MIDI output 3, 27

MIDI protocol 25

MIDI translator 1

Minimize 29

Mode 10, 4

moon 1

Movement 32

multimeter 31

## N

Name 13, 12

Noise 30, 14, 12

note velocity 16

note-on 15, 12, 28

New 8

noise gate 30, 14, 28

note-off 15, 12

## O

oCube 26

Orient 32

Output current 28

Open 8

output 27, 28

## P

Paste 9

peak 30, 14, 28

position 30, 22

Preset 30, 14, 12

processing 14

Pea 12

pitch-bend 15, 12

Power 3, 27, 28

PRESETS.INI 30

program-change 15, 12, 28



Proximity 32

Push 32

## R

Range 28

ReachClose 32

Receive buffer 25

redraw 29

reset values 19

Response 18, 17, 21

Restore 22

Reach 32

Receive 10

Record 8, 22

Reset 10, 24

resolution 28

Response mode 18, 17

Rotation 32

## S

Sales@InfusionSystems.com 1

Samples 30, 14, 12

Sampling interval 30, 28, 21

Sampling rate 28

Save 8

scramble 25

See 32

Sensor 10, 27, 29, 15, 16, 13, 14, 12, 32, 28, 5

sensor input 5

Sensorium 32

Show 22

signal processing 28

soldering 31

Stand alone 10, 26, 28

Supply voltage 28

Switching latency 28

sysex ID 10

system window 21

Sample interval 29

Sampling 21

Sampling latency 28

sampling speed 21

scaling 28

Screen 29

Send 10

sensor 15, 16

sensor window 15, 13, 14, 12

setup 3

signal noise 13

Slide 32

specification 28

Supply current 28

Switch 32

Switching rate 28

System 10, 23

## T

Tabs 9

Tech@InfusionSystems.com 1

Threshold 20, 17, 30, 14, 12, 28

Tilt2D 32

Top 30, 14, 12

TouchGlove 32

Turn 32

Tactile 32

Temperature 32

Tile 23

Toggle 18, 17

Touch 32

Trigger 18, 17

## U

unstable 29

## V

visualization 9

Volt 28

## **W**

Weight 32, 28

window menu 23

wire 27

window 23

Windows 29