SYNTHMASTER

VERSION 1.0.4

Credits

Concept, Design, Programming and Documentation by *Bülent Bıyıkoğlu*
User Interface and Web Development by *Umut Derviş*
Beta Testing and Demo Songs by *Selçuk Ergen*
Presets by *Teoman Pasinlioğlu* and *Aykut Şahlanan*

Many thanks to our dear friends at Muziktek.net: *Mahçem Öztürk, Cemal Öztürk, Doruk Somunkıran, Batuhan Bozkurt, Burak Tülbentçi, İlter Kalkancı and all the others…*

Copyright

For my lovely wife Elif Uysal-Bıyıkoğlu,

--B.B.

1 Introduction

SynthMaster is semi-modular software synthesizer and effects plug-in that combines many different synthesis methods such as additive synthesis, subtractive synthesis, ring modulation, amplitude modulation, frequency / phase modulation, pulse width modulation, osc sync and waveshaping.

SynthMaster consists of the following modules / sections:

- **3 Additive Oscillators**: Each multi-waveform oscillator has its own volume / frequency (pitch) envelope.
- **3 Multi-mode Filters**: Each filter has 23 different modes ranging from 1 pole (6dB/oct) lowpass to 4 pole (24 dB/oct) resonant self-saturating lowpass. Filters can be connected in three different configurations: Series, parallel and split.
- **4 Modulators**: The modulators can be used as LFOs (low frequency oscillators) or audio rate oscillators to modulate other modulators, multi-mode filters or the additive oscillators. Each modulator has its own envelope.
- **2 MIDI LFOs**: These 2 LFOs are used to generate MIDI CC messages to modulate instrument parameters. Since each CC can be linked to 16 instrument parameters, it is possible to modulate 32 different instruments parameters at the same time using the MIDI LFOs.
- **2 32 step Patterns**: The patterns could be used as waveforms for modulators or MIDI LFOs. Since the number of steps is user adjustable, it is possible to generate patterns in rhythms other than 4/4, such as 9/8, 8/7, 5/8, etc.
- **16 Step Arpeggiator**: The arpeggiator in SynthMaster is like no other arpeggiator you have seen before. Every step has its own volume, length and delta. The number of steps is user adjustable, so it is possible to generate arpeggios in rhythms other than 4/4, such as 9/8, 8/7, 5/8, etc.
- **8 Band parametric EQ**: The EQ could be constructed in different combinations such as parallel, series, or input + series. Each band in the EQ could be bypassed, which makes it possible to construct a very flexible parametric EQ.
- **Echo / Delay**: The echo effect has feedback and filter cutoff parameters, and its delay size is tempo synced.
- **16 Band Vocoder**: The vocoder uses analog modeled bandpass filters. Either the synth output or the audio input can be used as carrier/modulator signal for the vocoder.
- **Stereo Chorus / Flanger**: The chorus effect in SynthMaster has separate delay and rate controls for the left and right outputs, which widens the stereo width of the effect output.
- **Stereo Reverb**: The reverb effect in SynthMaster is based on physical modeling. You can specify physical parameters such as the room size, the distance of the left & right sound sources, and the width between the sound sources.

Since *SynthMaster* is also an effect plug-in, it is possible to apply all the effects (EQ, Vocoder, Echo, Chorus, Tremolo and Reverb) to the audio input signal as well. This means that you can use it not only as a software synthesizer, but also as a real-time audio effects processor.

Before discussing *SynthMaster* in more details, we would like to take this opportunity to thank you for choosing a KV331 Audio product. We sincerely hope that this product will bring you inspiration, pleasure and fulfill your creative needs.

1.1 System Requirements

- Windows XP Operating System
- 500 Mhz Pentium 3 or better processor
- 256 MB RAM
- 1024 x 768 or higher screen resolution
- For RTAS version: ProTools 7.0 or above
- For VST version: A host application that supports VSTi 2.0 or above extensions.
- For Dxi version: A host application that supports Dxi soft-synths
- ASIO supported sound card. ASIO is necessary to achieve low latency. If your onboard sound card does not have ASIO drivers, you can download ASIO4All at http://www.asio4all.com/. The standalone version of SynthMaster works with sound cards that don't have ASIO drivers as well, but in that case latency will be high, and real-time audio input will not be available in the standalone version of SynthMaster.
- MIDI Keyboard (recommended)

1.2 Installing SynthMaster

Installing *SynthMaster* is straightforward. Double click on the SynthMasterSetup.exe file that you downloaded from KV331Audio.com web site. That will bring up the setup wizard, which will guide you through the steps required to install *SynthMaster*:

1.2.1 Installation Paths for Presets and VST version of SynthMaster

When SynthMaster is installed, some files will be installed under fixed paths:

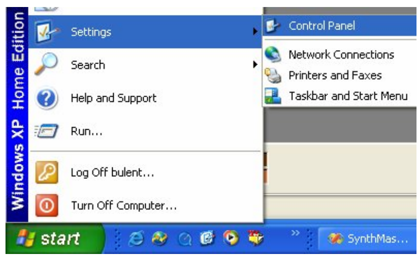
Presets	*<Common Files>\Digidesign\DAE\Plug-In Settings\SynthMaster*
Easy Parameter Presets	*<Common Files>\KV331 Audio\SynthMaster\Easy Parameter Presets*
Tuning Presets	*<Common Files>\KV331 Audio\SynthMaster\Tuning Presets*
Midi Links Presets	*<Common Files>\KV331 Audio\SynthMaster\Midi Links Presets*

Please note that the standalone, VST and VSTi versions of SynthMaster are all installed to the installation folder selected by the user.

1.3 Uninstalling SynthMaster

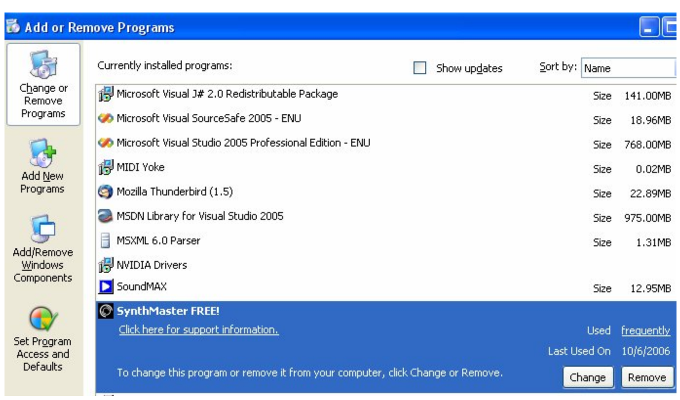
To uninstall *SynthMaster*:

- Click on Start->Settings->Control Panel. That will bring up the Control Panel:



- Double click on the “Add or Remove Programs” icon. That will bring up the “Add or Remove Programs” window:

- Scroll down the list of installed software on your computer, and click on “Remove” next to SynthMaster:



- Click on the “Yes” button on the confirmation dialog, that will remove *SynthMaster* from your computer:

1.4 Authorizing SynthMaster

IMPORTANT NOTE: Authorization can only be performed by users that are member of the *Administrators* group on the computer *SynthMaster* is being installed.

When you run any version of *SynthMaster* for the first time, the authorization window will appear. You will have to enter your serial number and authorization code to authorize *SynthMaster*, so that you can use it without any limitations:

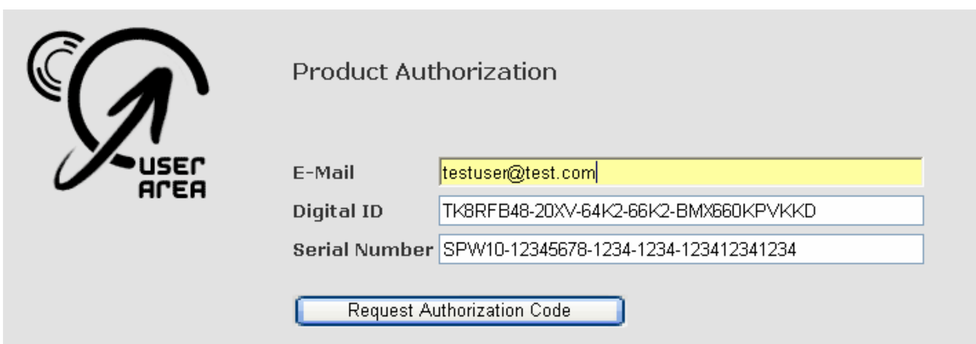
The Digital ID: TK8RFB48-20XV-64K2-66K2-BMX660KPVKKD you see on this window is a series of numbers and letters that uniquely identifies your computer. It will be regenerated (and hence you will be required to re-authorize) whenever:

- You re-format your hard disk
- You upgrade your processor

Serial number is a series of numbers and letters that uniquely identifies the product you purchased. It is e-mailed to you when you purchase and download *SynthMaster*. To authorize *SynthMaster*, you will have to enter the serial number on the text box next to Serial Number:

Authorization code is a series of numbers and letters that is generated by combining your Digital ID and Serial Number. To obtain your authorization code:

- Enter your product serial number on the text box next to Serial Number:
- Click on the Go to www.kv331audio.com for authorization button to connect to the SynthMaster authorization page on the KV331Audio.com web site. Your digital id and the serial number you entered will be submitted to the web site so that you don't have to enter them once again:



- On this page, you should enter the e-mail address you used to purchase *SynthMaster.* Once you enter all the required information, click on "Request Authorization Code" button. If the serial number and digital id you entered are valid, your authorization code will be generated and displayed on this page, and e-mailed to you as well. Copy this code and paste it on the text box next to Authorization Code: .
- Click on the Authorize button. If the serial number and authorization code you entered are valid, they will be saved on your computer, and your copy of SynthMaster is now authorized. (On the PC, they will be saved in the Registry, under HKEY_LOCAL_MACHINE\Software\SynthMaster)

- If you are not connected to the Internet, you can click on the 

button to skip the authorization process. Please keep in mind that, in unauthorized mode, SynthMaster will have the following limitations:

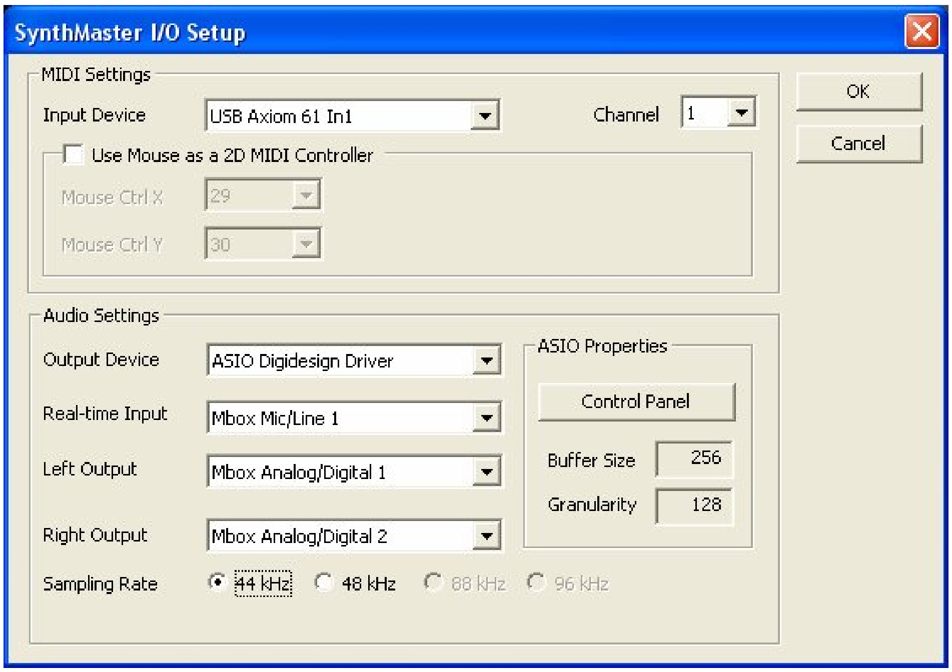
    - No more than 2 Plug-in instances are allowed
    - Polyphony for each plug-in instance is limited to 5 voices
    - Audio input is quantized to 10 bits.

2 SynthMaster Standalone

The standalone version allows you to run *SynthMaster* as a regular Windows application without the need for host application. It also supports real-time audio input, so you can use *SynthMaster* not only as a software synthesizer, but also as a real-time effects processor.

2.1 I/O Setup Screen

When you run *SynthMaster* standalone for the first time, the *I/O Setup* screen will appear. On this screen, you can select midi input and audio output devices, or change the sampling rate at which *SynthMaster* engine operates:



2.1.1 MIDI Settings

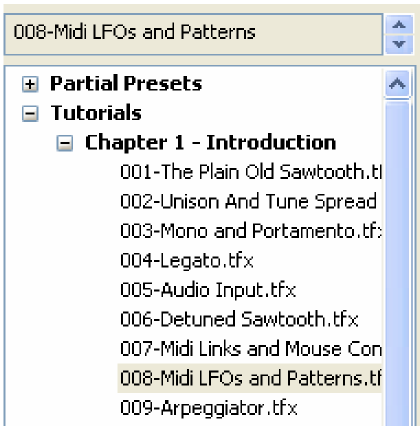
Input Device	From this dropdown list, you select the MIDI Input device. All MIDI Input devices that are available on your system are listed here. The first item you'll see on this list is "PC Keyboard". By selecting this item, you can turn your PC keyboard into a MIDI keyboard. The keys you can use for playing are: • *zx…m* : Octave 1 • *qw…p* : Octave 2 • *Up Arrow*: Octave Up • *Down Arrow*: Octave Down

Channel	Specifies the input channel, between 1-16. Only messages from the selected channel are processed.
Use Mouse as a 2D MIDI Controller *Mouse Ctrl X* *Mouse Ctrl Y*	With SynthMaster, you can turn your mouse into a MIDI controller. To enable this feature, you should check the “Use Mouse as a 2D MIDI Controller” check box. *Mouse Ctrl X* specifies the controller number for the X coordinate. Similarly, *Midi Ctrl Y* specifies the controller number for the Y coordinate. To activate the mouse as a MIDI controller, click on the middle mouse button. By default, *Mouse Ctrl X* is Controller 29, which is linked to ***Filter 1 Freq*** parameter. *Mouse Ctrl Y*, on the other hand, is Controller 30 which is linked to ***Filter 1 Resonance*** parameter. If you look at the Filters tab, you’ll see that whenever you move your mouse around, Filter 1 Frequency and Resonance knobs’ values are changing.

2.1.2 Audio Settings

Output Device	From this dropdown list, you select the audio output device. All audio devices in your system are listed here, including the ASIO enabled ones. If you see a name that has “ASIO” in it, make sure you select that one. With ASIO, you get low latency and also real-time audio input is supported so that you can use SynthMaster as a real-time effects processor.
Real-time Input	This is the input port name of the selected audio device. If you have any instrument / microphone connected to your audio device, make sure you select the correct input port, otherwise you will not be sampling and processing your input signal.
Left Output *Right Output*	These are the output port names of the selected audio device. By default, the first two output ports are selected. If the audio device you selected have more than 2 outputs and you are using the ones other than the first two, make sure you select those ones as output ports.
Sampling Rate	These option buttons specify the sampling rates available for the selected audio device. If you select a higher rate, latency decreases, but on the other hand the load on the CPU increases.

2.2 *Preset Browser Tree*



When SynthMaster launches, you will see the preset browser on the left hand side of the application window. Presets are categorized under folders, which are shown in bold to distinguish them from presets.

By clicking on the ⊞ icon, you can expand a folder and see sub-folders / presets under it.

2.2.1 Loading Presets

You can load a preset in various ways:

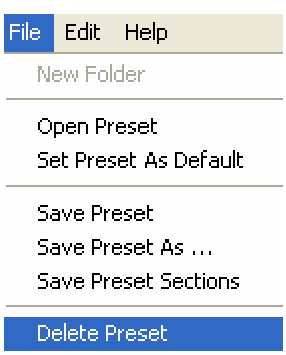
1. By double-clicking on the preset's name.
2. By clicking on the arrows to load the previous/next preset on the tree.
3. By selecting a preset on the tree and pressing the "Return" key on the keyboard.

2.2.2 Renaming Presets

To rename a preset, click on the preset's name, and then click again after a second:

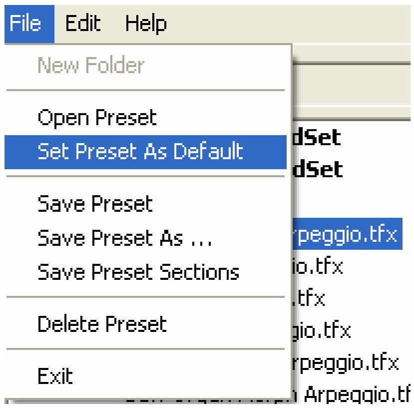
2.2.3 Deleting Presets

To delete a preset, click on the preset's name, and then choose "*Delete Preset*" sub-menu from "*File*" menu:



2.2.4 Setting a Preset as the “Default” Preset

You can set a preset as the “default” preset, so that it is loaded initially every time you start SynthMaster Standalone. To do that, click on the preset’s name, and then choose “*Set as Default Preset*” sub-menu from “*File*” menu:



2.3 *Recording SynthMaster output*

You can record the output of SynthMaster Standalone to a stereo .WAV file. To start recording:

- Click on the  button to select the output .WAV file. You can choose an existing file or enter a new file name.
- Click on the  button to start recording.
- To stop recording, click on the  button again.

3 SynthMaster User Interface

SynthMaster user interface has more than a thousand controls. Since it is not possible to fit all those controls on one page, controls are categorized under ***tabs*** that contain multiple *sections*:

- ***Easy Page***
- ***Oscillator 1:*** *Waveform, Volume Env, Frequency Env, EQ*
- ***Oscillator 2:*** *Waveform, Volume Env, Frequency Env, EQ*
- ***Oscillator 3:*** *Waveform, Volume Env, Frequency Env, EQ*
- ***Modulators 1-2:*** *General, Key Scaling, Mod Env*
- ***Modulators 3-4*****:** *General, Key Scaling, Mod Env*
- ***Filters:*** *General, Filter Env, Modulation Sources*
- ***Effects 1:*** *Parametric EQ, Chorus, Reverb, Echo, Vibrato, Tremolo, MIDI LFO 1, MIDI LFO 2*
- ***Effects 2:*** *Pattern 1, Pattern 2, Arpeggiator, Vocoder*

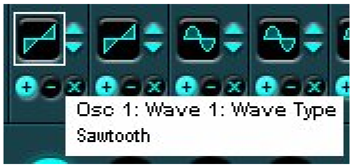
3.1 Control Types

On the user interface, you'll see different kinds of controls. Each control is linked to an instrument parameter:

	***Rotary knobs*** are used for parameters that have continuous values.
	***Sliders***, similar to rotary knobs, are used for parameters that have continuous values.
	***Toggle buttons*** are used for discrete parameters that can take only 2 values.
	***Group buttons*** are used for discrete parameters that can take 3-4 values.
	***Up/Down buttons*** are used to increment/decrement the value of discrete parameters by one.
	***Dropdown boxes*** are used for discrete parameters that can take more than 2 values. To change the value of a dropdown control: • Bring the mouse over the control • Click on the left mouse button. The possible values for the parameter will be displayed on a popup menu. • Click on the new value of the parameter on the popup menu. The control will show this new value.

3.2 Tooltip Support for Displaying Control Values

When you bring your mouse over a control and hold it there for a second, a popup tooltip window appears next to the control and displays the current value of the control:



3.3 Mouse-wheel Support for Changing Control Values

You can use the mouse-wheel to change control values. For discrete controls, the value will be incremented/decremented by 1. For continuous controls, the value will be incremented/decremented by *1/128 * (<Max Value>-<Min Value>).*

3.4 Editing Parameter Values

With SynthMaster, parameters could be edited at 3 levels:

- Preset: (all parameters)
- Tab: (parameters for a specific tab: Osc1, Osc2, etc…)
- Section: (parameters for a specific section of a specific tab: Osc1: Waveform, Filter 1: Envelope, etc…)

The following editing actions can be performed from the popup menu that is displayed by clicking the right mouse button:

As you see on this screenshot, the popup menu displays the current tab/section information according to the position of the mouse. The mouse is over *Osc1: Waveform*, so the current tab is *Osc1*, and the current section is *Osc1: Waveform*.

3.4.1 Copying all Parameters

When you select the "Copy -> Preset" sub-menu item on the popup menu, the values for the entire instrument parameters are copied onto the clipboard.

3.4.2 Copying Current Tab Parameters

When you select "Copy -> *Tab Name*" sub-menu item on the popup menu, the values of all parameters for the current tab are copied onto the clipboard.

3.4.3 Copying Current Section Parameters

When you select “Copy -> *Section Name*” sub-menu item on the popup menu, the values of all parameters for the current section are copied onto the clipboard.

3.4.4 Pasting all Parameters

When you select “Paste -> Preset” sub-menu item on the popup menu, the values for the entire instrument parameters are copied from the clipboard. This menu item will be disabled if the clipboard does not contain parameters that were copied at the preset level.

3.4.5 Pasting Parameter Values for the Current Tab

When you select “Paste->*Tab Name*” sub-menu item on the popup menu, the values of all parameters for the current tab are copied from the clipboard. This menu item will be disabled if the clipboard does not contain parameters that were copied at the correct tab level that has similar data structure as the current tab.

3.4.6 Pasting Parameter Values for the Current Section

When you select “Paste -> *Section Name*” sub-menu item on the popup menu, the values of all parameters for the current section are copied from the clipboard. This menu item will be disabled if the clipboard does not contain parameters that were copied at the correct section level that has similar data structure as the current section.

3.4.7 Resetting all Parameters

When you select “Reset -> Preset” sub-menu item on the popup menu, the values for the entire instrument parameters are reset to their default values.

3.4.8 Resetting Current Tab Parameters

When you select “Reset -> *Tab Name*” sub-menu item on the popup menu, the values of all parameters for the current tab are reset to their default values.

3.4.9 Resetting Current Section Parameters

When you select “Reset -> *Section Name*” sub-menu item on the popup menu, the values of all parameters for the current section are reset to their default values.

3.4.10 Randomizing all Parameters

When you select “Randomize -> Preset” sub-menu item on the popup menu, the values for the entire instrument parameters are set to random values.

3.4.11 Randomizing Current Tab Parameters

When you select “Randomize -> *Tab Name*” sub-menu item on the popup menu, the values of all parameters for the current tab are set to random values.

3.4.12 Randomizing Current Section Parameters

When you select “Randomize -> *Section Name*” sub-menu item on the popup menu, the values of all parameters for the current section are set to random values.

3.5 Loading Presets

SynthMaster has 2 types of presets:

- “Complete” Presets, which contain all instrument parameters, preset information, midi controller links and tuning details.
- “Partial” Presets, which only contain parameters for a particular tab/section. For instance, if you save “partial” settings for Osc 1, you can later load that onto Osc1, Osc2 or Osc 3.

3.5.1 Loading "Complete" Presets

To load a "complete" preset, click the right mouse button, and then select the "Load->Preset" sub-menu item on the popup menu. This item will display the list of "complete" presets that can be loaded:

3.5.2 Loading Presets for the Current Tab

To load a preset for the current tab, click the right mouse button, and then select the "Load->*Tab Name*" sub-menu item on the popup menu. This item will display a list of presets than can be loaded onto the current tab:

3.5.3 Loading Presets for the Current Section

To load a preset for the current section, click the right mouse button, and then select the "Load->*Section Name*" sub-menu item on the popup menu. This item will display a list of presets than can be loaded onto the current section:

3.6 Saving Presets

With *SynthMaster*, you can save 2 different types of presets:

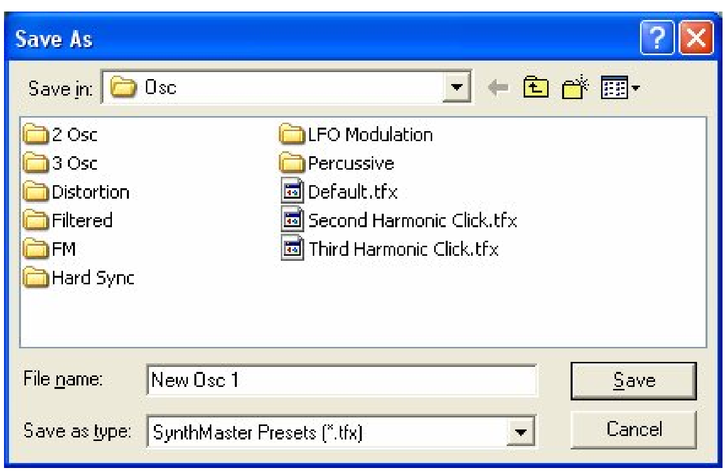
- “Complete” Presets, which contain all instrument parameters, preset information, midi controller links and tuning.
- “Partial” Presets, which might contain parameters for a particular tab/section, or multiple tabs/sections. For instance, if you save “partial” settings for Osc 1, you can later load that onto Osc1, Osc2 or Osc 3.

3.6.1 Saving “Complete” Presets

To save a “complete” preset, click the right mouse button, and then select the “Save->Preset” sub-menu item on the popup menu. This will bring up the “Save As” file browser window. Enter a new file name and hit *Save* :

3.6.2 Saving Presets for the Current Tab

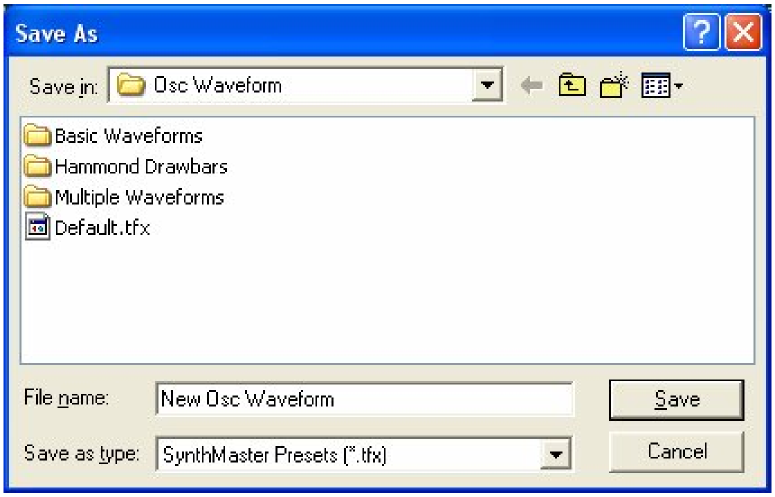
To save a “partial” preset for the current tab, click the right mouse button, and then select the “Save->*Tab Name*” sub-menu item on the popup menu. This will bring up the “Save As” file browser window. Enter a new file name and hit *Save* :



<u>Important Note</u>: Since partial presets are saved to fixed locations, do not change the directory to an upper directory!

3.6.3 Saving Presets for the Current Section

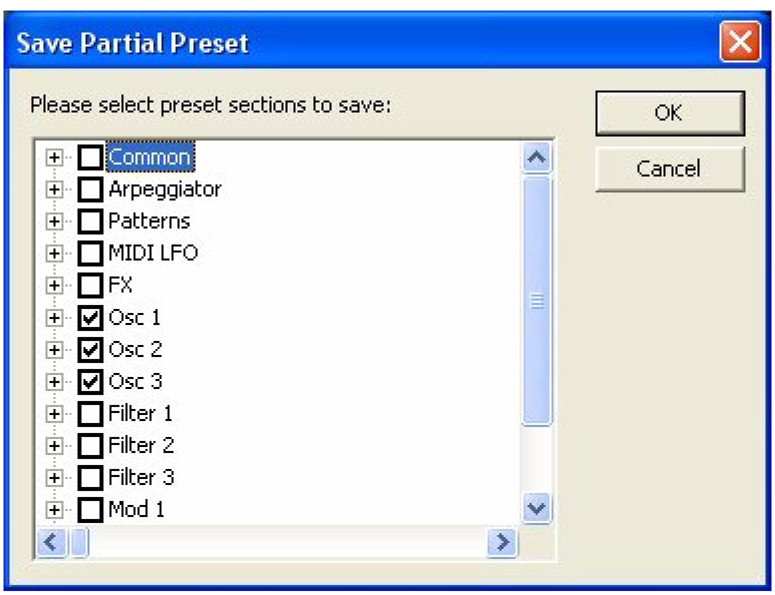
To save a “partial” preset for the current section, click the right mouse button, and then select the “Save->*Section Name*” sub-menu item on the popup menu. This will bring up the “Save As” file browser window. Enter a new file name and hit *Save* :



Important Note: Since partial presets are saved to fixed locations, do not change the directory to an upper directory!

3.6.4 Saving Multiple Tabs/Sections

To save a “partial” preset for multiple tabs/sections, click the right mouse button, and then select the “Save-> Multiple Sections” sub-menu item on the popup menu. This will bring up the *Save Partial Preset* screen. On this screen, you can select the tabs/sections that you want to save and then hit *OK* to save the selected tabs/sections as a partial preset:



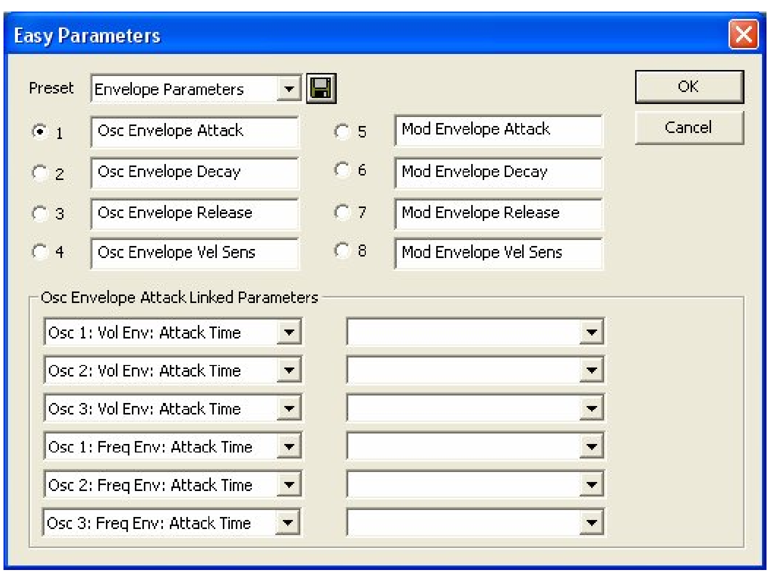
3.7 Easy Parameters

When you run *SynthMaster*, the “*Easy Page*” tab is selected by default. On this page, you see 8 knobs that can be linked to frequently used instrument parameters such as *cutoff* and *resonance*. Since one easy parameter can be linked to multiple instrument parameters, you can modify all of those parameters by just turning one knob.

3.7.1 Editing Easy Parameters

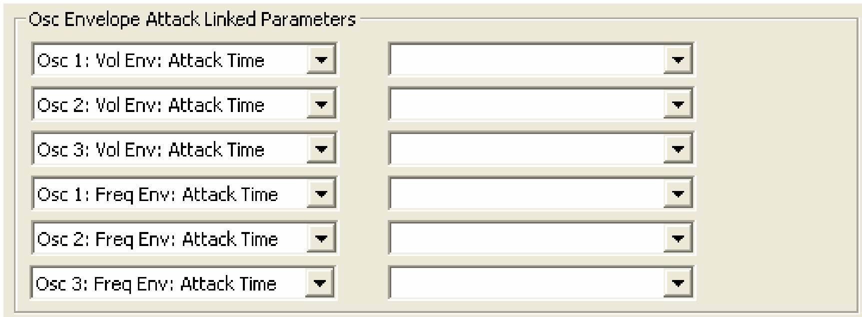
Easy parameters can be edited on the *Easy Parameters* screen, which can be accessed by clicking the 

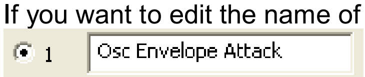
button on the *Easy Page* tab.



To edit an easy parameter:

- Click on the corresponding circle  to select the parameter. Instrument parameters that are linked to this easy parameter will be shown:



- To add a new linked parameter, select it from one of the 12 dropdown lists.
- To remove an existing linked parameter, select the first blank entry from the dropdown list.
- If you want to edit the name of the easy parameter, click on the textbox 

, and type the new name of the parameter.

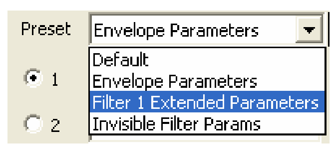
When you're done with your changes, click the *OK* button to save your changes. Easy parameter settings are saved to the SynthMaster configuration file when you exit SynthMaster, so that they will be restored back when you run SynthMaster the next time.

3.7.2 Easy Parameter Presets

SynthMaster supports saving easy parameter settings as presets, so that you can quickly access different easy parameter settings. Easy parameter presets are stored under the following location: *<Common Files Folder>\KV331 Audio\SynthMaster\Easy Parameter Presets*

3.7.2.1 Loading Easy Parameter Presets

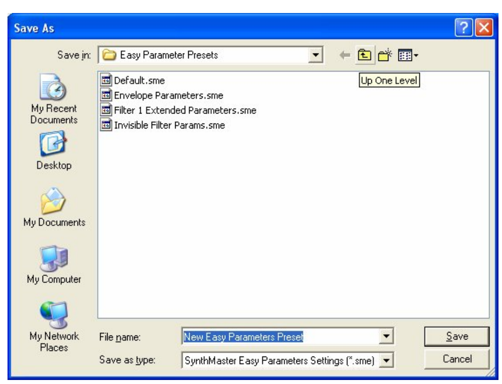
To see the existing easy parameter presets, click on the dropdown list, and then choose the preset that you want to load:



3.7.2.2 Saving Easy Parameter Presets

To save the current easy parameter settings as a new preset, click on the  button. This will bring up the "Save As" file browser window. Enter a new file name here and hit *Save* :

<u>Important Note</u>: Since easy parameter presets are saved to a fixed location, do not change the directory here or create a sub-directory.

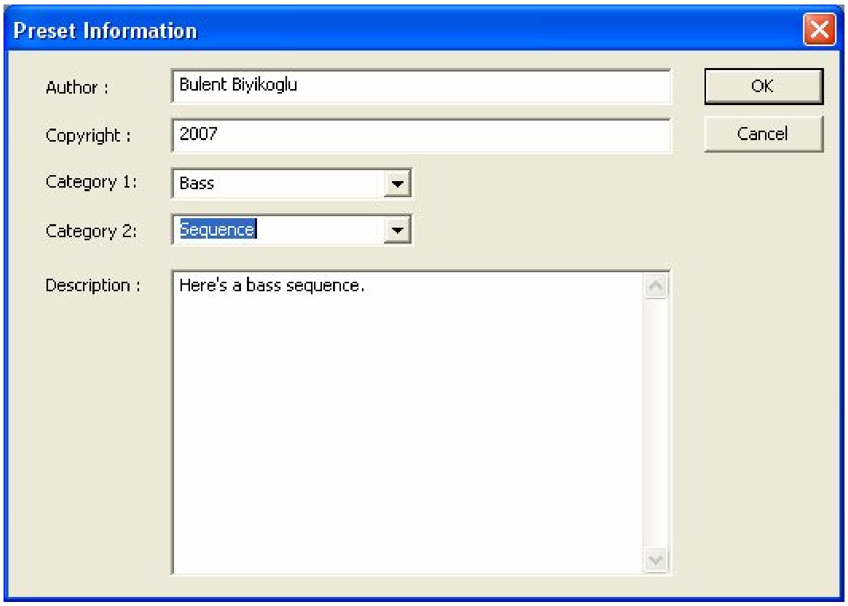


3.8 Editing Preset Information

In each "complete" preset, author, copyright, preset description, and two preset categories are stored. Those details are edited on the *Preset Information* screen, which is accessible by:

- Clicking on the 

button, or
- (<u>Standalone version only</u>) Choosing "*Preset Information*" sub-menu from the "*Edit*" menu.

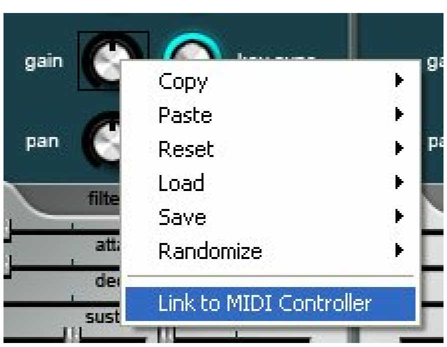


When you're done with entering the preset information details, you an click on the *OK* button to save your changes.

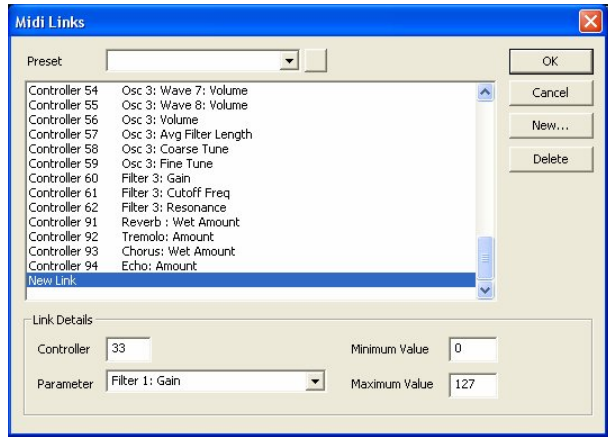
3.9 Linking Continuous Parameters to MIDI Controllers

SynthMaster allows you to link multiple continuous parameters (that are represented as sliders and knobs on the user interface) to a single MIDI Controller. With *SynthMaster*, linking a parameter to a MIDI controller is very easy since *SynthMaster* has intelligent *MIDI Learn* feature. All you have to do is:

- Send a control change (CC) message from your MIDI controller by moving a slider or turning a knob.
- Bring your mouse over a slider or a knob on the *SynthMaster* user interface, and right click the mouse button. The "Link to MIDI Controller" menu item will be active. Select this menu item:



- The *MIDI Links* screen will appear on the screen. Click on the *New* button on this screen to create a new link. You'll notice that the controller number and the parameter name values are populated automatically:



- The last step in creating the link is to enter the minimum and maximum values for our linked parameter:
    - *Minimum value* is the value of the linked parameter when the MIDI controller sends 0 for the controller value.
    - *Maximum value* is the value of the linked parameter when the MIDI controller sends 127 for the controller value.

When you save a "complete" preset, all midi links settings will be saved as well.

3.9.1 Removing an Existing Midi Link

To remove an existing link, first click on the link on the list to select the link, and then click on the Delete button to remove the link.

3.9.2 Midi Links Presets

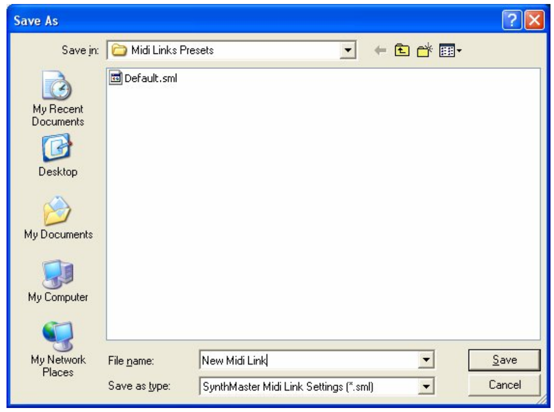
SynthMaster supports saving midi links settings as presets, so that you can quickly access different midi links settings. Midi links presets are stored under the following location: *<Common Files Folder>\KV331 Audio\SynthMaster\Midi Links Presets*

3.9.2.1 Loading Midi Links Presets

To see the existing midi links presets, click on the dropdown list, and then choose the preset that you want to load:

3.9.2.2 Saving Midi Links Presets

To save the current midi links settings as a new preset, click on the  button. This will bring up the "Save As" file browser window. Enter a new file name here and hit *Save* :

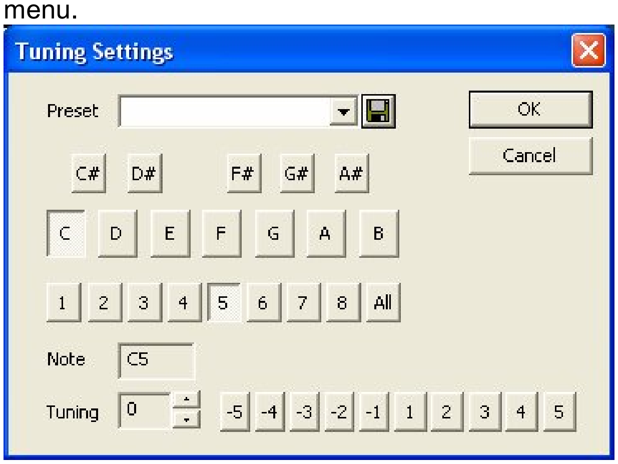


Important Note: Since midi links presets are saved to a fixed location, do not change the directory here or create a sub-directory.

3.10 Tuning Settings

SynthMaster supports custom tuning at the preset level, so each "complete" preset can have its own tuning settings. The tuning settings can be edited on the *Tuning Settings* screen, which is accessible by:

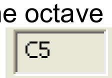
- Clicking on the  button on the SynthMaster user interface, or
- (Standalone version only) Choosing the *Tuning Settings* sub-menu item from the *Edit* menu.



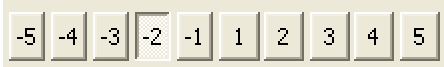
On this screen, you can

- Tune each note separately
- Tune notes at octave intervals
- Load a tuning preset
- Save the current tune settings as a preset

3.10.1 Tuning Each Note Separately

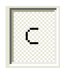
- Click on the note name  to select the note
- Click on the octave number  to select the octave. The selected note will be displayed: 

- Click on the up-down arrows  to set the tuning for the note. The tuning can be set between +/- 50 cents.

- To set the tuning, you can alternatively click on one of the buttons:


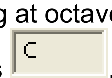
These buttons will set the tuning in multiples of 11 cents. In Turkish classical music, 1 whole note interval (200 cents) is divided into 9 equal intervals called "*coma*", which makes 22 cents approximately.

3.10.2 Tuning Notes at Octave Intervals

- Click on the note name 

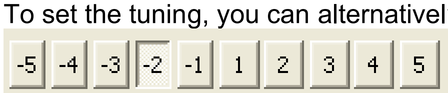
to select the note

- Click on the 

button to apply tuning at octave intervals. The selected note will be displayed without any octave numbers 

, indicating that the tuning will be applied to notes at all octaves.

- Click on the up-down arrows  to set the tuning for the note. The tuning can be set between +/- 50 cents, and it will apply to notes at all octaves. For instance, if the selected note is "C", the tuning will be applied to C0, C1, C2…C7.

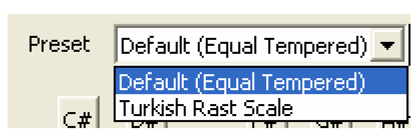
- To set the tuning, you can alternatively click on one of the buttons, as described above:


3.10.3 Tuning Presets

SynthMaster supports saving tuning settings as presets, so that you can quickly access different tuning settings. Tuning presets are stored under the following location: *<Common Files Folder>\KV331 Audio\SynthMaster\Tuning Presets*

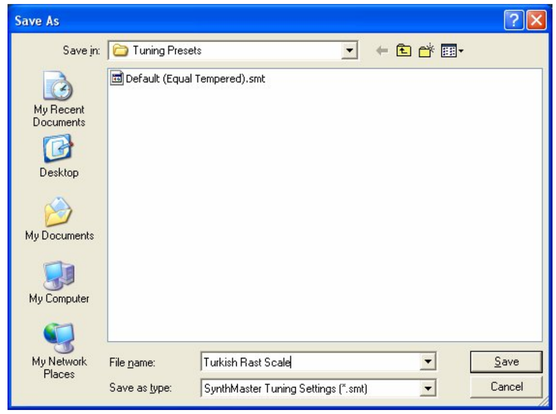
3.10.3.1 Loading Tuning Presets

To see the existing tuning presets, click on the dropdown list, and then choose the preset that you want to load:



3.10.3.2 Saving Tuning Presets

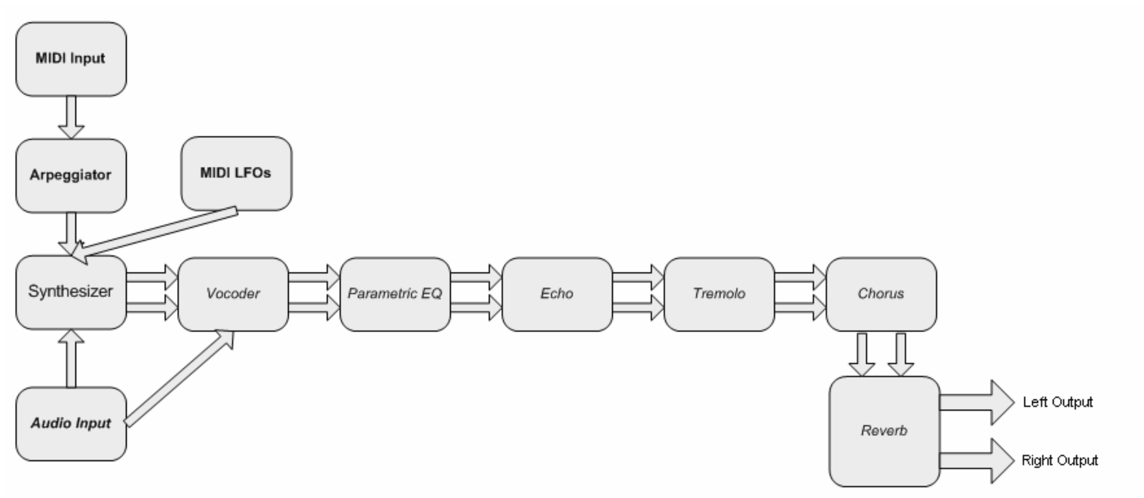
To save the current tuning settings as a new preset, click on the  button. This will bring up the “Save As” file browser window. Enter a new file name here and hit *Save* :



Important Note: Since tuning presets are saved to a fixed location, do not change the directory here or create a sub-directory.

4 SynthMaster Architecture

SynthMaster has a unique audio engine that makes it possible to use *SynthMaster* both as a software synthesizer and also as an effect plug-in:



As you see on this block diagram, *SynthMaster* has many processing blocks in its audio engine:

- The incoming MIDI input is first processed by the *Arpeggiator*.
- *The MIDI LFOs* generate CC (control change) messages that are used for modulating instrument parameters.
- The Synthesizer synthesizes the instrument sound by using the outputs of the Arpeggiator and the *MIDI LFOs*, and also the audio input.
- The stereo output of the Synthesizer goes through the effect chain in the following order:
    - Vocoder
    - Parametric EQ
    - Echo
    - Tremolo
    - Chorus
    - Reverb

4.1 Synthesizer

The synthesizer in *SynthMaster* has the following features:

- It is mono-timbral, i.e. it only accepts 1 channel MIDI input.
- It has variable polyphony between 8-64 voices, and it can work as a monophonic synthesizer as well.
- It has up to 4 "*Unison*" voices. In "*Unison*" mode, polyphony is not reduced as in other synthesizers.

4.1.1 Master Controls

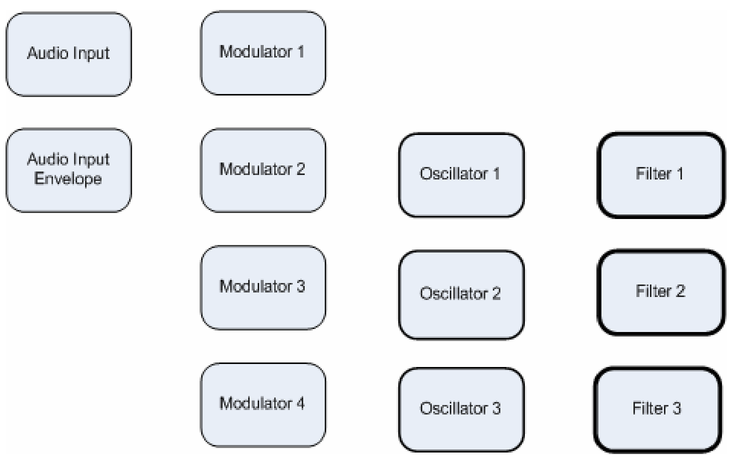
	***Master Volume*** controls the volume of the synthesizer output. Its value is between 0.0 - 1.0, increasing linearly.
	***Master tune*** is used to fine tune all the oscillators and modulators that are operating at audio rate. Its value is between –64, + 64 cents. 0 corresponds to 440 Hz A tuning.
 	By clicking the ***unison*** button, you turn on the unison mode which increases the number of voices that are playing, and that creates a chorus / phaser type effect. Since it wouldn't make much sense to play multiple voices that have identical parameters, we add a slight tune difference between them using the ***tune spread*** parameter. ***Tune spread*** is between 0.0 – 8.0 cents, and it also adds some random fine tune (between -/+4 cents) to each oscillator when they receive note on messages. The number of unison voices can be changed by clicking the right mouse button and then selecting the *"Unison"* sub-menu item on the popup menu: 
  	In ***mono*** mode, only one note can be played at a time. So, for instance if you are playing C1 and then press C2, the pitch suddenly changes from C1 to C2. The time of this pitch change is determined by ***portamento*** time. Portamento time is between 5 milliseconds – 20 seconds, increasing exponentially. In mono mode the lowest pressed note has the highest priority, so if you press C1 E1 G1, and then release G1, C1 is the note that continues to play, not E1. ***Legato*** mode is only effective in ***mono*** mode. If legato is on, none of the envelopes are reset to their initial levels when transitioning from one note to another, so there is continuity in envelopes levels.
	Clicking on the ***key input*** button turns SynthMaster into a FX Processor : The incoming audio signal is mixed with the synthesizer output, and the mix goes through the FX chain.
	Clicking on the ***tuning*** button brings up the *Tuning Settings* screen, where you can micro tune each individual note.
	Clicking on the ***preset info*** button brings up the *Preset Info* screen, where you can enter information about how the preset is constructed.
	Clicking on ***controllers*** button brings up the *Midi Links* screen, where you can create links between MIDI controllers and preset parameters.

4.1.2 Setting Polyphony

To set the maximum number of voices the synthesizer can play at a time, click on the right mouse button, and then select the "Polyphony" sub-menu item on the popup menu:

4.1.3 Voices

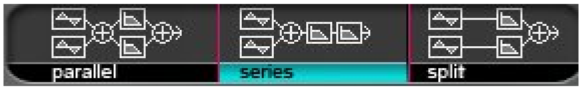
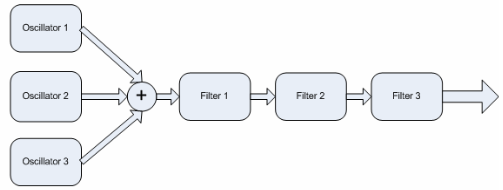
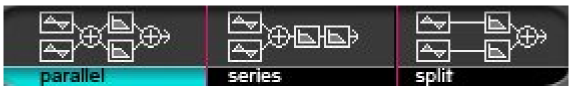
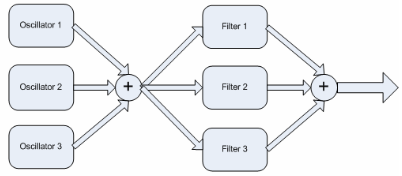
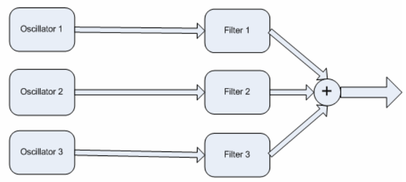
Each voice in *SynthMaster* has the following elements:



Each of the 4 modulators (or the audio input) can modulate:

- The phase of other modulators
- The phase of the same modulator (using feedback)
- The frequency (pitch) of other modulators
- The volume (amplitude) of any oscillator
- The phase of any oscillator
- The frequency (pitch) of any oscillator
- The cutoff frequency of any filter
- The resonance of any filter
- The pan of any filter

The filters can be connected to the oscillators in 3 different ways:

<table>
<tr><td>Series
</td><td></td></tr>
<tr><td>Parallel
</td><td></td></tr>
<tr><td>Split
</td><td></td></tr>
</table>

4.1.3.1 Oscillators

SynthMaster has 3 oscillators that are accessible from the *Osc 1*, *Osc 2*, and *Osc 3* tabs on the user interface. Each oscillator's parameters are categorized under 5 sections:

- General
- Waveform
- EQ
- Volume Envelope
- Frequency Envelope

4.1.3.1.1 General

This section contains the main parameters for the oscillator:

<table>
<tr><td></td><td>Bypass</td><td>When Bypass is on, the oscillator is inactive and does not consume any CPU cycles.</td></tr>
<tr><td></td><td>Volume</td><td>Volume controls the level of the oscillator waveform. Its value is between 0.0 and 1.0, increasing linearly.</td></tr>
<tr><td></td><td>Avg Filter Length</td><td>With the Avg Filter Length parameter, the waveform of the oscillator is filtered with a comb (Moving average) filter. The filter's length is between 0 and 127. When it is 0, no filtering is applied.

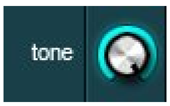
Moving average filters can be used to emulate plucked string sounds, such as electric/classical guitar. The following tutorial preset demonstrates this:

Chapter 1\016-Osc Avg Filter Length-Saw Osc.tfx</td></tr>
</table>

	**Coarse Tune**	**Coarse tune** and **fine tune** parameters are used to tune the oscillator. The overall tune of the oscillator at a specific note is calculated as: **Master tune** + **Tune Spread** + **Coarse tune** + **Fine tune** + Tune of the note (which can be set on the Tuning Settings screen) **Coarse tune** is between -64/+63 semitones. When the **Osc Mode** is set to  , **Coarse tune** is used to set the oscillator's fixed frequency between C0 – G10. **Fine tune** is between -64/+63 cents.
	**Fine Tune**	
	**Phase Velocity Sens.**	**Phase velocity sensitivity** is used to alter the phases of the 8 waveforms (that form the oscillator waveform) based on the MIDI velocity of the note the oscillator is playing. **Phase velocity sensitivity** is between -64/+63. For waveforms that have a harmonic number of zero (i.e. their value is constant), **phase velocity sensitivity** is used to alter the volumes of those waveforms using the MIDI velocity. This is especially useful when distortion is applied to the oscillator's output. The following tutorial preset demonstrates this: *Chapter 1\021-Distortion-Tanh-Sine Osc Plus DC.tfx* For noise waveforms, phase velocity sensitivity is again used to alter the color (tone) of the noise using the MIDI velocity. The following tutorial preset demonstrates this: *Chapter 1\005-Basic Waveforms-White Noise.tfx*
	**Osc Mode**	Osc Mode determines how the oscillator's frequency is set: • *Key Sync*: At this setting, the frequency is same as the frequency of the MIDI note the oscillator is playing. • *Fixed Freq*: At this setting, the oscillator has a fixed frequency, which can be set by changing the value of the **Coarse Tune** parameter. • *Osc Sync*: This setting only applies to Osc 2 or Osc 3. At this setting, the oscillator's frequency is synchronized to the frequency of Osc 1. Whenever Osc 1's instantaneous phase becomes zero, the oscillator's phase is reset to zero as well. This is known as "Hard Sync", and is widely used in analog synthesizers. For a demonstration of "Hard Sync", take a look at the following tutorial preset: *Chapter 1\036-Frequency Env-Osc Sync-2 Saws.tfx*

4.1.3.1.2 Waveform

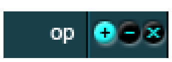
The waveform of each oscillator is constructed using 8 different waves. Each of these 8 waves has its own volume, phase, tone, harmonic number (frequency), shape and operator parameters.

<table>
<tr><td></td><td>Volume</td><td>Volume is between 0.0 – 1.0, increasing linearly.</td></tr>
<tr><td></td><td>Phase</td><td>Phase is between 0 – 359 degrees, increasing linearly.</td></tr>
<tr><td></td><td>Tone</td><td>By changing the tone of a wave, you are basically changing its timbre. The higher the tone, the brighter the wave will sound. Tone can be between 0-127, and each single increment corresponds to a 1 semitone increment in the lowpass filter cutoff that determines the tone of the wave. 127 basically means the cutoff will be at 1/2 sampling rate, 116 will be 1/4 the sampling rate, and so on...</td></tr>
<tr><td></td><td>Harmonic Number</td><td>Harmonic number is the frequency of the wave. Its value can be between 0-32.

When the harmonic number is zero, the wave is essentially a constant value, and its volume can be made sensitive to the MIDI velocity using the Phase Velocity Sensitivity parameter.</td></tr>
<tr><td></td><td>Wave Shape</td><td>Each wave's shape can be one of the 5 basic waveforms:
<table>
<tr><td></td><td>Sine</td><td>A sine wave contains only the 1st harmonic, so it is the most basic waveform that can be used in sound synthesis.</td></tr>
<tr><td></td><td>Triangle</td><td>A triangle wave contains only odd-harmonics, just like a square wave. However the amplitudes of odd harmonics are not as high as they are in a square wave.</td></tr>
<tr><td></td><td>Square</td><td>A square wave contains only odd-harmonics.</td></tr>
<tr><td></td><td>Sawtooth</td><td>A sawtooth wave contains all harmonics, so it is widely used in sound synthesis using subtractive synthesis.</td></tr>
<tr><td></td><td>White Noise</td><td>White noise contains all frequencies, so it is aperiodic. Only the first wave of an oscillator can have its shape set to white noise.

The noise waveform is passed through a lowpass filter, whose cutoff frequency is controlled by the tone parameter. The tone can be made sensitive to the MIDI velocity using the Phase Velocity Sensitivity parameter.</td></tr>
</table></td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td>
You might think that Pulse waves are omitted here as they are also one of the basic wave types. However, it is indeed possible to create Pulse waves with SynthMaster:

As seen above, a Pulse wave can be constructed by subtracting a sawtooth wave from another one that is at the same volume and frequency, but at a different phase (Phase of wave 2 controls the pulse width). Since the SynthMaster synthesis engine uses bandlimited waves, the generated pulse will be also bandlimited as well.
</td>
</tr>
<tr>
<td>
</td>
<td>Operator</td>
<td>
To better explain what the operator does, let’s write the equation of how the oscillator waveform is constructed:

Osc Waveform = (Wave 1 * Volume 1)
Operator 2 (Wave 2 * Volume 2)
…
Operator 8 (Wave 8 * Volume 8)

The following tutorial preset demonstrates how operators can be used to create more complex waveforms:

Chapter1\010-Additive Synthesis-Operators.tfx
</td>
</tr>
</table>

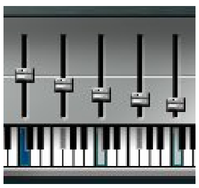
4.1.3.1.3 EQ

The EQ section is used for amplify/attenuate the volumes of oscillator waves, just like an equalizer. When the oscillator waveform is constructed using the 8 waves, each wave’s volume is multiplied by the level of the corresponding EQ band. There are 20 EQ bands, which are spaced equally at ½ octaves. Their values are between 0 – 2.

For instance, let's assume you have the following waveform settings for Osc 1:

	Harmonic #	Volume
Wave 1	1	1.0
Wave 2	2	1.0
Wave 3	3	
Wave 4	4	1.0
Wave 5	5	
Wave 6	6	
Wave 7	7	
Wave 8	8	

And the EQ settings are as follows:



Band #	Note	Level
10	*C5*	*64 (1.0)*
11	*F#5*	*48*
12	*C6*	*32 (0.5)*
13	*F#6*	*24*
14	*C7*	*16 (0.25)*

When the oscillator is playing C5, wave 1's volume will be multipled by EQ Band 10, wave 2's volume will be multiplied by EQ Band 12, and wave 4's volume will be multiplied by EQ Band 14.

After the oscillator waveform is constructed, you can distort the waveform using any of the 18 waveshaper functions:

Hard Limiter

Half Wave Rectifier

Half Wave Rectifier + Limiter

Full Wave Rectifier

Full Wave Rectifier + Limiter

Square

Square + Limiter

X * ABS(X)

X * ABS(X) + Limiter

Square Root

Sine

4 Step Quantizer

6 Step Quantizer

8 Step Quantizer

10 Step Quantizer

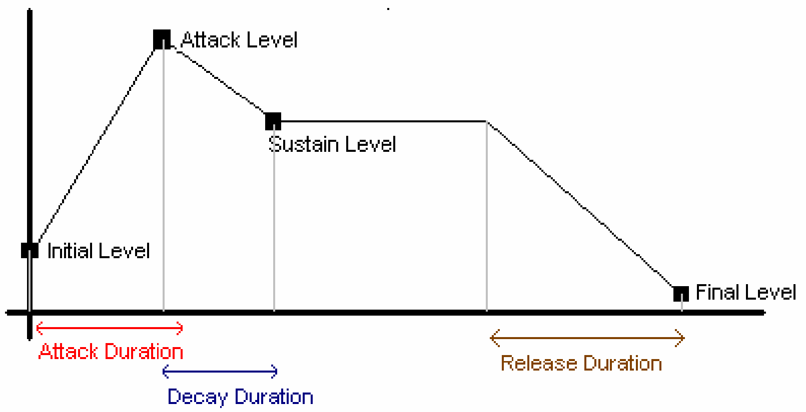
TanH Soft Clipper

ArcTan Soft Clipper

4.1.3.1.4 Volume Envelope

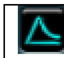
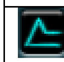
Each oscillator in *SynthMaster* has its own volume envelope. With volume envelope, the volume of the oscillator is changed over time.

The following graph illustrates the common parameters of an envelope:



	**Attack Duration**	This is the time it takes for the envelope to rise from the initial level to the attack level. When **tempo sync** is off, the **attack duration** is between 6 milliseconds – 20 seconds, increasing exponentially. When **tempo sync** is on, the **attack duration** is synced to the tempo, and the maximum attack duration is a whole note (= 4 quarters)
	**Initial Level**	This is the level at which the envelope starts when the oscillator starts playing. **Initial level** is between 0 and 1.0, increasing linearly.
	**Decay Duration**	This is the time it takes for the envelope to fall down to the **sustain level**, starting at the **attack level**. When **tempo sync** is off, the **decay duration** is between 6 milliseconds – 20 seconds, increasing exponentially. When **tempo sync** is on, the **decay duration** is synced to the tempo, and the maximum attack duration is a whole note (= 4 quarters)
	**Attack Level**	The envelope rises from the **initial level** to the **attack level**. **Attack level** is between 0 and 1.0, increasing linearly.
	**Sustain Level**	After the envelope reaches the **attack level**, it starts falling down to the **sustain level**, and stays at this level until it receives a "*Note Off*" message. **Sustain level** is between 0 and 1.0, increasing linearly.

	**Final Level**	When the envelope receives a “Note Off” message, it starts falling down to the **final level**. **Final level** is between 0 and 1.0, increasing linearly.
	**Release Duration**	This is the time it takes for the envelope to fall down to the **final level**, starting at the **sustain level**. When **tempo sync** is off, the **sustain duration** is between 6 milliseconds – 20 seconds, increasing exponentially. When **tempo sync** is on, the **sustain duration** is synced to the tempo, and the maximum attack duration is a whole note (= 4 quarters)
	**Key Sensitivity**	**Key sensitivity** is used to change the envelope durations based on the MIDI note number. When it is set to the max level (=127), envelope durations will double at every two octaves. So for example, if the envelope duration at C5 is 1 seconds, it will be: 0.500 seconds for C7 0.707 seconds for C6 1.414 seconds for C4 2.000 seconds for C3 2.823 seconds for C2 4.000 seconds for C1 The following tutorial preset demonstrates **key sensitivity**: *Chapter 1\ 027-Volume Env-Key Sensitivity.tfx*
	**Env. Amt. Vel. Sens.**	**Envelope amount velocity sensitivity** is used to alter the initial, attack, sustain and final levels based on the MIDI velocity of the note the oscillator is playing. **Envelope amount velocity sensitivity** is between -64/+63. When it is set to a positive value, the envelope levels increase as the MIDI velocity increases. Similarly, when it is set to a negative value, the levels decrease as the MIDI velocity increases. The following tutorial preset demonstrates **envelope amount velocity sensitivity**: *Chapter 1\ 028-Volume Env-Env Amount Vel Sens.tfx*
	**Env. Dur. Vel. Sens.**	**Envelope duration velocity sensitivity** is used to alter the attack, decay and release durations based on the MIDI velocity of the note the oscillator is playing. **Envelope duration velocity sensitivity** is between -64/+63. When it is set to a positive value, the envelope durations increase as the MIDI velocity increases. Similarly, when it is set to a negative value, the durations decrease as the MIDI velocity decreases. The following tutorial preset demonstrates **envelope duration velocity sensitivity**: *Chapter 1\ 028-Volume Env-Env Amount Vel Sens.tfx*

<table>
<tr>
<td>
</td>
<td>Slope Type</td>
<td>The slope type determines the slope of the attack, decay and release sections of the envelope. An envelope can have 3 types of slope:
<table>
<tr><td></td><td>Linear Exponential</td><td>The attack section has a linear slope, decay and release sections have exponential slopes.</td></tr>
<tr><td></td><td>Linear</td><td>All envelope sections have linear slope</td></tr>
<tr><td></td><td>Exponential</td><td>All envelope sections have exponential slope.</td></tr>
</table></td>
</tr>
<tr>
<td>
</td>
<td>Tempo Sync</td>
<td>When tempo sync is off, all envelope durations are between 6 milliseconds – 20 seconds, increasing exponentially.

When tempo sync is on, all envelope durations are synced to the tempo, and the maximum duration for an envelope section (attack, decay, release) is a whole note (= 4 quarters)

The following preset demonstrates a tempo synced volume envelope: Chapter 1\025-Volume Env-Tempo Sync.tfx</td>
</tr>
<tr>
<td>
</td>
<td>Modulation Source</td>
<td>The volume of the oscillator can be modulated by the following modulation sources:
Modulator 1
Modulator 2
Modulator 3
Modulator 4
Audio Input
Audio Input Envelope
Two types of modulation can be achieved based on the DC bias (constant value added to the waveform) of the modulator:
Amplitude Modulation (If the modulator has DC Bias).
Ring Modulation (if the modulator has no DC bias)
The following two tutorial presets demonstrate the different between amplitude modulation and ring modulation:
Chapter 3\002-Modulation Types-Amplitude Modulation.tfx
Chapter 3\001-Modulation Types-Ring Modulation.tfx</td>
</tr>
</table>

4.1.3.1.5 Frequency Envelope

Each oscillator in *SynthMaster* has its own frequency (pitch) envelope. With frequency (pitch) envelope, the frequency (pitch) of the oscillator is changed over time.

By using fast attack/decay times for frequency envelopes, you can create percussive sounds. The following presets demonstrate this:

- *Chapter 1\ 034-Frequency Env-Percussive Sine.tfx*
- *Chapter 1\ 035-Frequency Env-Percussive Sines.tfx*

By synchronizing the frequency of Osc 2/3 to Osc 1's frequency and using frequency envelopes you can create "*Hard Sync*" type sounds. The following presets demonstrate this:

- *Chapter 1\ 036-Frequency Env-Osc Sync-2 Saws.tfx*
- *Chapter 1\ 037-Frequency Env-Osc Sync-2Squares and Env Amt VS.tfx*

On the frequency envelope section of the oscillator, you can also specify modulation options:

<table>
<tr>
<td>
</td>
<td>Modulation Type</td>
<td>There are two types of modulations:
<table>
<tr><td>
</td><td>Frequency Modulation (FM)</td><td>When this type of modulation is selected, the frequency of the oscillator is modulated by the modulation source.</td></tr>
<tr><td>
</td><td>Phase Modulation (PM)</td><td>When this type of modulation is selected, the phase of the oscillator is modulated by the modulation source.

Phase modulation has been first used in sound synthesis by Yamaha in their DX7 synthesizer.

With SynthMaster, you can implement most of the DX7 algorithms. Check out the tutorial presets on modulation (Chapter 3), most of the DX7 algorithms are demonstrated there.</td></tr>
</table>
</td>
</tr>
<tr>
<td>
</td>
<td>Modulation Source</td>
<td>The volume of the oscillator can be modulated by the following modulation sources:

Modulator 1
Modulator 2
Modulator 3
Modulator 4
Audio Input
Audio Input Envelope

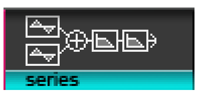
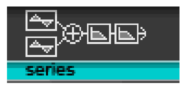
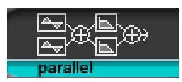
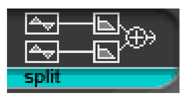
</td>
</tr>
</table>

4.1.3.2 Filters

SynthMaster has 3 filters that are accessible from the *Filters* tab on the user interface. Each filter's parameters are categorized under 3 sections:

- General
- Filter Envelope
- Modulation Sources

4.1.3.2.1 General

<table>
<tr><td></td><td>Bypass</td><td>When Bypass is on, the filter is inactive and does not consume any CPU cycles</td></tr>
<tr><td></td><td>Connection Type</td><td>The filters can be connected in 3 different ways:
<table><tr><td></td><td>Series</td><td>The filters are connected in series. With this type of connection, only the last filter in the chain has stereo output.</td></tr><tr><td></td><td>Parallel</td><td>The filters are connected in parallel.</td></tr><tr><td></td><td>Split</td><td>Each filter is connected to the corresponding oscillator.</td></tr></table>The following tutorial presets demonstrate how different types of connection types create different results on the same oscillators:
• Chapter 3\005-Filter Connections-Series.tfx
• Chapter 3\004-Filter Connections-Parallel.tfx
• Chapter 3\006-Filter Connections-Split.tfx</td></tr>
<tr><td></td><td>Cutoff Frequency</td><td rowspan="3">“Cutoff” frequency is the frequency at which the filter starts to attenuate frequencies above/below the cutoff, based on the filter type.

The instantaneous filter cutoff frequency can be written as:

Instantaneous Cutoff = Key Sync*MIDI Note Frequency
+ Env Amt*Filter Envelope
+ Cutoff Frequency

As you see above, the instantaneous cutoff value is dependant on 3 parameters: Cutoff Frequency, Envelope Amount and Key Sync.

Cutoff frequency is the lowest frequency the instantaneous cutoff can go down to. Its value is between 12.4 Hz – 18.9 kHz, increasing exponentially.

Envelope amount is the level of the filter envelope. Its value is between 0 – 1.0, increasing linearly.

Key Sync is used to change the instantaneous cutoff frequency based on the current MIDI note playing, as seen from the above equation. Its value is between 0 – 1.0, increasing linearly.</td></tr>
<tr><td></td><td>Envelope Amount</td></tr>
<tr><td></td><td>Key Sync</td></tr>
</table>

<table>
<tr>
<td>
</td>
<td>Gain</td>
<td> Gain is used to cut or boost the output volume of the filter. Boosting the output volume of a filter is especially useful when distortion (such as hard limiter) is applied after the filter.
Gain is between 0.25 (-12 dB) – 4.00 (+12 dB), increasing exponentially. At its default setting, its value is 1.00 (0 dB) </td>
</tr>
<tr>
<td>
</td>
<td>Pan</td>
<td rowspan="3"> The instantaneous pan level of a filter is calculated as:
Pan + Pan Velocity Sensitivity + Pan Key Sensitivity
Pan is a value between -0.5 and +0.5, which correspond to hard left and hard right panning respectively. The default value of pan is zero, which corresponds to center panning.
Pan Velocity Sensitivity adds an offset value to the pan level based on the velocity of the current playing MIDI note. The offset velocity level for Pan Velocity Sensitivity is 64. So, when the velocity is 64, Pan Velocity Sensitivity does not have any effect on the instantaneous pan level. For velocities higher than 64, the sound will:
come from the right of "Pan" if "Pan VS" is positive
come from the left of "Pan" if "Pan VS" is negative.
Similarly, for velocities lower than 64, the sound will:
come from the left of "Pan" if "Pan VS" is positive
come from the right of "Pan" if "Pan VS" is negative.
Pan Key Sensitivity, on the other hand, adds an offset value to the pan level based on the note number of the current playing MIDI note. The offset note for Pan Key Sensitivity is note 64 (E5), So, when the note number is E5, Pan Key Sensitivity does not have any effect on the instantaneous pan level. For notes higher than E5, the sound will:
come from the right of "Pan" if "Pan Key Sens" is positive
come from the left of "Pan" if "Pan Key Sens" is negative.
Similarly, for notes lower than E5, the sound will:
come from the left of "Pan" if "Pan Key Sens" is positive
come from the right of "Pan" if "Pan Key Sens" is negative.
<u>One important thing about Pan Key Sensitivity</u>: This parameter is not visible on the user interface, but you can use an <u>easy parameter</u> to control its value. By default easy parameter 8 is assigned to "Fiter 1 Pan Key Sens", so you can use that easy parameter to change the value of Filter 1's Pan Key Sensitivity. </td>
</tr>
<tr>
<td>
</td>
<td>Pan Vel. Sens.</td>
</tr>
<tr>
<td>Hidden</td>
<td>Pan Key Sens.</td>
</tr>
<tr>
<td>
</td>
<td>Resonance</td>
<td> Resonance controls the gain of the filter around the instantaneous filter cutoff frequency. High values of resonance correspond to high gains.
Resonance is between 0.6 – 100, increasing exponentially. </td>
</tr>
</table>

Filter Type

SynthMaster has 23 types of filters that you can use:

1st order Lowpass

Lowpass filters attenuate frequencies above the cutoff frequency. For 1st order (1 pole) filters, the slope of attenuation is 6 dB per octave.

1st order Highpass

Highpass filters attenuate frequencies below the cutoff frequency. For 1st order (1 pole) filters, the slope of attenuation is 6 dB/octave.

1st order Allpass

Allpass filters pass all the frequencies, but create phase distortion. They are mostly used to create phasing effects.

1st order LowShelve

A low shelve filter is similar to a lowpass filter. The equation for a low shelve filter can be written as:

1st order Low shelve = Input + (**Gain**-1.0**)** * 1st order Lowpass(input)

As seen on this equation, when gain is set to 1.0, low shelve filter passes the input signal as it is. However, when gain is greater than zero, the lowpass filtered signal is added to the input. Similarly, when gain is smaller than 1.0, the lowpass filtered signal is subtracted from the input.

Since the order of the lowpass filter is 1, the slope of attenuation for frequencies above the cutoff is 6 dB per octave.

1st order HighShelve

A high shelve filter is similar to a highpass filter. The equation for a high shelve filter can be written as:

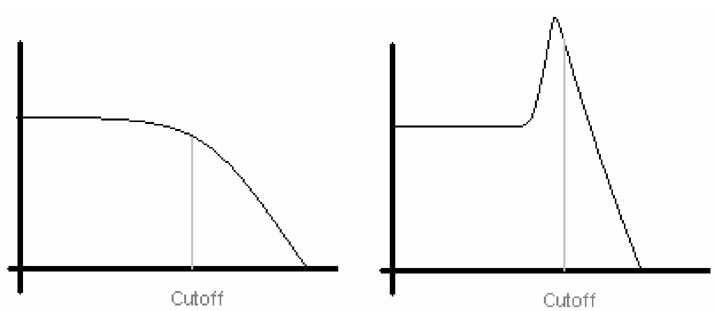
1st order High shelve = Input + (**Gain**-1.0**)** * 1st order Highpass(input)

As seen on this equation, when gain is set to 1.0, high shelve filter passes the input signal as it is. However, when gain is greater than zero, the highpass filtered signal is added to the input. Similarly, when gain is smaller than 1.0, the highpass filtered signal is subtracted from the input.

Since the order of the highpass filter is 1, the slope of attenuation for frequencies above the cutoff is 6 dB per octave.

2nd order Lowpass

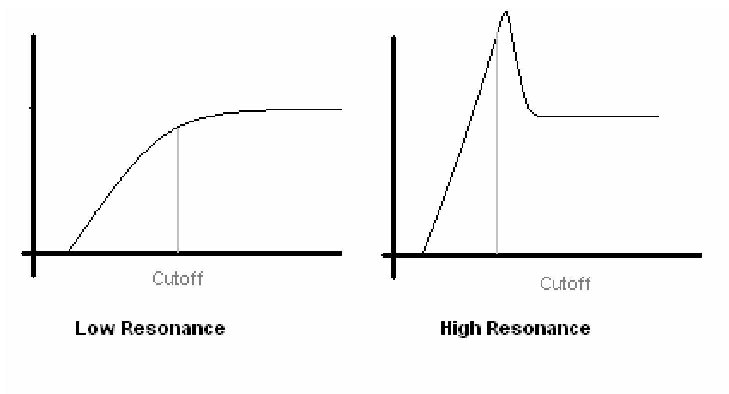
2nd order lowpass filters have 12 dB per octave attenuation slope above the cutoff frequency. The **resonance** parameter controls the filter gain around the cutoff frequency:



Low Resonance **High Resonance**

2nd order Highpass

2nd order highpass filters have 12 dB per octave attenuation slope below the cutoff frequency. The **resonance** parameter controls the filter gain around the cutoff frequency:



2nd order Allpass

Just like 1st order allpass filters, 2nd order allpass filters pass all the frequencies as well, but they create phase distortion.

2nd order LowShelve

The equation for a 2nd order low shelve filter can be written as:

2nd order Low shelve = Input + (**Gain**-1.0**)** * 2nd order Lowpass(input)

Since the filter order is 2, the attenuation above the cutoff frequency is 12 dB per octave, and the **resonance** parameter controls the filter gain around the cutoff frequency.

2nd order HighShelve

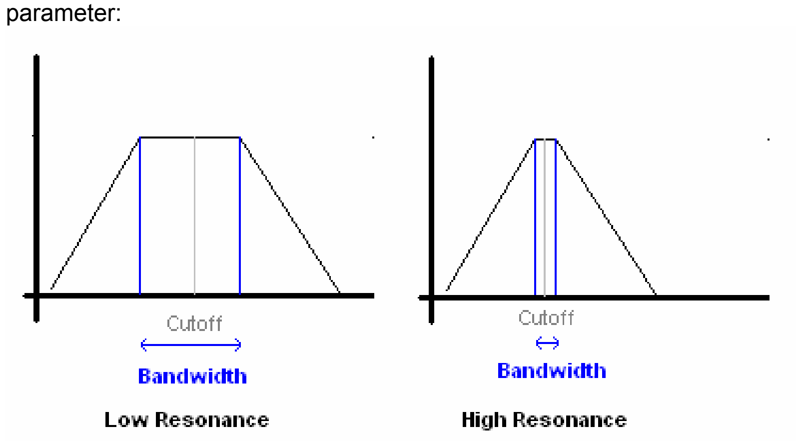
The equation for a 2nd order high shelve filter can be written as:

2nd order High shelve = Input + (**Gain**-1.0**)** * 2nd order Highpass(input)

Since the filter order is 2, the attenuation below the cutoff frequency is 12 dB per octave, and the **resonance** parameter controls the filter gain around the cutoff frequency.

2nd order BandPass

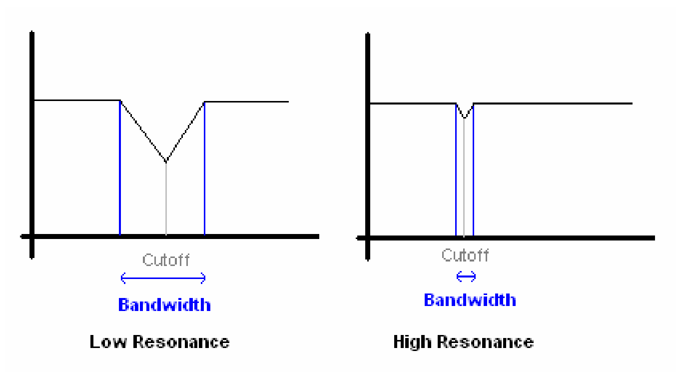
2nd order bandpass filters pass frequencies around the cutoff frequency, and attenuate others. The difference between the highest and lowest frequency in the pass band is called *bandwidth*. *Bandwidth* is controlled using the **resonance** parameter:



For 2nd order bandpass filters, the slope of attenuation outside the pass band is 6 dB per octave.

2nd order BandStop

2nd order bandstop filters attenuate frequencies around the cutoff frequency, and pass others. The difference between the highest and lowest frequency in the stop band is called *bandwidth*. *Bandwidth* is controlled using the **resonance** parameter:



For 2nd order bandstop filters, the slope of attenuation inside the stop band is 6 dB per octave.

2nd order Peaking

Peaking filters are a type of shelve filters. The equation for a 2nd order peaking filter can be written as:

2nd order Peaking = Input + (**Gain**-1.0**)** * 2nd order Bandpass(input)

4th order Lowpass

4th order lowpass filters are realized by connecting two 2nd order lowpass filters in series. The slope of attenuation above the cutoff frequency is 24 dB per octave.

4th order Highpass

4th order highpass filters are realized by connecting two 2nd order highpass filters in series. The slope of attenuation below the cutoff frequency is 24 dB per octave.

4th order BandPass

4th order bandpass filters are realized by connecting two 2nd order bandpass filters in series. The slope of attenuation outside the pass band is 12 dB per octave.

4th order BandStop

4th order bandstop filters are realized by connecting two 2nd order bandstop filters in series. The slope of attenuation inside the stop band is 12 dB per octave.

1st order Analog LP

1st order Analog HP

2nd order Analog LP

2nd order Analog HP

4th order Analog LP

4th order Analog HP

Just as in real analog filters, the “Analog” filters in SynthMaster have built-in saturation in their algorithms, so at high gain/resonance settings they will self-saturate and that will create distortion in their outputs.

The following tutorial presets demonstrate analog filters:

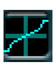
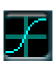
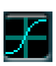
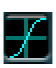
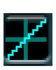
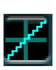
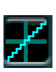
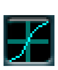
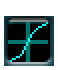
Chapter 3\012-Filter Types-Analog LP12.tfx

Chapter 3\013-Filter Types-Analog LP24.tfx

4th order analog lowpass / highpass filters are realized by connecting 2 2nd order analog lowpass / highpass filters in series, so they have more distortion compared to 2nd order analog filters.

Distortion Type

With SynthMaster, it is possible to connect a waveshaper to the output of the filter in series, so that the filter output can be distorted. The waveshaper can be any of the following 18 functions:

	Hard Limiter	The following tutorial preset demonstrates this type of distortion: *Chapter 3\014-Distortion Types-Hard Limiter.tfx*
	Half Wave Rectifier	The following tutorial preset demonstrates this type of distortion: *Chapter 3\015-Distortion Types-Half Rectifier+Limiter.tfx*
	Half Wave Rectifier + Limiter	
	Full Wave Rectifier	The following tutorial preset demonstrates this type of distortion: *Chapter 3\016-Distortion Types-Full Rectifier+Limiter.tfx*
	Full Wave Rectifier + Limiter	
	Input * Input	The following tutorial preset demonstrates this type of distortion: *Chapter 3\017-Distortion Types-Squarer.tfx*
	Input * Input + Limiter	The following tutorial preset demonstrates this type of distortion: *Chapter 3\018-Distortion Types-Squarer+Limiter.tfx*
	Input * ABS(Input)	The following tutorial preset demonstrates this type of distortion: *Chapter 3\019-Distortion Types-AbsXSquarer.tfx*
	Input * ABS(Input) + Limiter	The following tutorial preset demonstrates this type of distortion: *Chapter 3\020-Distortion Types-AbsXSquarer+Limiter.tfx*
	Square Root(Input)	
	Square Root(Input) + Limiter	
	Sin(Input)	The following tutorial preset demonstrates this type of distortion: *Chapter 3\021-Distortion Types-Sine.tfx*
	4 Step Quantizer	The following tutorial preset demonstrates this type of distortion: *Chapter 3\022-Distortion Types-4 Step Quantizer.tfx*
	8 Step Quantizer	The following tutorial preset demonstrates this type of distortion: *Chapter 3\022-Distortion Types-8 Step Quantizer.tfx*
	16 Step Quantizer	
	32 Step Quantizer	
	TanH Soft Clipper	The following tutorial preset demonstrates this type of distortion: *Chapter 3\024-Distortion Types-Tanh SoftClipper.tfx*
	ArcTan Soft Clipper	The following tutorial preset demonstrates this type of distortion: *Chapter 3\025-Distortion Types-ArcTan SoftClipper.tfx*

The following tutorial presets demonstrate how different types of filter connections change the sound of multiple filters that have distortion at their output:

- *Chapter 3\026-Distortion Types-2 LP12 Filters with SoftClip+Series.tfx*
- *Chapter 3\ 027-Distortion Types-2 LP12 Filters with SoftClip+Parallel.tfx*
- *Chapter 3\ 028-Distortion Types-2 LP12 Filters with SoftClip+Split.tfx*

4.1.3.2.2 Filter Envelope

The filter envelope provides an envelope signal for the filter so that its cutoff frequency can be changed over time. Filter envelope parameters are similar to Volume Envelope parameters.

4.1.3.2.3 Modulation Sources

The **cutoff frequency** of a filter can be modulated by:

Modulator 1
Modulator 2
Modulator 1 + Modulator 2
*Modulator 1 * Modulator 2*
Audio Input
Modulator 3
Modulator 4
Modulator 3 + Modulator 4
*Modulator 3 * Modulator 4*
Audio Input Envelope

The following tutorial presets demonstrate filter cutoff modulation:

- *Chapter 3\ 029-Modulation-Frequence-Synced Square LFOs.tfx*
- *Chapter 3\ 030-Modulation-Frequency-Synced Random SampleHold LFOs.tfx*
- *Chapter 3\ 031-Modulation-Frequency-Synced Pattern LFOs.tfx*
- *Chapter 3\ 032-Modulation-Frequency-Audio Rate Sine Mod.tfx*
- *Chapter 3\ 033-Modulation-Frequency-Audio Rate Sine Mod Speedup.tfx*

The **resonance** of a filter can be modulated by:

Modulator 1
Modulator 2
Modulator 1 + Modulator 2
*Modulator 1 * Modulator 2*
Audio Input Envelope
Modulator 3
Modulator 4
Modulator 3 + Modulator 4
*Modulator 3 * Modulator 4*

The following tutorial presets demonstrate resonance modulation:

- *Chapter 3\ 034-Modulation-Resonance-Synced Square LFOs.tfx*
- *Chapter 3\ 035-Modulation-Resonance-Synced Random SampleHold LFOs.tfx*
- *Chapter 3\ 036-Modulation-Resonance-Synced Pattern LFOs.tfx*

The **gain** of a filter can be modulated by:

Modulator 1
Modulator 2
Modulator 1 + Modulator 2
*Modulator 1 * Modulator 2*
Audio Input Envelope
Modulator 3
Modulator 4
Modulator 3 + Modulator 4
*Modulator 3 * Modulator 4*

The following tutorial presets demonstrate resonance modulation:

- *Chapter 3\ 037-Modulation-Gain-Synced Square LFOs.tfx*
- *Chapter 3\ 038-Modulation-Gain-Synced Random SampleHold LFOs.tfx*
- *Chapter 3\ 039-Modulation-Gain-Synced Pattern LFOs.tfx*

4.1.3.3 Modulators

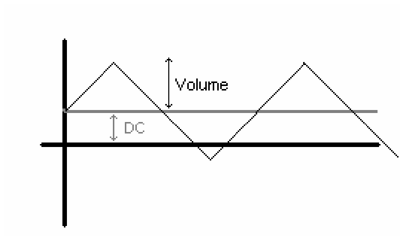
SynthMaster has 4 modulators that are accessible from the *Mods 1-2* and *Mods 3-4* tabs on the user interface. Each modulator's parameters are categorized under 3 sections:

- General
- Key Scaling
- Volume Envelope

4.1.3.3.1 General

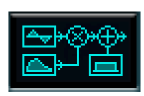
<table>
<tr><td>
</td><td>Volume</td><td rowspan="3">Volume controls the level of the oscillator waveform. Its value is between 0.0 and 1.0, increasing linearly.

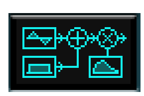
DC Level adds a constant offset to the modulator waveform:

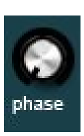


DC Level is between 0.0 and 1.0, increasing linearly.

DC Mode determines at what point the constant offset is added to the waveform:

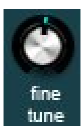
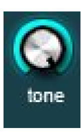
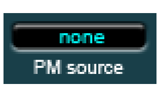
 (Envelope * Waveform) + Constant — At this setting, the constant value is added to the product of the volume envelope and the waveform.

 (Waveform + Constant) * Envelope — At this setting, the constant value is first added to the waveform, and then their sum is multiplied by the volume envelope.</td></tr>
<tr><td>
</td><td>DC Value</td></tr>
<tr><td>
</td><td>DC Mode</td></tr>
<tr><td>
</td><td>Poly Mode</td><td rowspan="2">When the modulator is running in poly mode, its phase is reset to the Phase value whenever the modulator receives a Note On message. In mono mode, the phase is not reset so the modulator keeps playing continuously.

Phase is between 0 – 359 degrees, increasing linearly.</td></tr>
<tr><td>
</td><td>Phase</td></tr>
</table>

<table>
<tr>
<td>
</td>
<td>Mode</td>
<td rowspan="3"> Mode determines how the modulator frequency is set:

Key Sync: At this setting, the modulator is running at audio rate, and its frequency is

MIDI note frequency * Frequency
Fixed Freq: At this setting, the modulator has a fixed frequency, which can be set by changing the value of the Coarse Tune parameter.
LFO: At this setting, the modulator runs as a low frequency oscillator, whose frequency can be set by changing the value of the Coarse Tune parameter. The LFO frequency is between 0.1 – 10 Hz, increasing exponentially.
Sync Beat: At this setting, the modulator again runs as a low frequency oscillator whose frequency is synced to tempo. Beat Multiplier dropdown control is used to set the synchronized frequency, which is between 2 whole notes and 1/64 note.
</td>
</tr>
<tr>
<td>
</td>
<td>Frequency</td>
</tr>
<tr>
<td></td>
<td>Beat Multiplier</td>
</tr>
<tr>
<td>
</td>
<td>Coarse Tune</td>
<td rowspan="2"> Coarse tune and fine tune parameters are used to tune the modulator when it's running in Key Sync mode. The overall tune of the modulator at a specific note is calculated as:
<u>Master tune</u> + <u>Tune Spread</u> + Coarse tune + Fine tune + Tune of the note (which can be set on the <u>Tuning Settings</u> screen)
Coarse tune is between -64/+63 semitones. When the Mode is set to 
, Coarse tune is used to set the oscillator's fixed frequency between C0 – G10.
Fine tune is between -64/+63 cents. </td>
</tr>
<tr>
<td>
</td>
<td>Fine Tune</td>
</tr>
<tr>
<td>
</td>
<td>Tone</td>
<td>By changing the tone of a modulator waveform, you are basically changing its timbre. The higher the tone, the brighter the waveform will sound. Tone can be between 0-127, and each single increment corresponds to a 1 semitone increment in the lowpass filter cutoff that determines the tone of the wave. 127 basically means the cutoff will be at ½ sampling rate, 116 will be 1/4 the sampling rate, and so on...</td>
</tr>
<tr>
<td>
</td>
<td>Modulation Source</td>
<td rowspan="3"> A modulator's phase or frequency can be modulator by other modulators, the audio input or its envelope.
Modulation source is used to select the modulation source.
Modulation type is use to select the modulation type:

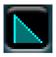
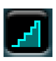
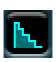
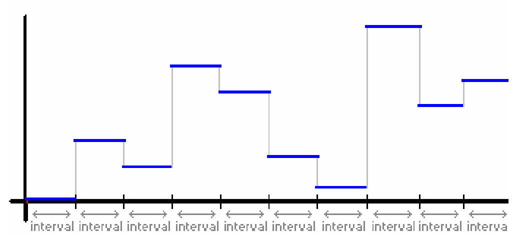

 Phase Modulation

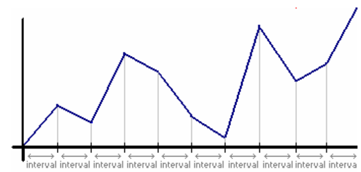
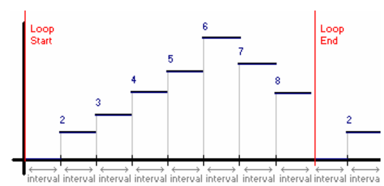
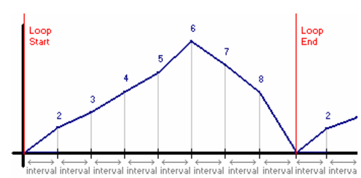
 Frequency Modulation

Modulation amount is used to set the modulation intensity. Its value is between 0 – 1.0, increasing linearly. </td>
</tr>
<tr>
<td>
</td>
<td>Modulation Type</td>
</tr>
<tr>
<td>
</td>
<td>Modulation Amount</td>
</tr>
</table>

Wave Type

15 types of wave shapes are available for the modulator waveform

	*Sine*	Can be used as an audio rate oscillator or low frequency oscillator.
	*Triangle*	Can be used as an audio rate oscillator or low frequency oscillator.
	*Sawtooth*	Can be used as an audio rate oscillator or low frequency oscillator.
	*Reverse Sawtooth*	This is the inverted (phase reversed) sawtooth Can be used as an audio rate oscillator or low frequency oscillator.
	*Square*	Can be used as an audio rate oscillator or low frequency oscillator.
	*White Noise*	White noise contains all frequencies, so it is aperiodic. The noise waveform is passed through a lowpass filter, whose cutoff frequency is controlled by the T**one** parameter.
	*Sample and Hold Triangle*	At this setting, a tempo synced triangle waveform is sampled and held at regular intervals. The interval length is controlled using **Coarse Tune**, which is between 1/128th note – Whole note. Since the waveform is synced to tempo, the oscillator mode should be set to 
	*Sample and Hold Sawtooth*	At this setting, a tempo synced sawtooth waveform is sampled and held at regular intervals. The interval length is controlled using **Coarse Tune**, which is between 1/128th note – Whole note. Since the waveform is synced to tempo, the oscillator mode should be set to 
	*Sample and Hold Reverse Sawtooth*	At this setting, a tempo synced inverted (phase reversed) sawtooth waveform is sampled and held at regular intervals. The interval length is controlled using **Coarse Tune**, which is between 1/128th note – Whole note. Since the waveform is synced to tempo, the oscillator mode should be set to 
	*Random Sample and Hold*	At this setting, a white noise waveform that is between 0.0 – 1.0 is sampled and held at regular intervals. The **Phase** (attack) and **Tone** (decay) parameters control the time it takes to rise/fall down to the next samples value.  If **Mode** is set to *LFO*, the sample and hold interval is controlled using the **Coarse Tune** parameter, and its duration varies between 0.1 seconds – 10.0 seconds. If **Mode** is set to *Sync Beat,* the sample and hold interval is synced to tempo, and its duration varies between 2 whole notes and 1/64 note.

	*Random Glide*	At this setting, a white noise waveform that is between 0.0 – 1.0 is sampled at regular intervals, and then the sampled values are linearly interpolated to construct the "*glide*" waveform.  If **Mode** is set to *LFO*, the glide interval is controlled using the **Coarse Tune** parameter, and its duration varies between 0.1 seconds - 10 seconds. If **Mode** is set to *Sync Beat,* the glide interval is synced to tempo, and its duration varies between 2 whole notes and 1/64 note.
 	Pattern 1 Sample and Hold Pattern 2 Sample and Hold	At this setting, *Pattern 1/2* steps are sampled and held at regular intervals. The **Phase** (attack) and **Tone** (decay) parameters control the time it takes to rise/fall down to the next step.  The number of steps can be specified by changing the values of **Loop Start** and **Loop End** parameters:  If **Mode** is set to *LFO*, the sample and hold interval is controlled using the **Coarse Tune** parameter and its duration varies between 0.1 seconds - 10 seconds. If **Mode** is set to *Sync Beat*, the sample and hold interval is synced to tempo, and its duration varies between 2 whole notes - 1/64th note.
 	Pattern 1 Glide *Pattern 2 Glide*	At this setting, *Pattern 1/2* steps are linearly interpolated to construct the *"glide"* waveform:  The number of steps can be specified by changing the values of **Loop Start** and **Loop End** parameters:  If **Mode** is set to *LFO*, the glide interval is controlled using the **Coarse Tune** parameter and its duration varies between 0.1 seconds - 10 seconds. If **Mode** is set to *Sync Beat*, the glide interval is synced to tempo, and its duration varies between 2 whole notes - 1/64th note.

4.1.3.3.2 Key Scaling

Key Scaling is used for amplify/attenuate the volume of the modulator based on the MIDI note number, similar to the EQ of an oscillator. The modulator's volume is multiplied by the level of the corresponding band. There are 20 bands, which are spaced equally at ½ octaves. Their values are between 0 – 2.

4.1.3.3.3 Volume Envelope

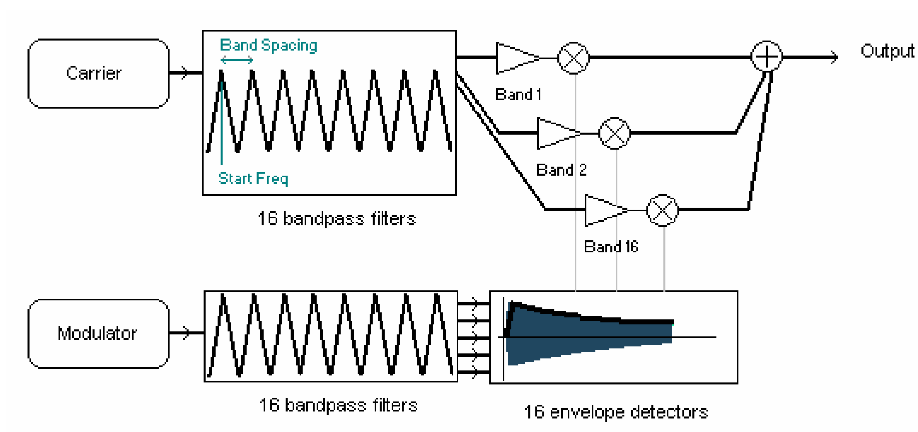
The volume envelope provides an envelope signal for the modulator so that its volume can be changed over time. Its parameters are similar to oscillator Volume Envelope parameters.

4.2 Vocoder

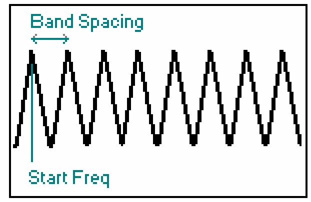
A vocoder combines two audio sources to produce a third "*blended*" sound. The two audio inputs used in a vocoder are called ***carrier*** and ***modulator***. To combine the audio sources, the vocoder

- Splits both the carrier and the modulator into frequency bands using bandpass filters
- Detects the envelope of each band in the modulator using envelope detectors.
- Multiplies each envelope level with the corresponding band in the carrier and adds all the bands to generate the "*blended*" output.

The following graph illustrates how the vocoder in *SynthMaster* works:

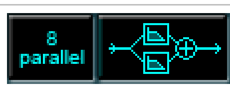
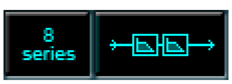
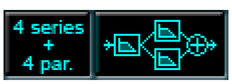
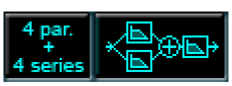


	**Bypass**	When **Bypass** is on, the vocoder is inactive and does not consume any CPU cycles.
	**Modulator**	The **modulator** could either be the synthesizer output, or the audio input (sidechain). Usually, it is the audio input (sidechain).
	**Carrier**	The **carrier** could either be the synthesizer output, or the audio input (sidechain). Usually, it is the synthesizer output of *SynthMaster*.
	**Envelope Follower Attack**	After the modulator input is split into 16 bands, the envelope of each band is detected by an envelope follower, which has 2 parameters: **Attack** and **Decay**.
	**Envelope Follower Release**	**Attack** controls how quickly the envelope rises when the incoming audio increases in amplitude. Similarly, **Release** controls how quickly the envelope falls back towards zero when the amplitude of the incoming signal drops.

	**Start Frequency**	**Start frequency** controls the center frequency of the 1st band. Its value is between 32 – 128 Hz, increasing exponentially. **Band spacing** controls the spacing between subsequent bands. Its value is between ¼ octave – 1 octave, increasing exponentially. 
	**Band Spacing**	
	**Bands 1-16**	Each band's volume can be set from the corresponding slider. **Band volume** is between 0.0 – 2.0, increasing linearly.

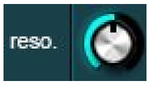
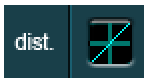
4.3 Parametric EQ

SynthMaster has a parametric equalizer that consists of 8 filters that can be connected in different combinations:

	*8 Parallel*	At this setting, all of the 8 filters are connected in parallel.
	*8 Series*	At this setting, all of the 8 filters are connected in series.
	*4 Series + 4 Parallel.*	At this setting, the first 5 filters are connected in series, and the last 3 are connected in parallel with the 5th filter.
	*4 Parallel + 4 Series*	At this setting, the first 4 filters are connected in parallel, and the last 4 are connected in series. This connection type is similar to 4 Series + 4 Parallel, but its output will be different when filters have distortion at their outputs.
	*Input + 8 Series*	At this setting, all of the filters are connected in series. The input is filtered by the filters and then added to the output of the last filter in the chain.

Each of the 8 filters has the following common parameters:

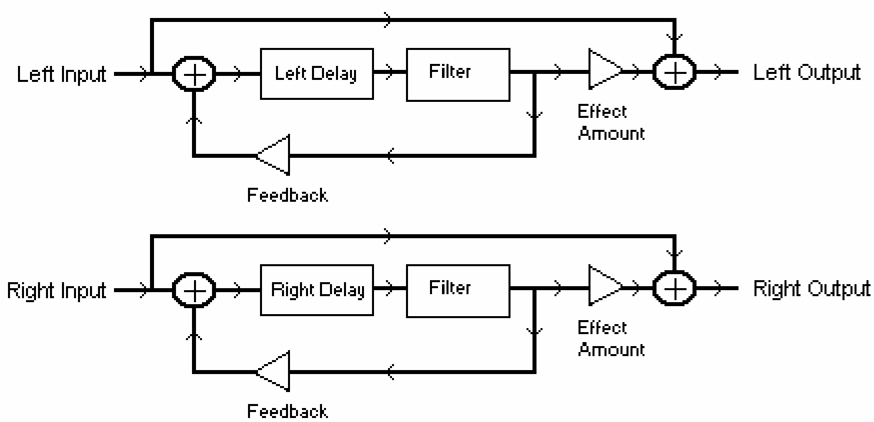
	**Bypass**	Using the **bypass** button, you can turn on or off a filter. When the filter is off, it is inactive and does not consume any CPU cycles.
	**Filter Type**	Filter types are explained in detail on pages 40-42 of this document.
	**Cutoff Frequency**	**Cutoff frequency** is the frequency at which the filter starts to attenuate frequencies above/below the cutoff, based on the filter type. Its value is between 12.4 Hz – 18.9 kHz, increasing exponentially.

	**Resonance**	**Resonance** controls the gain of the filter around the instantaneous filter cutoff frequency. High values of resonance correspond to high gains. Its value is between 0.6 – 100, increasing exponentially
	**Gain**	**Gain** is used to cut or boost the output volume of the filter. Boosting the output volume of a filter is especially useful when distortion (such as hard limiter) is applied after the filter. **Gain** is between 0.25 (-12 dB) – 4.00 (+12 dB), increasing exponentially. At its default setting, its value is 1.00 (0 dB)
	**Distortion**	With *SynthMaster*, it is possible to connect a waveshaper to the output of a filter in series, so that the filter output can be distorted. Distortion types are explained in detail on page 43 of this document.

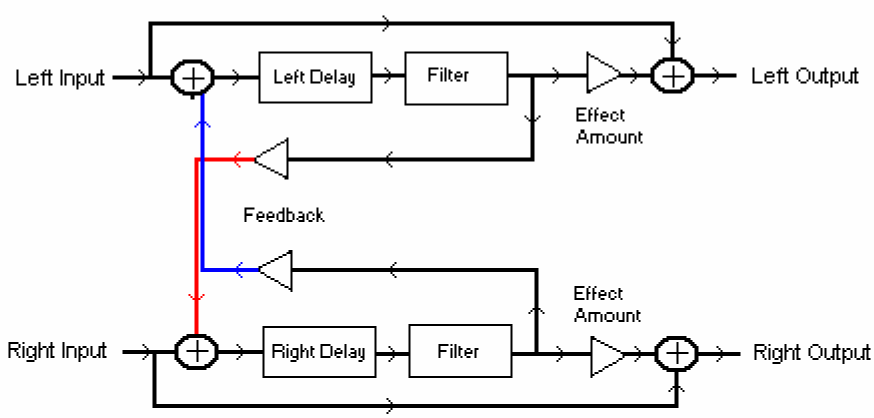
4.4 Echo

With the *echo* effect, you can create tempo synced stereo delay effects such as simple delay, feedback delay or ping pong delay.

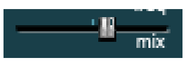
When the **echo type** is set to "*normal*", the left/right delay lines are not connected:



When the **echo type** is set to "*ping pong*", the left delayed signal is fed back to the right delay line, and similarly the right delayed signal is fed back to the left delay line:

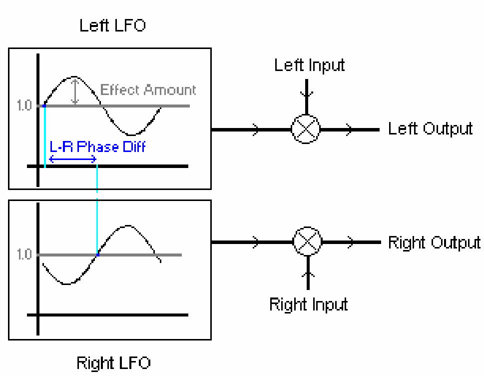


The *echo* effect has the following parameters:

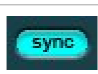
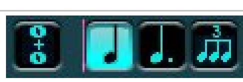
	**Bypass**	When **Bypass** is on, the *echo* effect is inactive and does not consume any CPU cycles.
	**Effect Amount**	**Effect Amount** controls the amount of wet effect output mixed with the dry input. Its value is between 0.0 – 1.0, increasing linearly.
	**Cutoff Frequency**	After the L/R inputs are delayed, they are filtered by a 1st order filter. **Cutoff Frequency** controls the cutoff frequency of this filter. When its value is greater than 64, the filter becomes *lowpass* filter. When its value is smaller than 64, the filter becomes a *highpass* filter.
	**Feedback**	**Feedback** controls the amount of delayed signal that is fed back and mixed with the dry input. Its value is between -1.0/+1.0, increasing linearly.
	**Left Delay**	**Left Delay** controls the tempo synced delay duration of left channel. Its value is between a whole note and 1/32nd notes.
	**Right Delay**	**Right Delay** controls the tempo synced delay duration of right channel. Its value is between a whole note and 1/32nd notes.

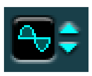
4.5 Tremolo

With the *tremolo* effect, you can modulate the amplitude of the stereo Synthesizer output with 2 separate LFOs (Left/Right). The following graph illustrates how the *tremolo* effect works:



The *tremolo* effect has the following parameters:

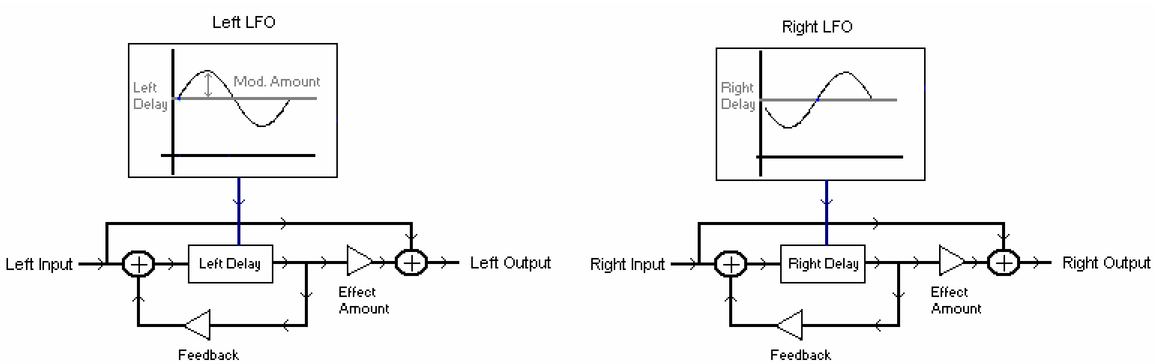
	**Bypass**	When **Bypass** is on, the *tremolo* effect is inactive and does not consume any CPU cycles
	**Effect Amount**	**Effect amount** controls the volume of the left/right LFOs. Its value is between 0.0 – 1.0, increasing linearly.
	**Tempo Sync**	When the **Tempo Sync** is on, the LFO frequencies are synced to tempo and **Beat Multiplier** dropdown control is used to set the synchronized frequency. Its value is between 2 whole notes and 1/64th note.
	**Synchronized Speed**	
	**LFO Speed**	When the **Tempo Sync** is off, **LFO Speed** controls the LFO frequencies. Its value is between 0.1 Hz – 10.0 Hz, increasing exponentially.

<table>
<tr><td>
</td><td>L-R Phase Diff</td><td>L-R Phase Diff controls the phase difference between the left / right LFOs. Its value is between 0 – 359 degrees, increasing linearly.
The following tutorial preset demonstrates how L-R Phase Diff parameter can be used to create panning effects: Chapter 4\ 019-Tremolo-Tempo Synced Triangle LFO-LR Phase 180.tfx</td></tr>
<tr><td>
</td><td>L-R Speed Ratio</td><td>L-R Speed Ratio controls the ratio between the left / right LFOs' frequencies. Its value is between 0.25 – 4.00, increasing exponentially.</td></tr>
<tr><td></td><td>LFO Wave Shape</td><td>The LFO wave shape can be one of the 4 basic waveforms:
<table>
<tr><td></td><td>Sine</td><td>A sine wave contains only the 1st harmonic, so it is the most basic waveform that can be used in sound synthesis.</td></tr>
<tr><td></td><td>Triangle</td><td>A triangle wave contains only odd-harmonics, just like a square wave. However the amplitudes of odd harmonics are not as high as they are in a square wave.</td></tr>
<tr><td></td><td>Square</td><td>A square wave contains only odd-harmonics.</td></tr>
<tr><td></td><td>Sawtooth</td><td>A sawtooth wave contains all harmonics, so it is widely used in sound synthesis using subtractive synthesis.</td></tr>
</table></td></tr>
</table>

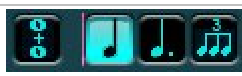
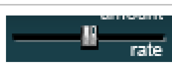
4.6 Chorus

With the *chorus* effect, you can make synthesizer sounds sound *"thick"*. *Chorus* effect is generated by adding a slightly delayed version of a sound to itself. The trick is that the delay length is modulated by an LFO (low frequency oscillator). When the delay length is very short (1-10 milliseconds) and there is feedback, the effect is called *Flanger*.

The following graph illustrates how the *chorus* effect works in *SynthMaster*:

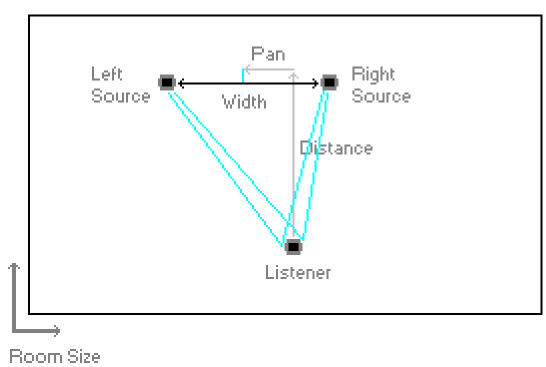


The *chorus* effect has the following parameters:

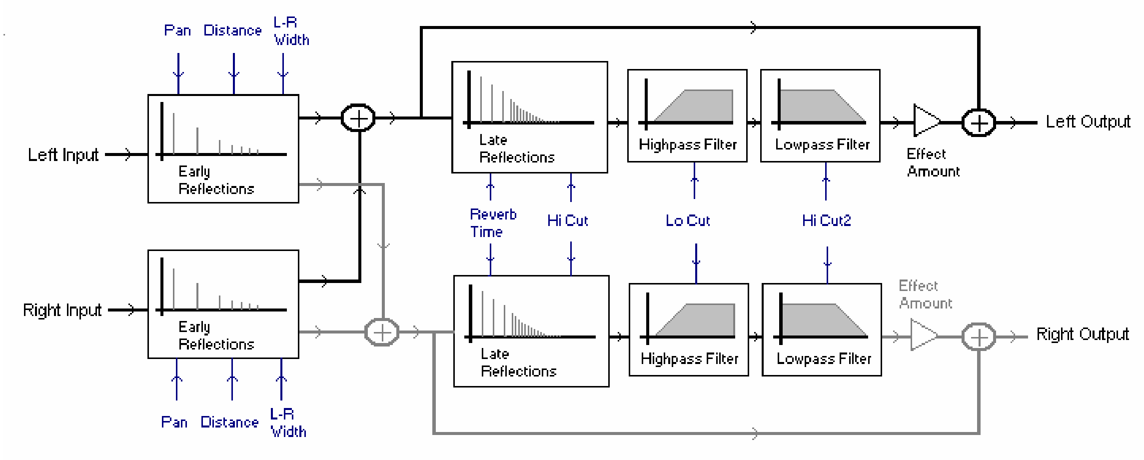
	**Bypass**	When **Bypass** is on, the *chorus* effect is inactive and does not consume any CPU cycles
	**Wet Amount**	**Wet Amount** controls the amount of wet effect output mixed with the dry input. Its value is between 0.0 – 1.0, increasing linearly.
	**Modulation Amount**	**Modulation amount** controls the volume of the left/right LFOs. Its value is between 0.0 – 1.0, increasing linearly.
	**Tempo Sync**	When the **Tempo Sync** is on, the LFO frequencies are synced to tempo and **Beat Multiplier** dropdown control is used to set the synchronized frequency. Its value is between 2 whole notes and 1/64th note. When the **Tempo Sync** is off, **LFO Speed** controls the LFO frequencies. Its value is between 0.1 Hz – 10.0 Hz, increasing exponentially.
	**Synchronized Speed**	
	**LFO Speed**	
	**L-R Phase Diff**	**L-R Phase Diff** controls the phase difference between the left / right LFOs. Its value is between 0 – 359 degrees, increasing linearly.
	**L-R Speed Ratio**	**L-R Speed Ratio** controls the ratio between the left / right LFOs' frequencies. Its value is between 0.25 – 4.00, increasing exponentially.
	**Left Delay**	**Left delay** controls the delay duration of the left channel. Its value is between 1.0 – 100 milliseconds, increasing linearly.
	**Right Delay**	**Right delay** controls the delay duration of the right channel. Its value is between 1.0 – 100 milliseconds, increasing linearly.

4.7 Reverb

With the *Reverb* effect, you can simulate the reflections of a sound that occur in a room. SynthMaster has a unique *reverb* effect that allows you to specify physical parameters such as **Room Type**, **Distance**, **Pan** and **L-R Width**. All of these parameters are used to calculate the early reflections portion of the reverb effect:



The overall block diagram for the reverb effect is as follows:



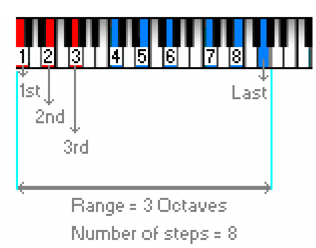
4.8 Arpeggiator

With the *Arpeggiator*, you can automatically play alternating notes or chords. *SynthMaster* has a very customizable *arpeggiator* that allows you to create multiple step arpeggios.

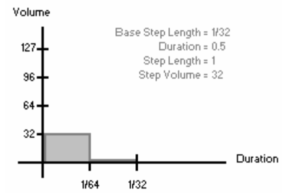
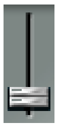
The working principle of the *arpeggiator* is very simple. Basically, you press multiple keys simultaneously and the *arpeggiator* plays the pattern which is defined by the *arpeggiator* parameters. For instance, let's assume you press C4, E4 and G4 simultaneously and the *arpeggiator* has the following settings:

The *arpeggiator* has 8 steps, and the range is set to 3 octaves. Since the *arpeggiator* is in *Arpeggiate* mode, one note will be played at a time. The played notes will be:

1. C4 (1st note)
2. E4 (C4 +1)
3. G4 (E4 +1)
4. C5 (G4 +1)
5. E5 (C5 +1)
6. G5 (E5 +1)
7. C6 (G5 +1)
8. E6 (C6 +1)



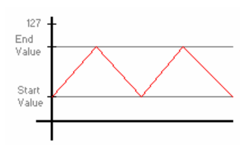
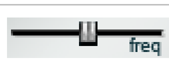
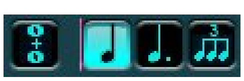
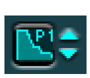
As seen above, each step has 3 parameters:

	**Step Length**	**Step Length** controls the length of a step, in multiples of the **Base Step Length**. **Duration** controls at what percentage of the total step length the note will be on. For instance, for the first step above, step length is 1. Since duration is set to 0.5, the note will be on for the first half of the length:  By changing **Duration**, you can create legato or staccato patterns. **Step Volume** controls the MIDI velocity (volume) of the step. Its value is between 0-127. By setting it to zero, you can create silences in the pattern. Using **Step Delta**, you can set which note will be played next in the pattern. Positive values refer to higher notes, and negative values refer to lower ones.
	**Step Delta**	
	**Step Volume**	

4.9 *MIDI LFOs*

SynthMaster has 2 *MIDI LFOs* that can be used to generate continuous MIDI control change (CC) messages. Since multiple instrument parameters can be linked to a single MIDI controller, it is possible to change the values of up to 16 instrument parameters by using a single MIDI LFO.

Each *MIDI LFO* has the following parameters:

	**Bypass**	**Bypass** is used to turn on/off the LFO.
	**Start Value**	**Start Value** and **End Value** control the minimum and maximum values of the LFO output. Their values are between zero – 127: 
	**End Value**	
	**Controller Number**	The LFO generates CC messages for the controller specified by **Controller Number**.
	**LFO Speed**	When the **Tempo Sync** is on, the LFO frequency is synced to tempo and **Beat Multiplier** dropdown control is used to set the synchronized frequency. Its value is between 2 whole notes and 1/64th note. When the **Tempo Sync** is off, **LFO Speed** controls the LFO frequency. Its value is between 0.1 Hz – 10.0 Hz, increasing exponentially.
	**Tempo Sync**	
	**Synchronized Speed**	
	**LFO Wave Type**	LFO wave types are similar to modulator wave types, which are explained in detail on pages 47-48 of this document.