My Synthesizer Laboratory

- ⌘ *A NOSTALGIC RETROSPECTIVE*
- ⌘ *THREE KIND OF REAL SYNTHESIS*
- ⌘ *THE VINTAGE SYNTHESIZER*
- ⌘ *THE MANUAL*

A NOSTALGIC RETROSPECTIVE

The synthesizers of today are amazing. The reason is that they use recordings of real instruments. Even sounds from vintage synthesizers are recordings of, well, vintage synthesizers. That is why they sound so great.

However, this was not possible to achieve in the early days of sound synthesis. The modern synthesizers use sound that is stored in digital memories, same as the memory in your computer. Memories were simply far too expensive when the early synthesizers were built. The memory used in a modern synthesizer contains memory that would have costed hundreds of thousands of dollars, or even many million dollars. And the amount of memory was simply not available on the market by the time.

Therefore, vintage synthesizers had to actually generate the sound, to synthesize it. That is why they are called synthesizers. The synthesizers of today really are not synthesizers, because they do not really synthesize the sound. Still we call them synthesizers, and that is all right. They do not only use recorded sounds, but also modify and enhance them to make it an as real musical experience as possible, which is part of what 'real' synthesizers do after having synthesized a sound.

In the youth of the synthesizers there were two different philosophies regarding the use of synthesizers. Some people adapted both of those; some preferred their choice, while some even disliked synthesizers just because they did not sound like 'real' instruments.

The philosophies were of course those that the synthesizer could be used to replace real instruments, or to constitute a new kind of instrument.

It is true that they were a poor substitute for 'real' instruments, but did they really need to be that? Synthesizers could create amazing sounds never heard before, so to me it is absolutely a new kind of instrument. I see no reason not to think that way. There are so many different kinds of instruments, and many are inferior to the synthesizer, if thought of as competitors. But thinking of instruments as competitors and only use the most versatile instrument is really not a good idea. That would rule out a lot of instruments, like a drum, that cannot play different pitches, the didgeridoo, which still makes very interesting sounds etcetera. I could continue

making examples, but I imagine you get the picture. The old style synthesis could make sounds no one even thought about using in music.

Thanks to a lot of musicians and composers the synthesizers came to be used as its own type of instrument. I will not list any names, bands or records, but I must mention that I remember when a friend played Pink Floyd's album Umma-Gumma for me. I had never heard anything like it before, and I loved it immediately. Since then I have heard and collected a lot of albums, and many of them used synthesizers to create fantastic fantasy sounds.

My Synthesizer Laboratory lets you put together your own synthesizer using your own ideas of how to synthesize sounds. If you are too young to remember the days they ruled, you will have a revelation, if you do remember the days, you will have not only a trip down memory lane, you will, have a laboratory that lets you play with sounds anyway you like. That is also true even if you are too young to remember those days.

Read more in the next chapter about how it all works and in the chapter 'Manual' how to use the laboratory.

THREE KIND OF REAL SYNTHESIS

There are essentially three ways used to synthesize sound:

- ⌘ The addition method
- ⌘ The subtraction method
- ⌘ The frequency modulation method

Before we get into the different methods, you need to know a few things about sounds. Each sound has a characteristic, a timbre. A tone from e.g. an instrument sounds a certain way. The most fundamental sound is the sine wave. A sine wave is a waveform that cycle in an even way, like a spinning wheel or a pendulum. A point on the spinning wheel goes round and round and never deviates from its path, while the weight at the bottom of the pendulum swings forth and back. When a sound with that characteristic hits our ear, we hear a tone that virtually has no characteristic. Or you could call that the lack of characteristic is its character.

If you whistle a tone, you are creating a sound rather similar to a sine wave. If you play a clarinet, you make waves that have a lot more character. So, what is it that makes that character?

Starting with a smart guy by the name Fourier, and some other mathematicians that followed up his work, a theory was formed. That theory stated that any waveform can be broken down into the sum of a number of sine waves. So, we should be able to create a sound that is similar to that of a clarinet, or any other waveform, if we only could identify, create and put together those sine waves.

The sine waves that create the character of the sound are all of higher frequency than the note played. The note played is the base frequency, and all other sine waves are called ‘overtones’. It is those overtones that make the character of the sound. (However, some synthesizers allows you to add sub frequencies to your sound, but that does not matter to the theory described here, since the sub frequency becomes the new base frequency.)

The sound waves and overtones are created by oscillators. There has been some different kind of oscillators, running in the range from cogwheels via radio tubes to transistor circuits and integrated circuits. Anything that can oscillate can also be used to create a tone.

My Synthesizer Laboratory is made in such a way that you can not only create sounds using any of those three methods, but you can also combine them.

Now back to the three different methods.

The addition method uses mostly sine waves. One is for the base frequency. Then waves of higher frequencies are added to the sound in order to make the characteristic of the sound.

Just think of a Hammond style organ. It has a number of sliders with different overtones that can be added by adjusting the sliders. Even a church organ uses a similar method where different sets of pipes can be added by pulling their respective knobs.

Just because we add the overtones, this method is called addition synthesis.

The subtraction method works the other way around, and is the most common type used by vintage synthesizers. In this method a waveform that already has a lot of overtones is used, and a filter is then used to reduce the overtone spectra.

The square wave and the saw-tooth wave are typical waves used in subtraction synthesis. If you ever saw an old-fashion synthesizer you probably noticed that those waveforms were available to select.

The frequency modulation **method is a bit special. As far as I know it was Yamaha that invented, or more correct, invented the use of frequency synthesis, to sound synthesis with their DX series of synthesizers. Frequency modulation had been used for other purposes before that, but as far as I know, never for sound synthesis.**

The frequency modulation uses two or more sine waves, usually all or some of them using the base frequency, the frequency corresponding to the keyboard key pressed. When modulating a sine wave with another sine wave that has the same frequency or a multiple of it, that creates overtones. This is because the modulated oscillator changes frequency following the value of the modulating oscillator, but still comes to the end of the cycle all at the correct time, or a multiple of the time. Thus, the base frequency is still there, but since the oscillator changes frequency *within* each cycle, overtones are created.

In addition to the three methods mentioned above, My Synthesizer Laboratory has some rudimentary support for wave files of the type .wav and .mp3. This is a bit off topic, but since modern synthesizers use waves, it feels like it might be fun to work with them too. However, the files are only played. Filters and envelopes are not applied to the sounds from the wave files.

Wave files are used in two types of setup:

- **Instrument**
- **Drum set**

The difference is that an instrument has some samples of a real instrument at different pitches. If there are keys omitted, those keys are using the nearest wave file transposed to sound at the pitch corresponding to the played key.
A drum set has one wave file for each keys that should be in use. The other keys does not produce any sound, and the wave file is not transposed to any other frequency. So, an instrument has one sound that is played at different pitch depending on key pressed, while a drum set has different sound for each key, and the sound is never transposed.

In the waveform selector of the oscillator GUI you find instrument and drum set last when selecting wave form. They are called ‘Wave’ and ‘Drumset’ respectively.

When using instrument or drum set you will have to supply a folder containing the actual sampled instrument sounds.

THE VINTAGE SYNTHESIZER

COMPONENTS OF A VINTAGE SYNTHESIZER

Vintage synthesizers contained a number of different modules. They did not all use all kind of modules, but at least some of these modules:

- ⌘ Oscillator
- ⌘ LFO oscillator
- ⌘ Filter
- ⌘ Pitch envelope generator
- ⌘ Envelope generator
- ⌘ Keyboard
- ⌘ Pitch bender
- ⌘ Modulation wheel

Oscillator:

The oscillator generated the sound wave. Its frequency is depending on which key is pressed and the position of the pitch bender. It could generate some of these wave shapes:

- ⌘ Square wave
- ⌘ Saw tooth wave (up and/or down)
- ⌘ Triangle wave
- ⌘ Sine wave
- ⌘ White noise
- ⌘ Random values (like low frequency noise)

LFO oscillator

The LFO oscillator is a low frequency oscillator that does not change pitch depending on key pressed nor the position of the pitch bender. It has some knob or slider to set its frequency. It is used to modulate some of the other modules. E.g. the oscillator could be frequency modulated to emulate vibrato, or amplitude modulated to emulate tremolo.

Filter

The filter is normally used to reduce a part of the oscillator frequency spectrum. Filtering a square wave can e.g. soften its sound, or emulate a wah-wah effect if the center frequency of the filter can be changed.

Pitch envelope generator

This allows for an oscillator to change pitch over time. The time starts when a key is pressed and changed the pitch until it ends, the time is finished. Some pitch envelope generators also has a time slice that starts when the key is released. This is useful only if the envelope generator has a release time, otherwise the tone is silenced when the key is released.

Envelope generator

The envelope generator changes the oscillator amplitude over time. It normally has three times and one level:

- Attack time, the time from the key press till oscillator full amplitude is reached. The oscillator starts at zero amplitude and rises to full amplitude during attack time.
- Decay time, the time after attack time is done until oscillator reaches the requested sustain level. The oscillator lowers its amplitude from full down to the sustain level.
- Sustain level, the level held after attack and decay until key is released.
- Release time, the time it takes for the sound to decrease from sustain level (or the current level when the key was released) down to zero after the key is released.

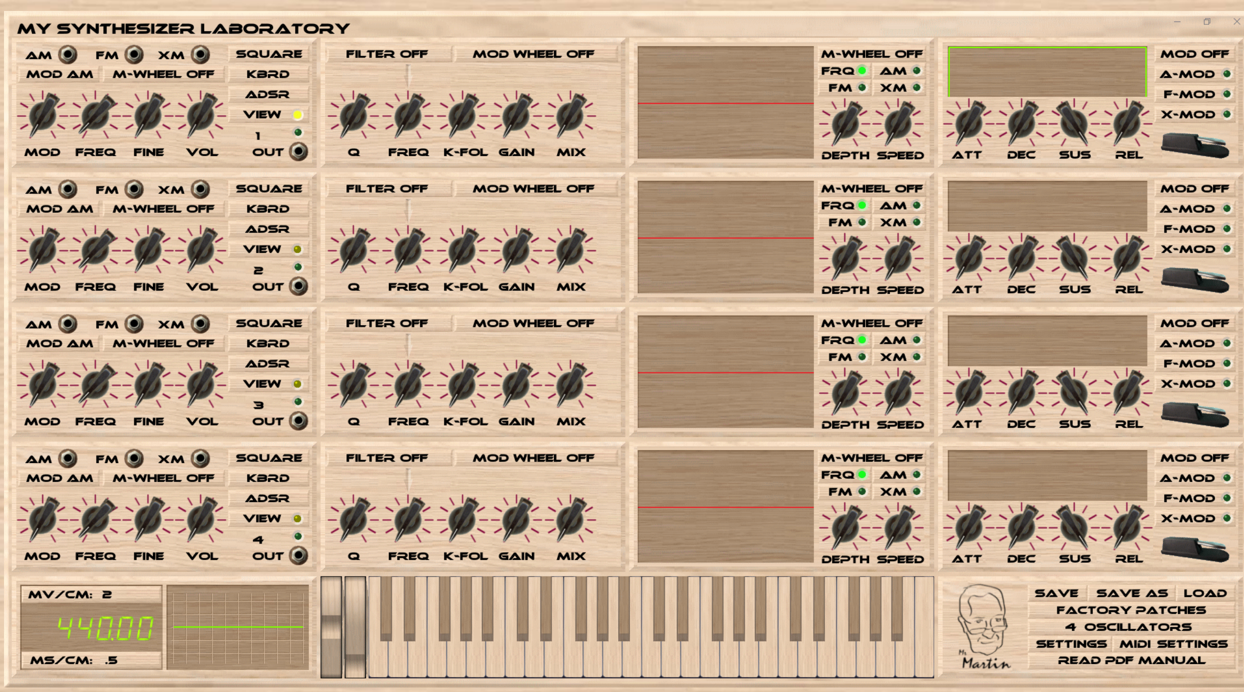
The envelope generator can be used for e.g. let the sound 'tune in' and/ or 'tune out' softly, but can also generate a short start 'click' by lowering the sustain level and add a short decay time, in order to emulate e.g. the hammer sound of a piano or a plectrum releasing a guitar string.

The keyboard and the pitch bender used some different techniques to change the frequency of the oscillator. There was also a possibility to use other sources to change the frequency since early synthesizers used resistance or voltage to control the oscillator pitch, however, the MIDI eventually emerged and synthesizers started to us that protocol. This meant that the keyboard and other controllers suddenly were a separate part and no more a part of the synthesizers. Even if it did not look that way you could on most synthesizers see a setting that connects/disconnects the keyboard from the rest of the synthesizer.

The modulation wheel should also not be considered a part of the synthesizer since it communicates via MIDI just like all controllers do. It can be used to control different parameters in the synthesizer, but are mostly meant for a specific function, therefore assigned a controller id.

THE MANUAL

My Synthesizer Laboratory consists of a few modules. All together it looks like this after startup:



The modules are:

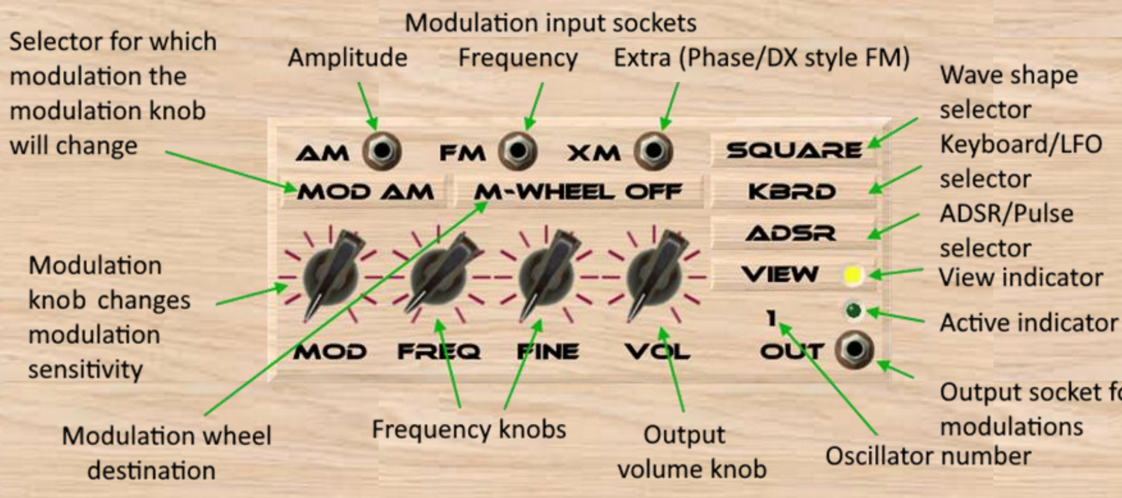
- ⌘ Oscillator
- ⌘ Filter
- ⌘ Pitch and ADSR envelopes
- ⌘ Display
- ⌘ Pitch bender and modulation wheel
- ⌘ Keyboard
- ⌘ Control panel

Oscillator:

The oscillator can be modulated by another oscillator by connecting one oscillator's OUT socket to another oscillator's AM, FM or XM input socket. In order to avoid circular modulation connections there is a rule: The modulated oscillator must have a higher number than the modulating oscillator.

The oscillator can operate as sound generator following the keyboard or as a low frequency generator, LFO, with a fixed frequency. As LFO it can modulate other oscillator to create various effects. In order to operate as a sound generator the volume must not be set to zero. When raised, the green indicator lights up to let you see which oscillators are actually producing sound.

The keyboard modulation wheel can be set to modify any of the knob settings for more dynamic playing.



The oscillator has the following wave forms:

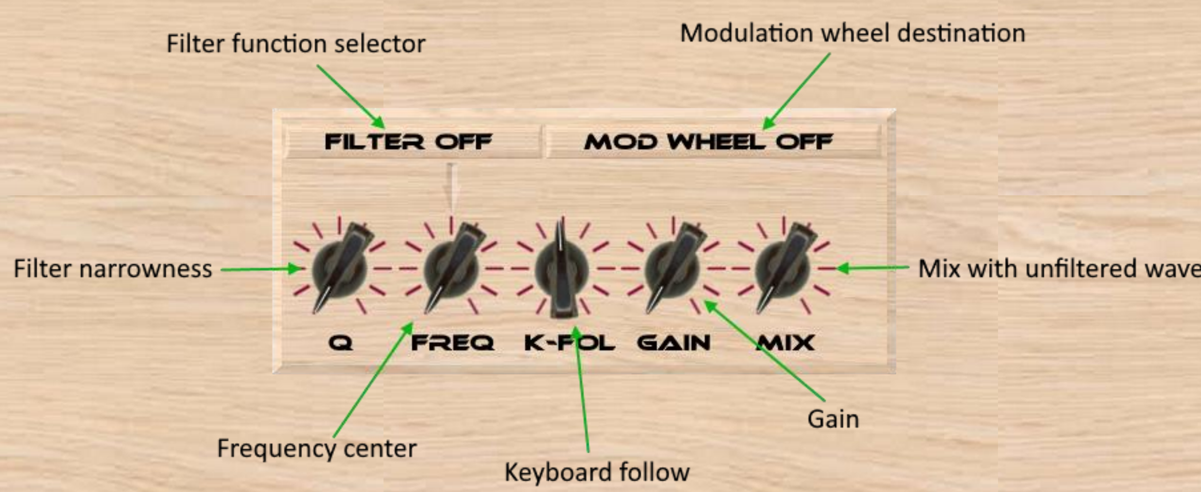
- ⌘ Square
- ⌘ Saw-tooth down
- ⌘ Saw-tooth up
- ⌘ Triangle
- ⌘ Sine
- ⌘ Random
- ⌘ Noise

Only one oscillator can be monitored in the display at any time since there is always only one display:

You can select which oscillator to monitor by clicking on its View button/indicator. Additionally most controls you change fin the oscillator GUI will make that oscillator the viewed one. On the left hand you initially se the frequency that would be played if you press the standard A key. It will change when you change the frequency. If you play some keys you will see the frequency of the last key released when no more keys are pressed.

On the right hand you see an oscilloscope image of the wave produced. It will change when you select another wave shape or change the filter or DX style FM synthesis parameters. The time and voltage settings for the oscilloscope are located above and under the frequency display.

Each oscillator is connected to a filter. Depending on layout you may have to use the View selector to make setting for a certain filter.



The function selector has five settings:

- ⌘ Filter off: No filtering is done
- ⌘ Fixed: Center frequency is set only with the FREQ knob
- ⌘ ADSR Pos: Frequency center is set by the FREQ knob with the value of the ADSR envelop generator added to it
- ⌘ ADSR Neg: Just like ADSR Pos but the ADSR value is subtracted instead causing an inverted effect
- ⌘ Pitch env: Like ADSR but using the pitch envelope generator instead.

The modulation wheel selector allows you to control any one of the five knobs from the modulation wheel of your keyboard.

The Q value controls how narrow the filter is, i.e. how few of the frequencies of the frequency spectrum the filter lets pass.

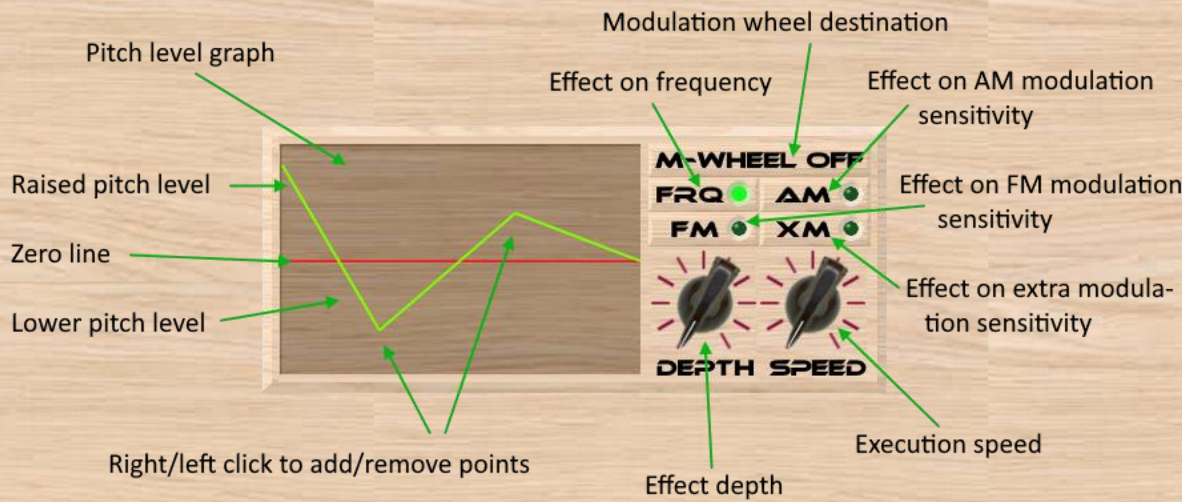
The frequency center moves the filter around in the frequency spectrum, just like a wah-wah pedal.

The keyboard follow decides how much the center frequency is affected by what key is pressed on the keyboard. When set to zero, the filter is not affected at all by which key is pressed.

The gain lets you compensate the volume loss that results from filtering out parts of the frequency spectrum.

The mix knob lets you pass the unfiltered waveform through mixed with the filtered version.

The pitch envelope changes the pitch over time at key on, but can also be used to change the modulation sensitivities of the oscillator.



The pitch level graph is for editing and viewing the pitch change progress. Start by left clicking on the left hand of the graph at the height you want the pitch to start. A line is created that ends on the right side with the value zero. That makes the oscillator use the correct pitch after the pitch envelope time is over.

Add and remove additional points using left and right mouse button. If you remove any of the points when only two points remains, both will be removed. You can only remove the rightmost point if there are more than two points. Doing this lets you set an endpoint at a frequency that is not the correct one for the key pressed, but can be useful for some other effects, like when controlling modulation.

The modulation wheel selector allows you to control either the depth knob or the speed knob from your keyboard.

The depth knob sets how much the pitch is changed. It kind of changes the amplitude of the effect of the graph.

The speed knob sets how fast the graph is executed.

The modulation indicators can be turned on or off in order to let the pitch envelop control the depth of an effect used on the oscillator. Any combination of the frequency and modulation indicators can be set causing the pitch envelope to have multiple simultaneous effects.

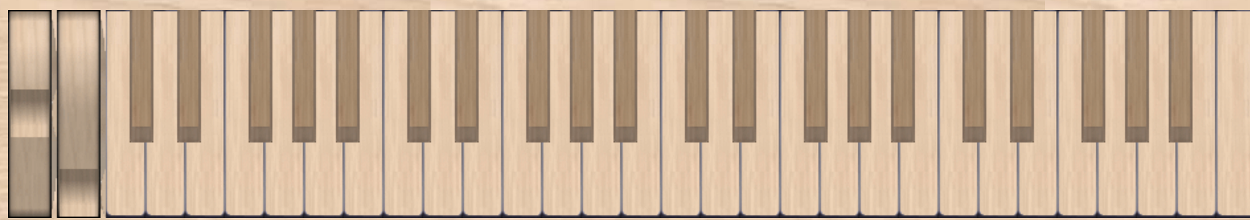
The envelope generator, ADSR is a graphical tool to control the level over time at key on and key off. It also has an indicator/button for the hold pedal. All oscillator has its own ADSR but the hold pedals are linked to always be pressed and released together since you can only have one hold pedal.

- ⌘ A – attack time, the tone starts slowly rather than abrupt
- ⌘ D – decay time, after the attack time is finished, the tone level uses the decay time to reach the set sustain level
- ⌘ S – sustain level, the level of the tone after attack and decay times are finished, and until the key is released
- ⌘ R – release time, the time it takes for the tone to fade out after the key is released

Any one knob can be selected by the modulation wheel destination selector to follow the modulation wheel. Any of the oscillators modulation input sensitivity can also be controlled by the ADSR.

The filter has settings to let the center frequency vary following the ADSR as it is or inversed. While using the ADSR to control the filter it is often better not to use the ADSR for changing the tone volume, thus the oscillator has a button to select between ADSR and pulse. Setting it to pulse causes the volume of the oscillator to be at set level without following the ADSR.

The keyboard. You do not have to connect a MIDI keyboard in order to play notes, but to really play music you really need one.

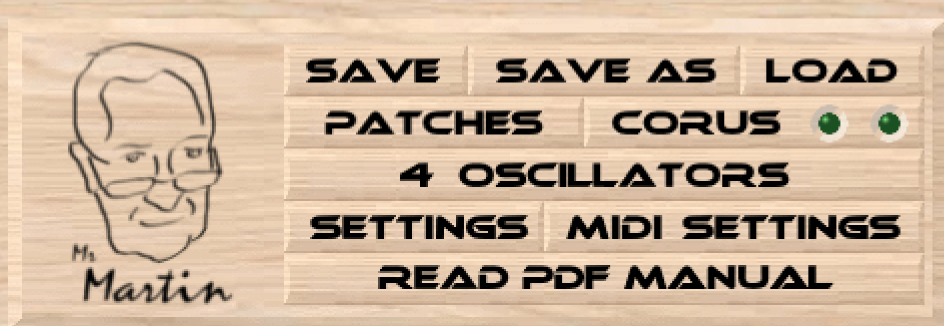


On the left side there is a pitch bend wheel and a modulation wheel. You can use then like vertical sliders. If you use a MIDI keyboard they will follow the wheels of the keyboard. If you use a controller, like Novation LaunchKey, they will also follow CC21 and CC22.

The keys are velocity sensitive. Since you cannot hit a key harder using the mouse, click on different part of each key in order to get different velocity. The velocity is lowest at the top of the keys and higher further down. Vintage synthesizers did not have velocity sensitivity from the beginning, but eventually most of the newer models were equipped with it. My Synthesizer Laboratory has settings for each oscillator where you can activate or deactivate velocity sensitivity.

You can actually try a chord on the keyboard by pushing down a key with the mouse and then move the mouse pointer away from the key. It will then not get the key off event, thus keep sounding. Now you can add notes to play a chord by doing the same with other keys. Do the opposite in order to turn of the keys. You can also toggle the hold pedal to turn off all sounding oscillators.

The control panel is there to let you do settings, save and load you patches, try the factory patches, select chorus, select layout and to admire my auto-portrait.



Since polyphony allows for six keys to be played at the same time, you can here choose what will happen if you press a seventh key. Should the 'oldest' key be released, or should the 7'th key stay dead?

You can select speed and depth for all three chorus selections.

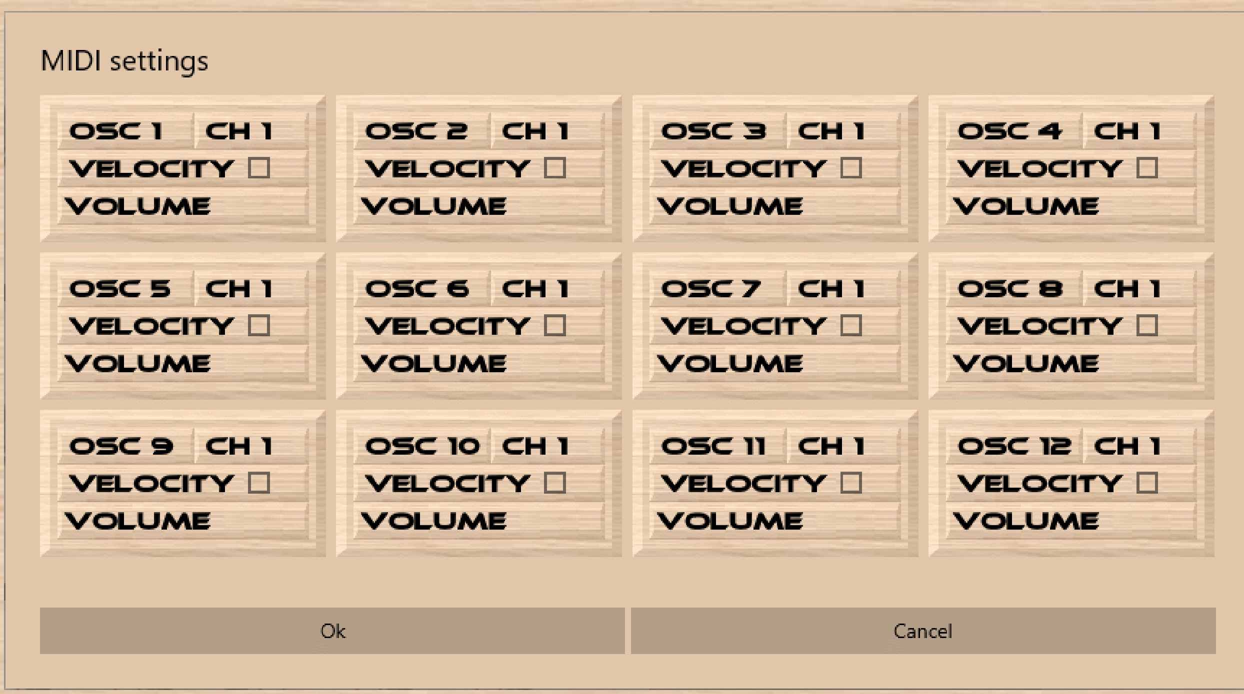
You can let My Synthesizer Laboratory listen to all MIDI input devices, or select one in the list.

Sine tones sounds weaker in the lower frequency range. You can select a factor for gradually raising the amplitude under 200 Hz.

You can map most one-byte Control Change (CC) messages to any knob you like. Note that this is also depending on what MIDI channel is carrying the message and which oscillators are listening to that channel.

Each oscillator is originally set to listen to MIDI channel one. You can here select any MIDI channel or all MIDI channels.

Velocity sensitivity can be activated for each oscillator, and set to affect volume, modulation or both.



THE DIFFERENT LAYOUTS

Since each oscillator 'owns' a filter, a pitch envelope generator, an ADSR envelope generator and a display and all those modules except the display has multiple controls you can manipulate, displaying all twelve oscillators with all their modules would take up too much screen space. However, the display has almost no controls. It will display the frequency and the graph for the currently viewed oscillator.

When changing any control, except the oscillator modulation sensitivity knob, in any of the oscillator modules, that will automatically select that oscillator as viewed. You can see that by looking at the LED on the View button. You can also manually select to view an oscillator by clicking its View button.

However, having to do that makes it less efficient to use the user interface. Therefore there are a few different layouts. You can elect to have access to fewer oscillators thus getting a better overview for each oscillator or more oscillators at the cost of having to use the View button more often. The four layouts are:

- ⌘ Four oscillators: Gives you access to all module controls direct.
- ⌘ Six oscillators: Gives you access to all filter controls, but both pitch and ADSR envelop generator GUI's are shared and you need to make sure to click view for the oscillator for which you are adjusting envelop generator settings.
- ⌘ Eight oscillators: Same as for six oscillators, except that there are only four filter GUI's shared by two oscillators each.
- ⌘ Twelve oscillators: All oscillators have its own GUI but all other controls share one common GUI.

Note that the oscillator GUI's are for the first oscillators. E.g. four oscillators always displays GUI's for oscillators number one through four, six oscillators displays GUI:s for oscillators one through six etcetera.

Also note that all oscillators are still present, even if you do not see them in the GUI. You may freely change GUI at any time without upsetting your settings.

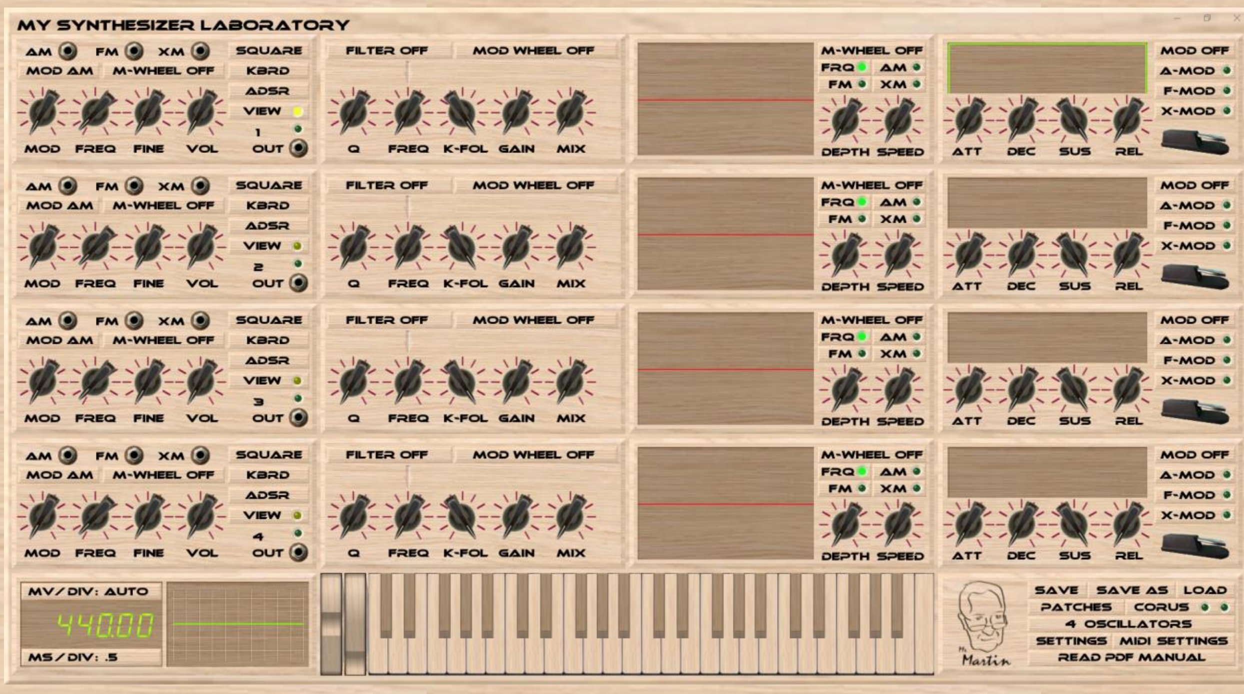
In some layouts there are one or two free spaces for which I found no use, so I put in some pictures with graphics related to sound synthesis.

If you are using a GUI with more oscillators you will have to use the 'View' to make the envelope generators show settings for a specific oscillator.

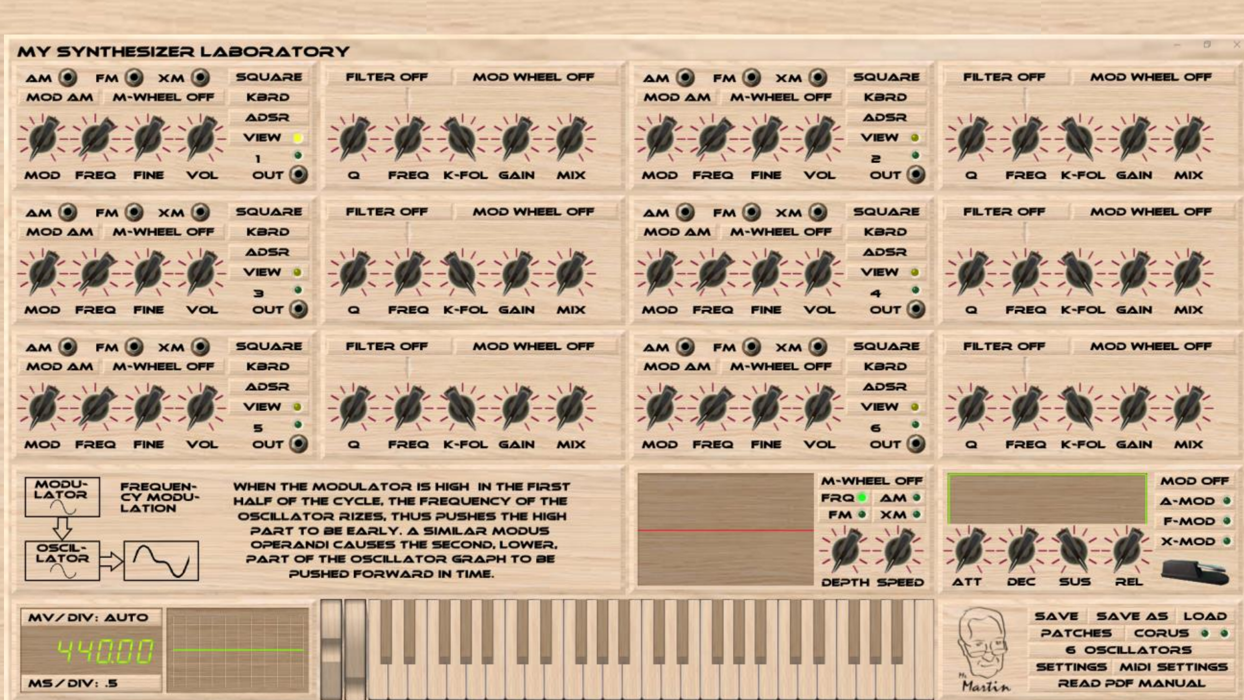
If you are using the GUI with eight oscillators, you have only four filter GUIs and must also verify that the filter GUI is targeting the correct oscillator of the two above.

If you are using the GUI with twelve oscillators you will have only one filter GUI. Make sure you are setting the filter for the oscillator you want to filter.

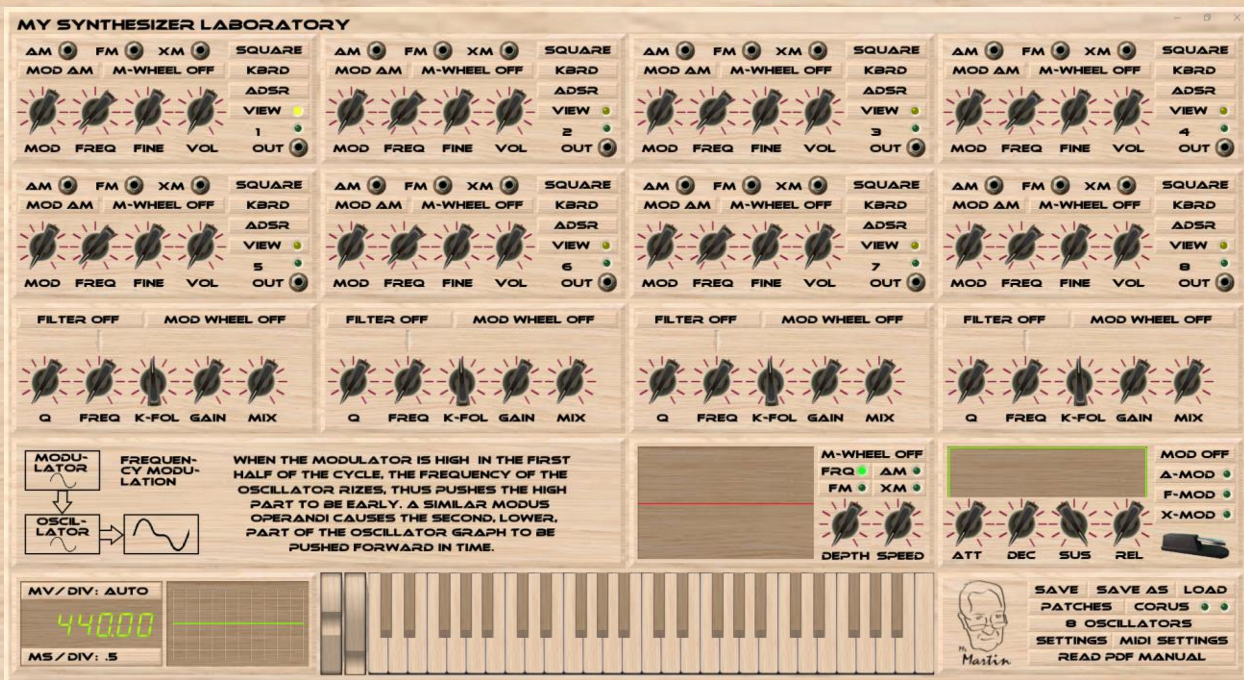
Here are the four layouts, layout with four oscillators:



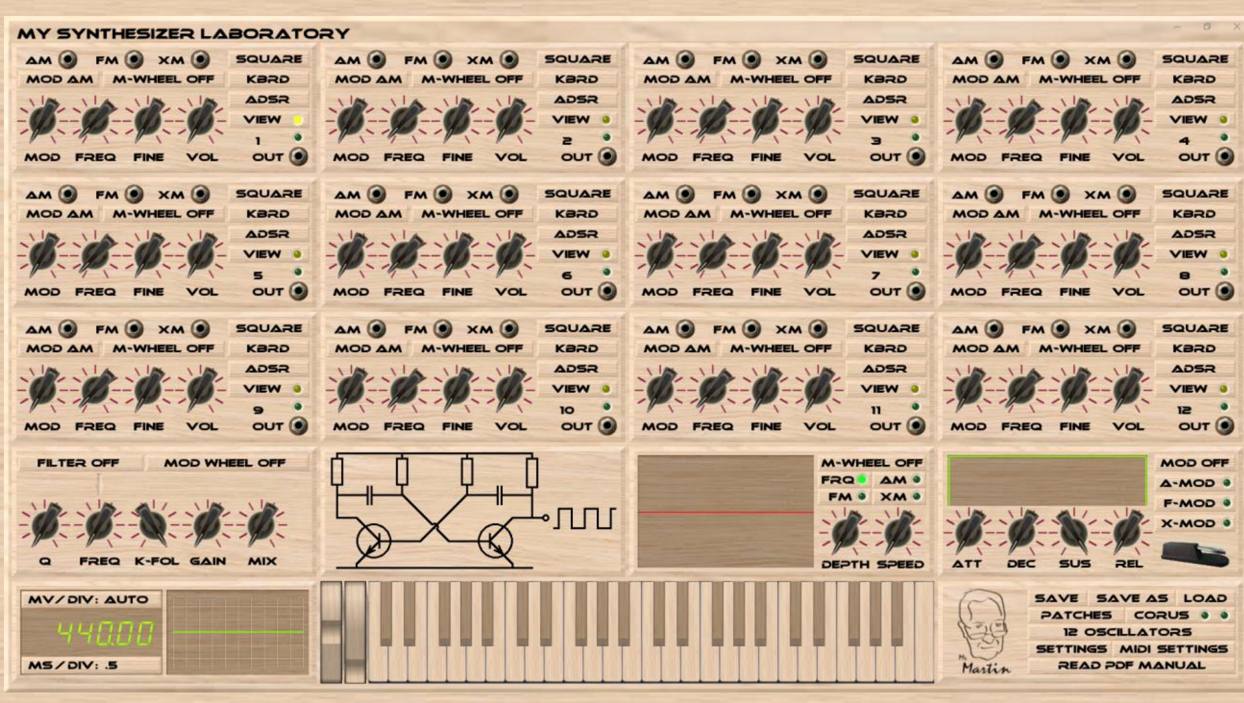
Layout with six oscillators:



Layout with eight oscillators:



Layout with twelve oscillators:



Making connections:

You can connect one oscillator with another oscillator by clicking the output of the one that will act as modulator, and one of the inputs of the one that will be modulated. You can click them in any order, but if you plan to use a y-cable to connect one output to more than one input, you must click the output first. See why in the next paragraph.

You can disconnect a cable by simply clicking the input it is connected to.

You can only connect to an oscillator with a higher number. The number is written next to the output socket. This is enforced in order to avoid circular connections. Such connections are bad enough when applying them to physical electronics. Doing it to a computer program will however cause a stack overflow and crash the program and possibly even the computer. So it is a good rule.

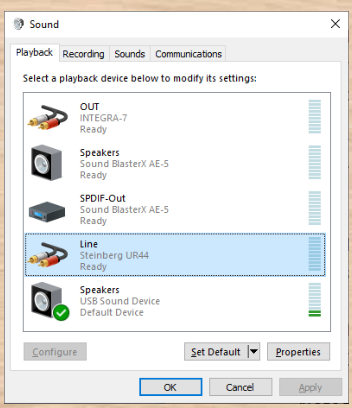
Using a DAW:

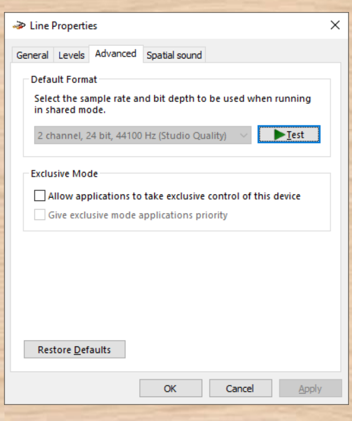
Since My Synthesizer Laboratory can assign different MIDI channels to the oscillators it is also possible to play from a DAW. You can use the oscillators to create different sounds for your instruments of your intended orchestra, and connect them to different MIDI channels.

DAW means Digital Audio Workstation, but in the time of vintage synthesizers they were just MIDI editors, and the advanced beasts available today can still play MIDI.

One problem is, however, that My Synthesizer Laboratory cannot be seen by a DAW. Why? Because it is not a MIDI driver, it is an application. Adding a MIDI driver would take a lot of time and money because it must be accepted by Microsoft and be implemented in Windows. But there is a simple solution. I use a MIDI interface to which I connect MIDI OUT to MIDI IN. Then I tell the DAW to send to that interface. Since My Synthesizer Laboratory detects all useful MIDI devices it will received what goes through that MIDI OUT to MIDI IN cable.

Note that some DAW programs, e.g. Cubase, claims full control over your MIDI interface, thus preventing My Synthesizer Laboratory from detecting it. The remedy is to start My Synthesizer Laboratory *before* starting the DAW. Windows also has a setting to allow or deny applications to take exclusive control over a device:





Right click the little speaker in the Windows 10 taskbar and select ‘Sounds’. In Sounds, select your MIDI device and click ‘Properties’. In Properties make sure that ‘Allow applications to take exclusive control of this device’.

My Synthesizer Laboratory makes a lot of calculations in order to synthesize and assemble the sound from all oscillators involved. It therefore needs a powerful computer.

I have tested it on an Acer Aspire 7552G laptop with an AMD Athlon II Dual-Core, Processor model P340 with a frequency of 2.2 GHz, 4 GB of memory, and the program runs on it. But using one oscillator with a filtered square wave limited the functionality in that sense that I could only play one note at a time. When playing a chord, the processor did not quite managed to deliver all calculations, leaving drop-outs in the sound.

I have developed it on a computer with 16 Gb of memory and an Intel i9-11900F 12-core processor and it runs fine on it.

Another test using a similar computer with an Intel i5 processor revealed also some drop-outs when using a more complex synthesis and playing chords, but it was still useable.

Also tested on an Asus ROG Strix G513 and it worked well.

All in all, you do not have to buy a super computer as long as you do not make too complex synthesis and play too many notes at the same time. Note that you have twelve oscillators that can each play six notes at the same time, adding up to 72 oscillators synthesizing sound simultaneously, depending on your setup.