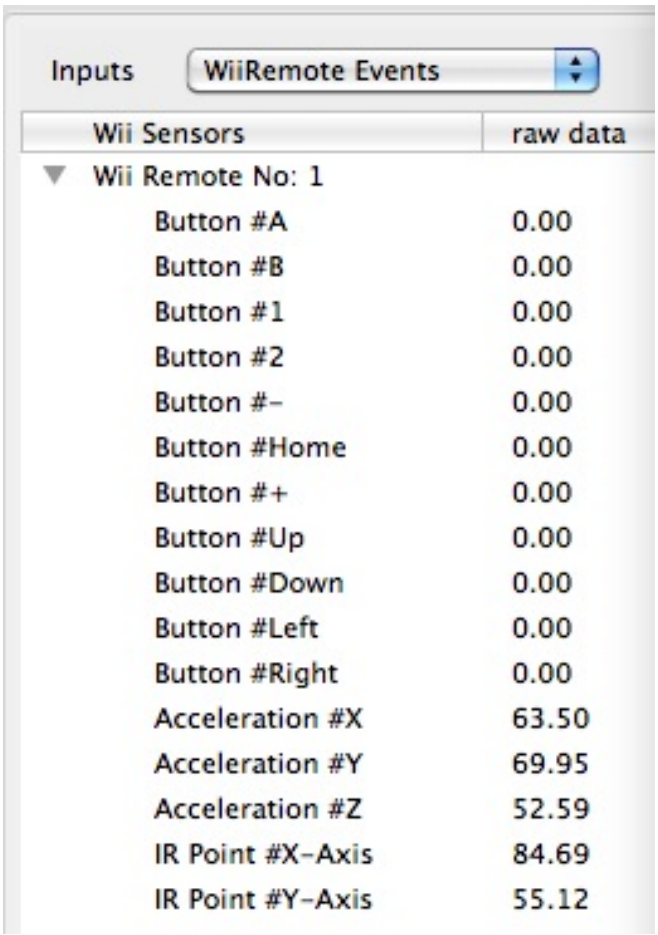
Welcome to junXion version 4.5! This document describes all the additions and changes since v4.1 and contains all the info you need if you are already familiar with v4.1
If you are new to junXion, please check out the junXion Manual first!

Input Sensors

• WiiRemote Events

IR tracking

IR sensors supported now, either use IR bar or 2 candles, if the IR source is bright enough, junXIon will be able to use the X and Y position of the IR tracking as an Input Sensor. Keep in mind that this works better at longer distances, so it is best to put the IR source at least 2 meters away. The angle of view of the IR camera in the Wii remote is small, so the movement range you can work in is limited.



• Arduino Events

General

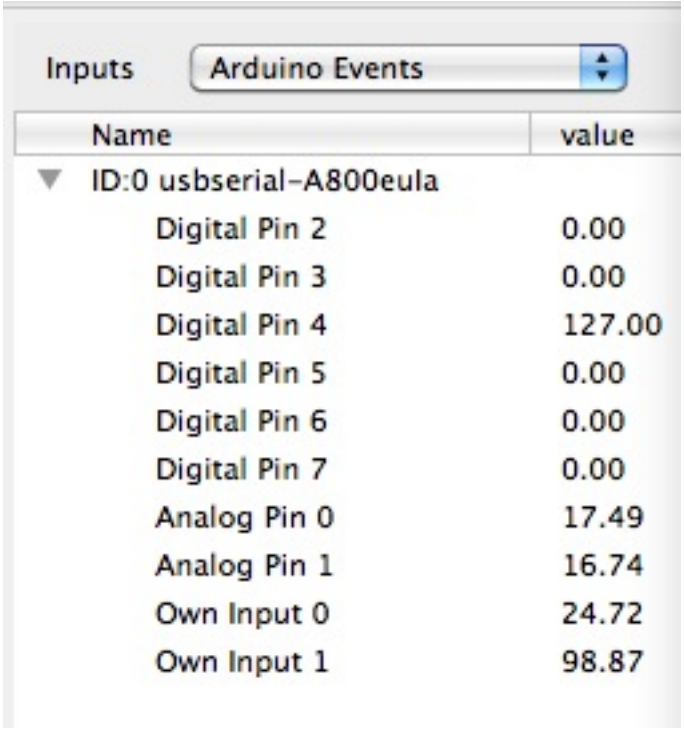
Arduino support has been greatly improved and enhanced in junXion v4.5. Now all the available Arduino boards are supported and you are provided with the sketches needed for the board you use. Before you can use your new Arduino board communicating with junXion v4.5, you will need to upload the latest Arduino2junXion sketch into the board. Some new boards are registering as a virtual modem port as soon as you connect them to your Mac computer, most of the older boards use the virtual serial port driver software, called the FTDI driver. When you download the latest Arduino package from the site http://www.arduino.cc/ it also includes the FTDI driver.

Before you start using the Arduino software make sure you have installed the FTDIUSB serial driver if needed for your board. After that, you can connect your Arduino board to a USB port. Now from the junXion v4 folder find the Sketch file named Arduino2junXion_xxxx.pde, where xxxx refers to the type of board you have, 6a14d means 6 analog and 14 digital inputs, and double-click on this file to start up the Arduino program. In the Arduino program make sure to select the right serial port (see the Arduino documentation for more info), compile the Sketch and upload it to the board. Your Arduino board is now ready for communication with junXion v4.5 and you can quit the Arduino program.

The communication speed between junXion and the Arduino boards has been enhanced a lot by optimizing the data packages, this results in lower latency.

Advanced

If you feel at home in creating Sketches, you may want to optimize the Arduino2junXion sketch. For example in the Arduino sketch you can define which inputs you are using in your sensor setup, thus optimizing your data traffic. If you are using multiple Arduino boards, the Sketch can be used to give each board a unique ID which will be used as a reference in junXion v4.5. For more info, see the comments in the Arduino2junXion_xxxx.pde sketch.



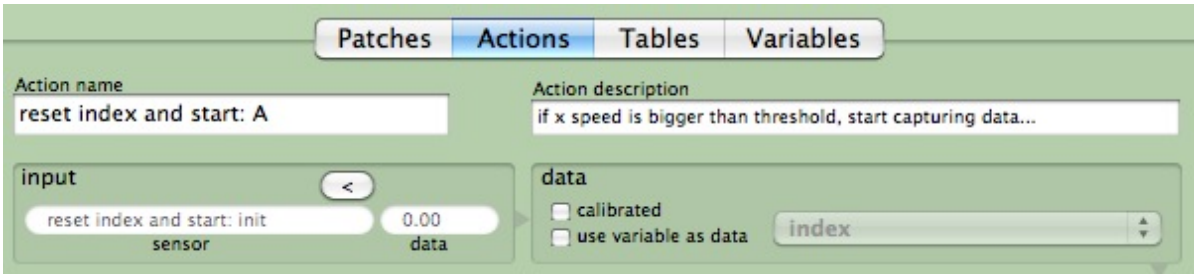
The ‘Own Input’ can be used to send internally processed Arduino data to junXion v4.5, see the Sketch comments for more info.

Actions

- **Changes**

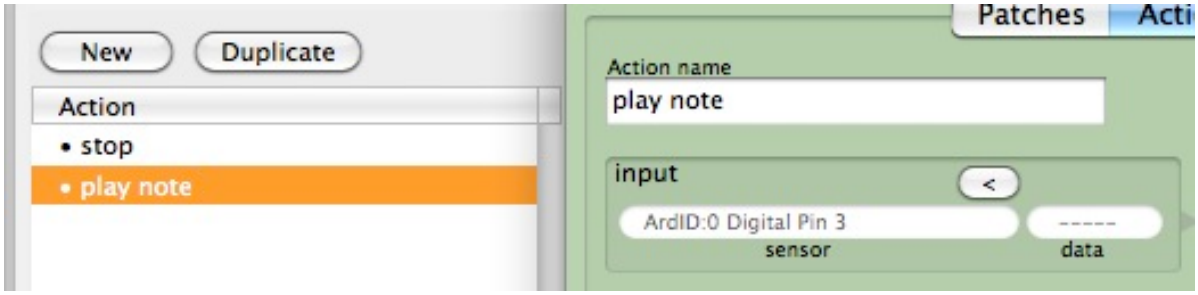
Action description

each Action now has a description field, this will be stored in the Configuration as well. It allows you to write a little comment what the Action is doing, etc. This is a handy feature (when used) to document your Configuration.



input

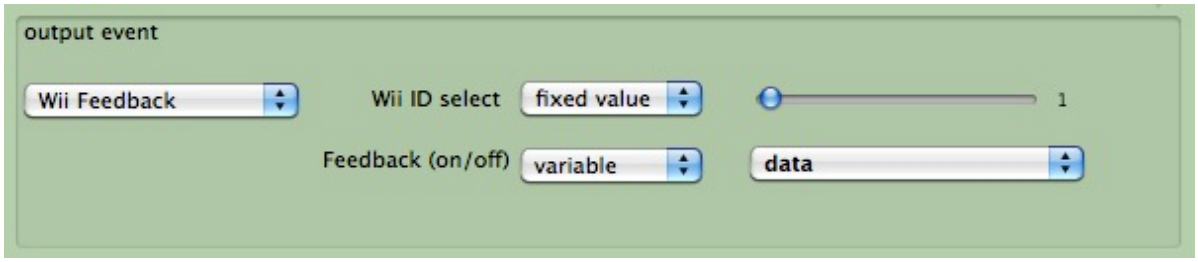
The Input sensor box will always show an Input Sensor if one is connected to the Action. Even if multiple Input Sensors (ISR's) are connected to the same Action, it will show the name of the last connected ISR.



Another situation is when an Action is used in a State that is not currently the active State. Also in this case the ISR will be shown, but without data.

output event

As an extra **output event** in the Actions view, you can now select the menu item **Wii Feedback**.

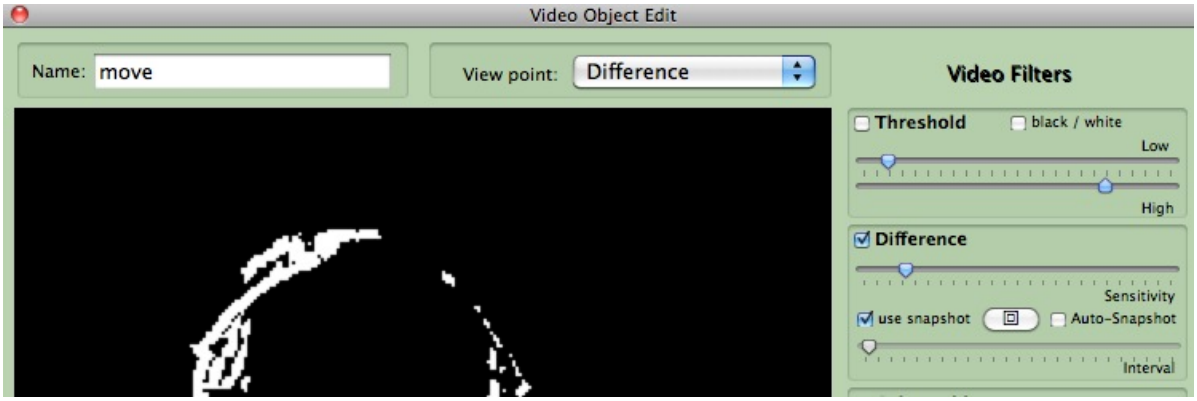


The Feedback parameter has only two values, ON or OFF. With the Wii ID select parameter you can choose which Wii to send the Feedback event to, this is in the range of 1 to 4. Any message send to a non-connected Wii will be ignored.

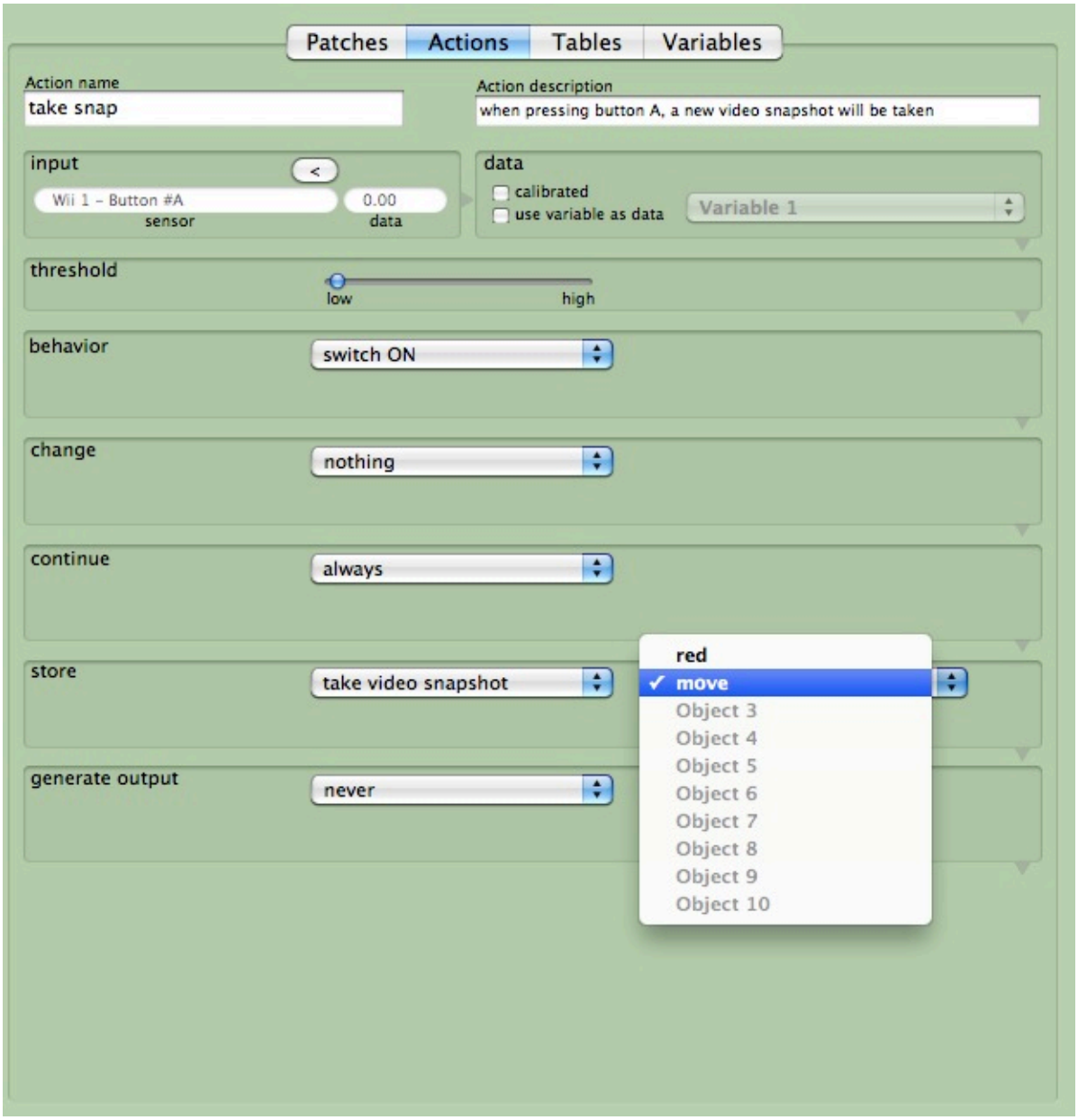
store

when using Video Input, with the Difference filter, there is a possibility now to take a new 'Snapshot' with another ISR. The **store** menu in the Action contains a new menu item named *take video snapshot*. The best way to do this is to connect a switching ISR to this Action, so that everytime the switch changes its state (on/off), a new snaphot will be taken in the selected Video object.

For this to work properly, the selected Video Object has to use the Difference filter. Also the *use snapshot* box has to be checked, see picture.



In the following example, Button A of a WiiRemote game controller will be used to take a new snapshot for this Video Object.
Note that for its **behavior**, this Action is set to *switch on*, meaning that only when the button is pressed (not released), a new snapshot will be taken.

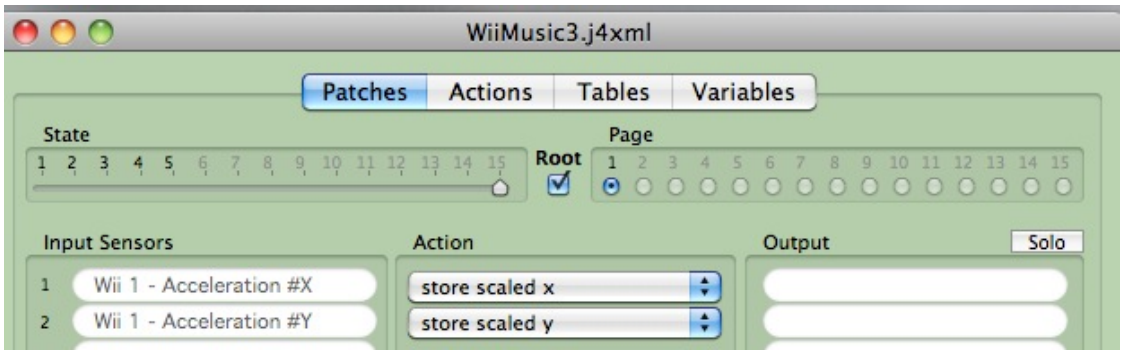


Patches

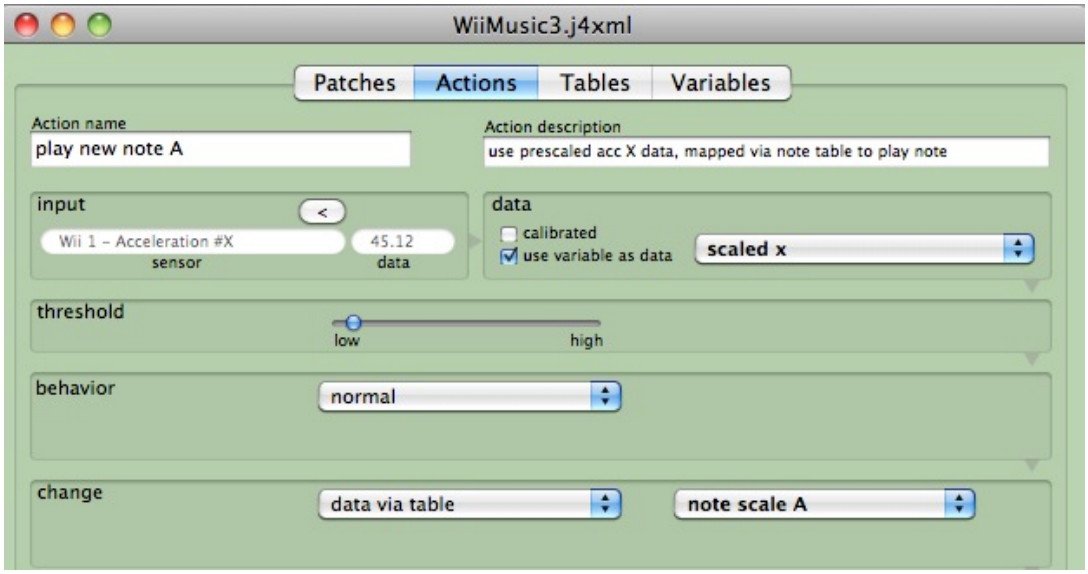
- **Changes**

Root State

added to the 15 States is the so called Root State. This State is special, since it is always active, no matter which of the other 15 States you are in. The idea behind this is that you can do ‘global’ processing stuff in this State, and the other States can then be used for specific settings. The Root State can be selected by clicking on its checkmark box, this will display that State and disable the State slider.



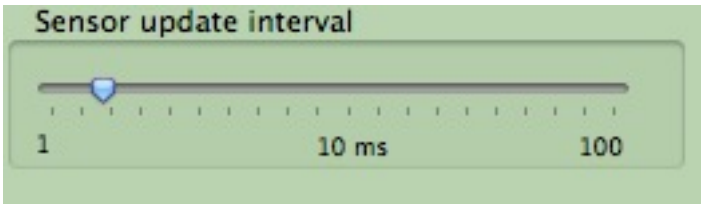
now you can add the Patches that you want always use, no matter which other State you are in. If you are only using one State, there is no need to use the Root State of course.
Once you uncheck the Root box, junXion v4.5 will show the current ‘normal’ State you are in and if Patches have been made in the Root State, the box’s name above it will be shown in bold, **Root.** This is visual reference for you that the Root State contains active Patches.
A typical example of Root State usage could be that you use this State with Patches that are just scaling your ISR’s into the full data range, store those values into **Variables** and then in all the other Patches in other States, you use those Variables as the input, instead of the raw ISR data.



Preferences

• Sensor update interval

this slider allows you to change the default update interval time of 1 millisecond into a higher number. In a lot of cases it is not needed to check for new sensor data every 1ms, and by choosing a higher interval time there is less CPU load by junXion 4.5.

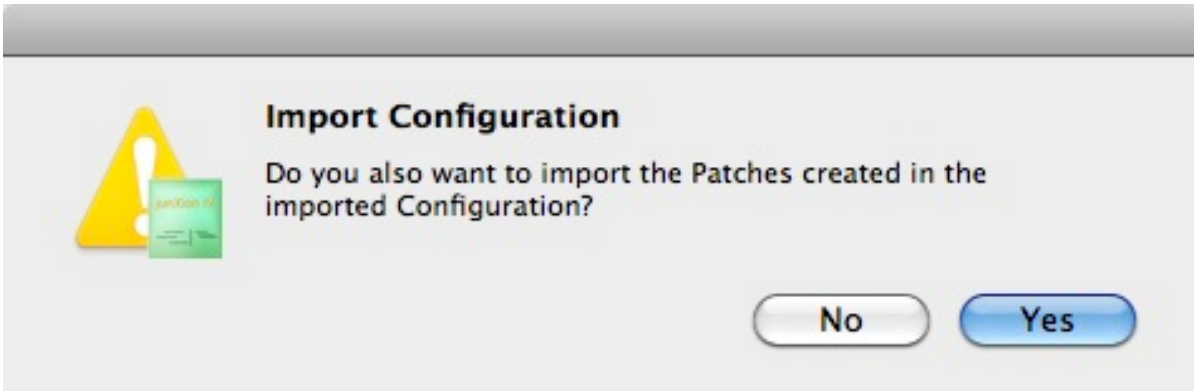


Menus

• File

Import...

This feature allows you to merge an existing Configuration with your current Configuration. When selecting this menu item, junXion v4.5 will ask you if you want to import the whole Configuration (including the Patches) or just its Actions, Tables and Variables.



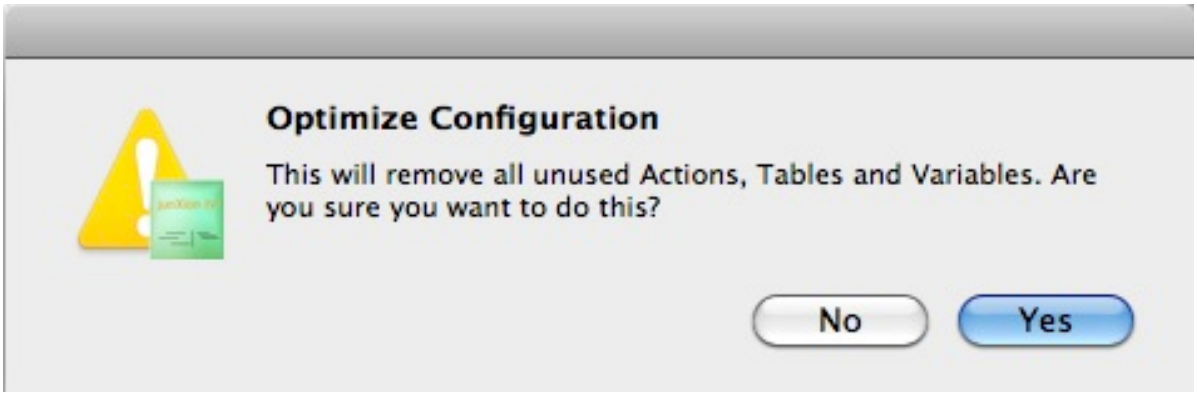
If you don't want to import the Patches by selecting No from the above dialog, only the Actions, Tables and Variables as used in the to import Configuration will be added to your existing one. Sometimes this is a better option, because if your existing Configuration and the to be imported one both use multiple States, things can get confused. Generally speaking this feature is very handy to merge an example set of Patches with your existing ones.

If you want to combine 2 of your own Configurations, it is strongly adviced first to optimize both Configurations, using the **Optimize Configuration...** feature.

- **Edit**

  Optimize Configuration...

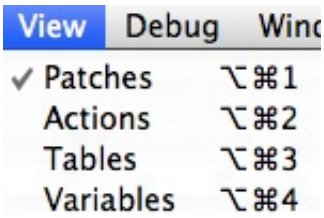
  when selecting this menu from the Edit menu, junXion v4.5 will look at you current Configuration and ask you if you want to remove all unused Actions, Tables and Variables.



  This is a neat way of cleaning up your Configuration because all the unused stuff will be deleted. How does junXion v4.5 decide what is used and what not? By first checking all its Patches for all States, thus knowing which Actions are being used then checking which Tables and Variables are being used by those Actions and finally marking all the other Actions, Tables and Variables to be deletable.

- **View**

  Added keyboard shortcuts



  quick navigations between the four different Views in junXion v4.5 is now possible by using keyboard shortcuts.

- **Debug**

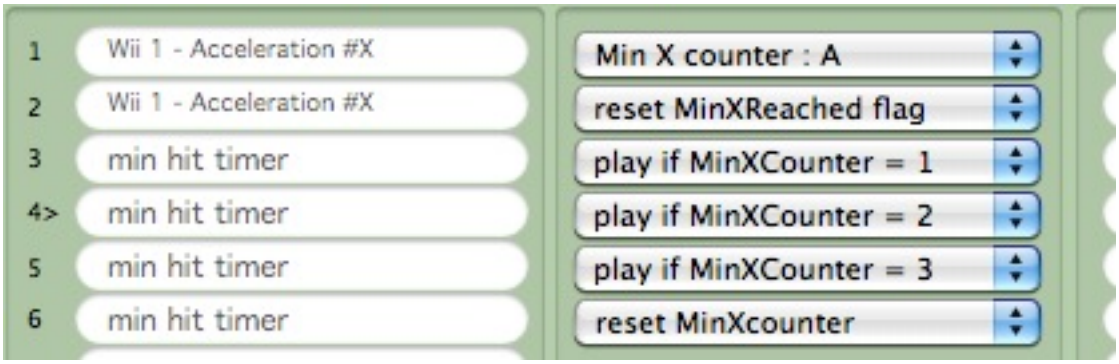
  sometimes, to figure out why something is not working as you think it should, it is handy to manually step thru each of the Patches, instead of having these Patches be executed a thousand times per second. To use this feature, a new menu with three new menu items have been added to the program, resulting in the **Debug** menu.

Stop

By selecting the **Stop** menu item, junXion v4.5 will stop continuous execution of the Patches that are used. This is also handy sometimes to temporarily put the program on hold, or sleep mode, for example if you want to quickly stop all your sequencing Timers.

Single Step

When the **Stop** menu item has been selected, the **Single Step** item will be enabled allowing you to step thru the Patches one by one. junXion v4.5 will show which Patch you will execute with a little > sign before the Patch line.



Run

to go back to normal running mode again.

General

• Closing a window

Closing the normal or the zoomed (Status) window will assume you want to quit junXion and ask you if you want to save your changes. This means that at least the Status window always will be visible.

• Quitting junXion

When you have been using an Arduino board in your Configuration and at some point quit junXion, it will take a little longer to close down the program because the virtual port shutdown time is quite slow. This is normal.