

the V-Synth TweakBook

Second Edition (Preview)

Using the Roland V-Synth series
synthesizers to cross new sonic frontiers

Artemiy Pavlov



The V-Synth TweakBook

This book dives into the synthesis and sound processing depths provided by Roland's amazing V-Synth series synthesizers. The featured educational material describes both the underlying concepts and possible usage of available synthesis blocks and technologies, and is provided with sound structure diagrams, parameter map, example patches and audio demos.

Copyright

All contents and materials of this book are copyright 2005-2007 Artemiy Pavlov, all rights reserved. Redistribution or publishing of this book in whole or part, in any form, without an agreement with the author is strictly prohibited and is subject to prosecution by law.

About The Author

Artemiy Pavlov was born in 1981 in Kharkov, Ukraine, in a family of an electronics engineer and a math teacher, both of which were also musicians. He naturally paid a lot of interest in sound and it's physical aspects, and soon from passion this developed into a profession. Among others, Artemiy's recent creations include factory materials for Roland and Native Instruments, development of synthesizer plug-ins, as well as popular publications on sound design. He also manages the worldwide Roland Clan community he founded in 2003, and is a PhD student at the Kharkov National University for Radioelectronics. Artemiy is married since 2001 with a daughter born 2004.

You can read more about Artemiy Pavlov at these sites:

- ▶ <http://artemiypavlov.com>
- ▶ <http://sineshine.com>

Contents

Introduction	4
VariCraze	6
Welcome to Elastic Audio	6
Beat Control	7
Tame the Time	8
Modulation Exploration	10
Envelope Modulator	10
COSMology	12
Side-Band Filter	12
TB Filter	13
Parameter Tweak Map	15
Oscillator	15
COSM	15
TVA	16
Credits	17


Introduction

Before you start

This book is a great knowledge resource for those who want to take their sound design skills to a new level and learn how to use the V-Synth's range of building elements both alone and in various simple or complex combinations. To make the instructions herein compact and easy-to-read, it is assumed that:

- ▶ You are at least at an intermediate user level with your V-Synth. The [Parameter Tweak Map](#) presented at the end of this book will help you navigate the patch edit screens.
- ▶ You are familiar with the common concepts, terminology and abbreviations used in synthesizers and sound processors (oscillator, envelope, LPF, LFO, etc.).

Supplementary Materials



Each article in this book is supplied with example patches as well as audio demos of most these sounds. These materials are designated as  **Patch Name** in the text (with the number of a given patch specified for easier locating).

- ▶ Example patches are available in the *TweakBook.prf* project. You can transfer it to the V-Synth's card or user storage area and then load with [MODE] → *DISK* → *Load Project* (card area only) or import into empty patch slots in your own project using [MODE] → *DISK* → *Import Files*.
- ▶ Sound demos are in MP3 format and have the same names as patches.



Note that part of the examples require functionality present only in V-Synth OS 2.0 (like the step modulator and ROM waveforms)

Book Concepts

All along this book we will use patch structure diagrams to represent how a sound is formed using the V-Synth's building blocks - oscillators 1 and 2, modulator, COSM units 1 and 2 and the master TVA (read the next subsection for instructions on how to recreate patches from these structures). For reading convenience, here we further develop the non-exact¹ but already established naming approach of synthesis and also use colors to differentiate different synthesis element (device) types.

Color	Element	Description
	<i>Generator</i>	A device that generates audio signals. Example generators are oscillator, PCM sample player, external input signal.
	<i>Modifier</i>	A device that processes signals changing their amplitude, spectral and other characteristics. Typical Modifiers are amplifier, filter, booster, wave shaper.

¹ In general, synthesis blocks cannot be strictly differentiated. For example, a filter is a Modifier indeed, but when it self-oscillates it also becomes a Generator. Likewise, a Generator may be used as a Controller for other Generator or Modifier, and so on.

Color	Element	Description
	<i>Controller</i>	A device that generates one-shot or cycled control signals typically in the sub-audio frequency range. Example Controllers are envelope generator, LFO, step modulator.
	<i>Modulator</i>	A device that combines two signals in a certain way, or makes the properties of one device controlled by another device. Typical Modulators are ring modulator, frequency modulator, envelope modulator.

Note that internally in the V-Synth, *Generator* and *Modifier* blocks (e.g. oscillators and COSM units) contain *Controller* elements as well (namely up to three envelope generators and one LFO in each). Just to avoid clutter and complexity in the drawings, we will represent these blocks as whole.

Where appropriate, within a displayed element block, graphics like filter frequency response curves, oscillator and LFO forms and envelopes will be drawn.

How to Create Patches

In order to recreate a patch from a given structure diagram, at first, make sure to initialize (clear) the patch first. This can be done by touching [MENU ▼] in the top right corner of the V-Synth's screen and selecting *Patch Init*.

Then, set the parameters presented on the structure accordingly on the V-Synth, block by block. It is best to follow the signal hierarchy and start with *Generator* blocks, then proceed to *Modulator*, *Modifiers* and finally to *Controller* blocks, and hear how the sound builds up with each step. Again, the [Parameter Tweak Map](#) will help you find which screen/tab a given parameter is adjusted at.

VariCraze

Welcome to Elastic Audio

VariPhrase is a unique and simply amazing technology that allows you to manipulate the frequency and time characteristics of audio independently, making it elastic. The pitch, formant and playback speed of samples can be separately changed and modulated in realtime by controllers, envelopes, LFO generators and the step modulator. To understand VariPhrase in an essence and realize it's possibilities, it is convenient to look at it in two different domains: spectrum and time.

In the *spectral domain*, each melodic sound usually has two groups of frequency components: the base tone and the formant. Typically, such sounds are formed by two different objects, one of them being an oscillator (which defines the frequency of the base tone - the base frequency and it's harmonics), and the second one is a filter/resonator (which affects the base tone's harmonic content but also introduces non-harmonic oscillations which form the formant).

Below are a few examples of different sounds and which objects form the base tone and shape the formant.

Type of sound	Base tone	Formant frequencies
Human voice	Vocal folds	Chest, Throat, Mouth
Guitar, Violin	String	Body
Flute, Trumpet	Body	Body
Jaw harp	Metallic ribbon	Throat, Mouth
Subtractive synthesizer	Oscillator	Resonant filter

Formant is a very important property of the sound. For instance, two humans may sing the same note (generating the same base tone with vocal folds), but their obviously-different chests, throats, mouthes and other organs (serving as resonators) define their unique voices. Also, there may be two guitars with same strings and frets, but different body geometry and materials will give them different sound characters.

Analyzing a sample, VariPhrase designates which components represent the base pitch (and even base pitches of sequential notes in a monophonic phrase) and which stand for the formant frequencies², and allows you to vary and modulate them independently through the *Pitch* and *Formant* parameters, also not changing the time characteristics of this sample. With the *Robot Voice* function the pitches of all sequential notes in a phrase get aligned to the pitch of the key you play.

In the *time domain*, VariPhrase lets you play audio at any speed, reverse and even freeze it in realtime without touching the spectral domain (pitch and formant). It is also possible to play a sample in polyphonic legato mode, so that if you press a key while holding the previous key, the next sample doesn't restart but instead starts just where the previous was playing (allows for creating harmonies, realtime rephrasing, etc.). In addition, VariPhrase allows to trigger individual chops of a phrase in a sequence or lay them out on the keyboard like a kit.

Turning any sample into an elastic VariPhrase waveform simply requires you to encode it using

² Note that for purely harmonic signals like saw or square waveforms, VariPhrase doesn't allow to change the formant because there aren't any formant frequencies present in them.

one of the four algorithms which are especially designed for different types of source sounds and how the actual encoded samples will be used. These algorithms are *Lite*, *Solo*, *Backing* and *Ensemble*, and below is a table giving you an overview of them.

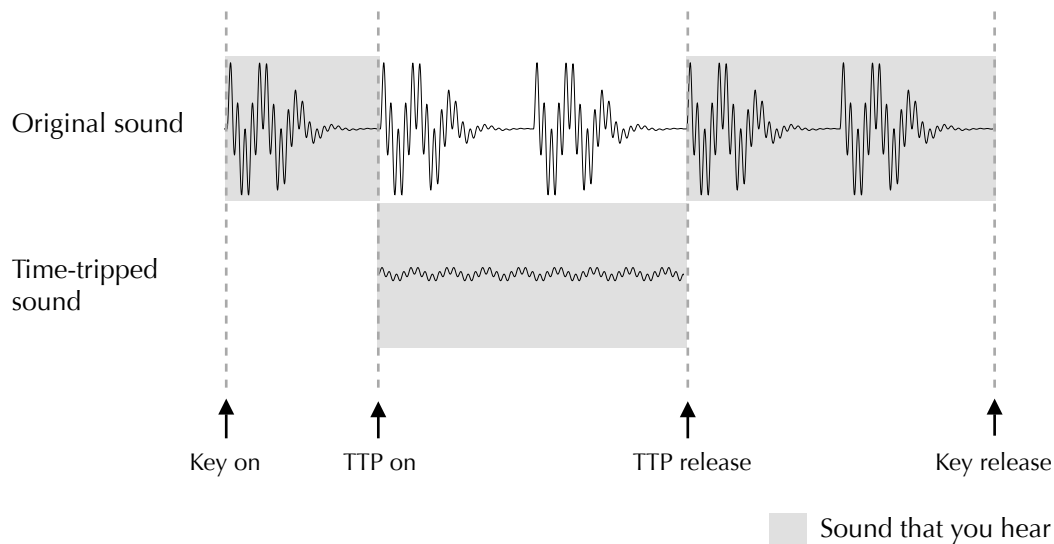
Encoding	Applications
<i>Lite</i>	This encoding type is intended for samples that will be used with VariPhrase off. Do not use this encoding for serious VariPhrase processing.
<i>Solo</i>	Special algorithm for monophonic audio samples only, with all VariPhrase features available.
<i>Backing</i>	Intended for non-pitched sounds such as drums, effects, etc. <i>Robot Voice</i> and <i>Formant</i> controls not available.
<i>Ensemble</i>	Algorithm for polyphonic phrases and otherwise multi-pitched sounds like instrument+drum loops and such. <i>Robot Voice</i> and <i>Formant</i> controls not available.

Summing everything discussed above up, here are just a few examples of how VariPhrase can be used:

- ▶ You can stretch the pitch of a sample across several octaves without altering it's character and speed.
- ▶ By modifying the formant frequencies of a sample, you can alter it's distinct character - for instance, change the gender of the human voice or the body type of a guitar.
- ▶ Legato playback allows for "re-phrasing" monophonic phrase samples and creating polyphonic harmonies.
- ▶ Using the realtime controllers you can manipulate sample's time like if you controlled the playback of a vinyl disc or a tape with your hand, yet without altering sound's pitch and character.
- ▶ You can freeze a sample at any point and turn it into an eternal waveform. For example, it's possible to make a choir sound out of a vocal phrase, or extract a single guitar note out of a phrase to create a guitar patch.
- ▶ Using *Robot Voice*, you can freeze the pitch of a sample, so that all notes get "aligned" to the note you play from the keyboard. For instance, if you have a vocal phrase singing notes C4, F4, G4, they all will shift to whatever key you press (i.e. if it's the A3 key, all of these notes will shift to A3). This allows you to create harmonies and "paraphrase" samples.
- ▶ It's possible to tempo-synchronize samples very easily and with much less artifacts compared to traditional time stretching.

Beat Control

Probably the most simple yet fun application of VariPhrase is controlling the playback speed and direction of a sample using the TimeTrip Pad. And there's a great feature hidden behind the *Time Trip Beat Keep* switch which allows a tempo-synchronized loop to stay in sync with the tempo even if you travel though it's timeline. For better understanding of how this works, examine the figure below.



Beat keeping at work

When you start working with the TimeTrip Pad, the V-Synth moves the virtual playhead along the sample waveform according to the movements of your fingers. However, it keeps the timeline of the original sample running, so that when you release your finger off the TTP, you hear the sample playing back from the same position as if you did not touch the TTP.

What this allows for are various freezes, scratches and breaks which are non-destructive to the timeline of the sample: the loop will always be in sync with the beat. **084 Scratch Me**

You can use this functionality to create unique realtime performances:

- ▶ Set up two different tempo-synchronized loops on oscillator 1 and 2, and switch *TimeTrip Beat Keep* on only for one of them. This way you'll be able to scratch and freeze one loop while the other is playing, and they will always be in sync.
- ▶ Create a patch with one oscillator running a loop, and another having a bassline (for example, made using the step modulator). Again, you can time-travel with the loop and it will always be in sync with the bassline. **083 Storm Break**

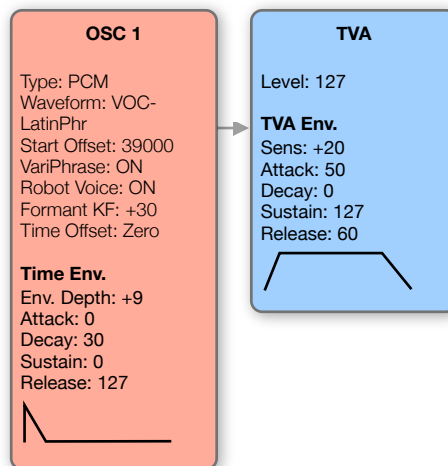
Tame the Time

You have almost surely played enough with freezing the samples' time using the realtime controllers already, but among other things the V-Synth also allows you to take complete control over a PCM sample's time using envelopes and LFOs. One of the many applications of this is taking a portion of a sample, freezing it's pitch and time and turning into a continuous tone. For instance, you can turn a vocal phrase into a choir sound or a guitar lick into a single string pluck.

The core tricks for achieving this are as follows. First you select a VariPhrased waveform of a monophonic sound, preferably with Solo encoding. With the *Offset* parameter you can set the start point of the sample playback and thus select the portion of it you'd like to use. Freeze the pitch with *Robot Voice* to make sure the PCM oscillator's pitch is bound exactly to the keys you play. Now, freeze the time by setting *Time Offset* to "Zero". And then, set a time envelope to play the sample a bit and then slow down and freeze. Thus, you won't just loop the very start of the sound but instead, for instance, a vocal phrase like "laah" will turn into an eternal "laaaaaaa".

Below is an example vox patch made out of a male vocal phrase. Try changing the *Start Offset*. to freeze different syllables of the vocal phrase. Adjust the the Formant parameter to change the

character of this chorus sound to make it sound like a lower male, a female, or a child-like voice. Note that a *Hexa Chorus* MFX unit is also used in this example patch to multiply the sound and make it sound like a lush chorus. **004 Frozen Vox**



Time freezing opens up doors to a new universe

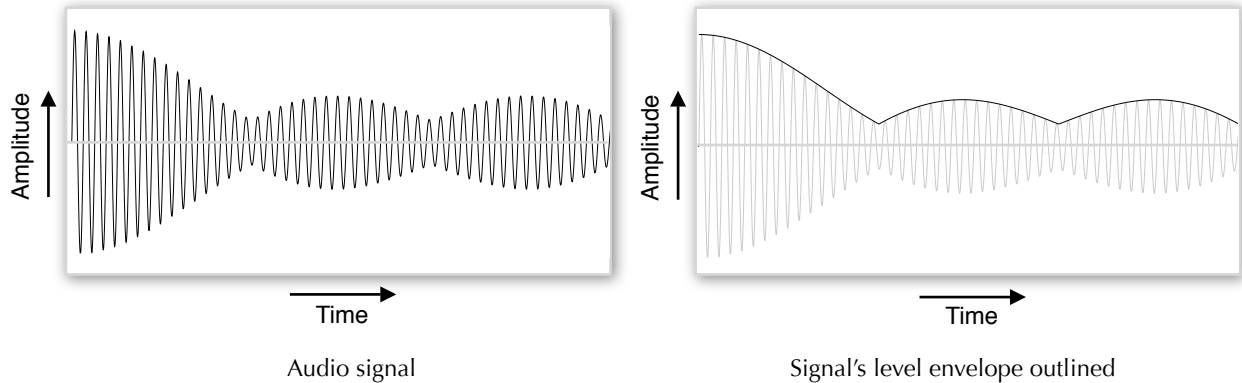
You can apply this technique to other source samples, and with some more tricks:

- ▶ Extract and freeze a guitar pluck from a guitar lick sample. Set OSC1 LFO to a *Random* waveform, *Rate* of 70-80 and *Time LFO Depth* of +1..3. This will make the sample play and randomly play a little further or backwards from the note you extracted, also playing parts of adjacent notes in the sample. A resulting sound can be a very interesting and lively guitar patch. **053 Guitar Alive**
- ▶ If you use spectrally rich, atonal samples with Ensemble or Backing encoding, time freezing will cause many artifacts which may not only sound like buzzing, metallic but also more soft sounds. In certain cases, you can turn these into very interesting new creations: in this following example, a special effect sample is turned into a lush bell sound. **082 Gamelan Bell**

Modulation Exploration

Envelope Modulator

If you look at some sound's waveform (say, in a sample editor), you can see that it's level changes in time, and you can put an imaginary line going above the waveform peaks. These changes are called a signal's level envelope:



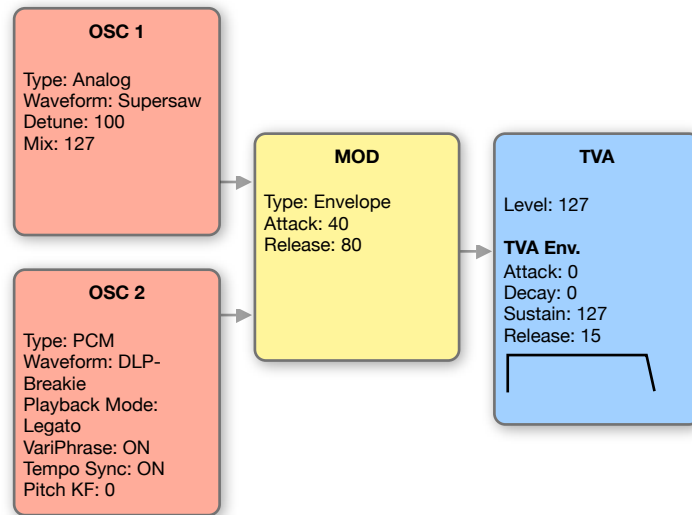
Example sound and it's envelope

The *Envelope Ring* modulator in the V-Synth is what is called an “envelope detector” - in simple words, it “extracts” the envelope of the signal level coming out of OSC2, and lets you apply this envelope onto OSC1. The *Attack* and *Release* parameters of this modulator allow to define how fast it reacts to the changes in the incoming signal level. Higher values for *Attack* make the sound smoother, lower values make it more percussive. Increasing the *Release* parameter makes the envelope sustain longer before each next envelope peak.

The V-Synth's envelope modulator is typically used to apply a drum loop's envelope onto another sound which makes it's amplitude change rhythmically. To try this out, create a patch presented on the next diagram. 022 MotionTrance

Following this example, note that:

- ▶ The drum loop is tempo-synchronized. This allows you to have the step modulator or LFOs be in sync with the final sound's pulsations.
- ▶ The *Playback Mode* of the PCM oscillator is set to legato which lets you go from one note or chord to another smoothly without re-triggering the drum loop.
- ▶ *Pitch KF* (keyfollow) for the PCM oscillator is off (set to 0) to eliminate the slight loss of sound's percussiveness as VariPhrase processes it ± 2 octaves from it's original pitch.



Making motion sounds with the envelope modulator

To get more advanced with this, use the following tips:

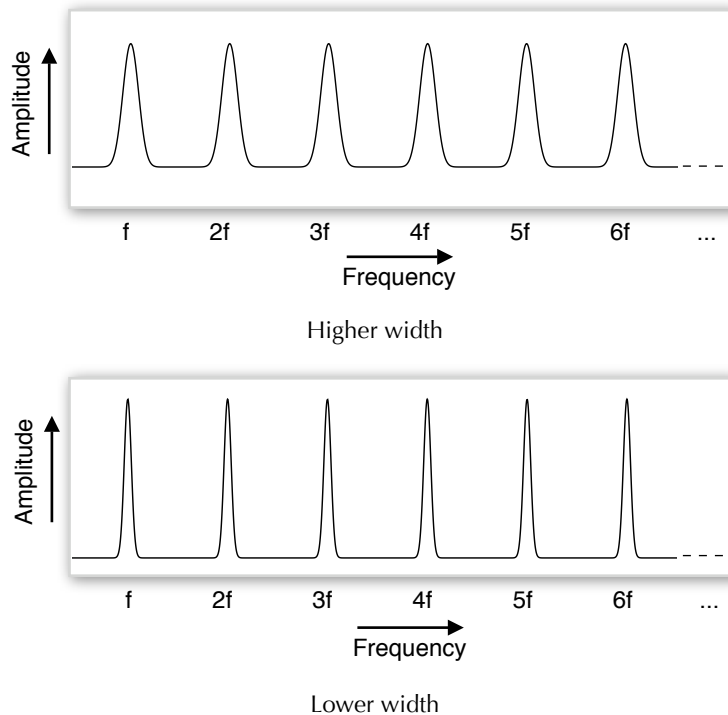
- ▶ Use a TVF COSM block (low-, band- or high-pass) to define which frequency components of OSC2 affect OSC1. Structure 3 allows you to put COSM2 after OSC2 before it gets to the MOD block. **036 Wave Coder**
- ▶ You can use a frozen vocal phrase as the base sound (carrier) to create very interesting rhythmic vox sounds. **065 Vox Motion**
- ▶ Add a filter after OSC1 and modulate it with a step-modulator - it will add some more rhythmical sound changes and again will be played legato.

COSMology

Side-Band Filter

Side-band filter (SBF) passes through only those frequencies which are the harmonics (multiples) of its base frequency³ (f) and attenuates all non-harmonic partials. Its frequency response basically corresponds to a set of very narrow band-pass filters each positioned at the multiples of the base frequency ($f, 2f, 3f, 4f\dots$). The core parameter of the SBF is width: the lower it is, the more of the non-harmonic partials get filtered out, and the more narrow the individual bands become.

Such type of a filter makes it possible to turn any non-pitched signal such as noise or a drum/percussion loop into a musical tone while pertaining its amplitude characteristics. Below are graphical representations of side-band filter's frequency response.



Side-band filter frequency response

In the V-Synth, two side-band filters are available: 1st and 2nd order, with the latter having narrower pass bands than the former, thus producing a more deep effect. The base frequency of both of the side-band filters is tied to OSC1 and doesn't react to OSC2. This makes possible to mute OSC1 and control the side-band filter's base frequency using the settings from OSC1 while not changing the pitch of OSC2.

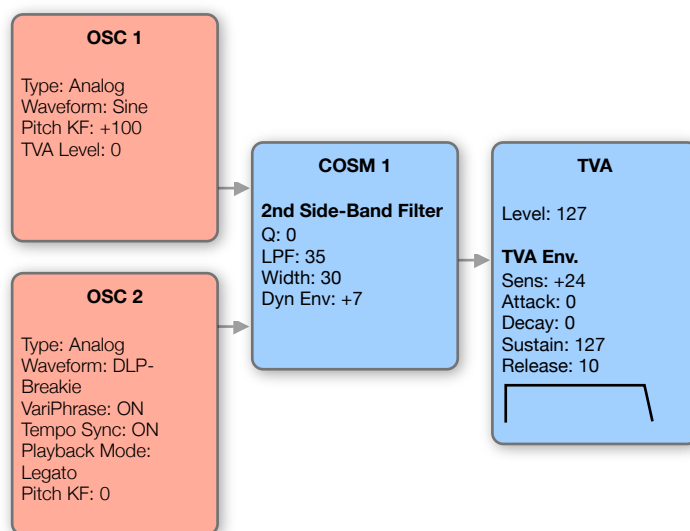
The Q parameter of the 2nd-order SBF, when decreased, allows for greater filtering. There is a built-in low-pass pre-filter with the LPF parameter controlling its cutoff frequency, lower value will give a softer sound and higher values will make it brighter. The $Dyn Env$ parameter makes the SBF's *Width* modulated by the incoming signal's level - this lets you get more percussive sounds, as the attack/peaking portions will have less filtering from the side-band filter. *Detune* also lets you slightly offset the base frequency of the SBF against OSC1's pitch.

To begin learning the V-Synth's SBF, route OSC1 with a noise waveform into a COSM block

³ It is better to call it a *base frequency* rather than a *cutoff frequency* simply because the latter typically means there is one band gap or peak in the filter's frequency response while the side-band filter has many.

with SBF1 or SBF2, then start turning the *Width* parameter of the side-band filter down and see how the noise becomes a more and more mellow melodic sound. With a simple falling TVA envelope for OSC1, you can create string-like sounds. To make the sound more like a real plucked or blown instrument, make the decay portion of it have more inharmonic partials by applying a falling envelope for SBF *Width*. **006 SBF Koto** **025 Star Sitar**

For a drum loop, play with the *Q*, *LPF*, *Width* and *Dyn Env* parameters to get the feel of the SBF's response. Just as on the patch depicted below, you can also do a little trick to eliminate the loss of sound's percussiveness due to the drum loop being pitch-shifted with VariPhrase: use OSC2 for the PCM loop, but a silenced OSC1 for the actual pitch control. **013 SBFuture**



Turn any drum loop into a metallic-sounding melodic texture

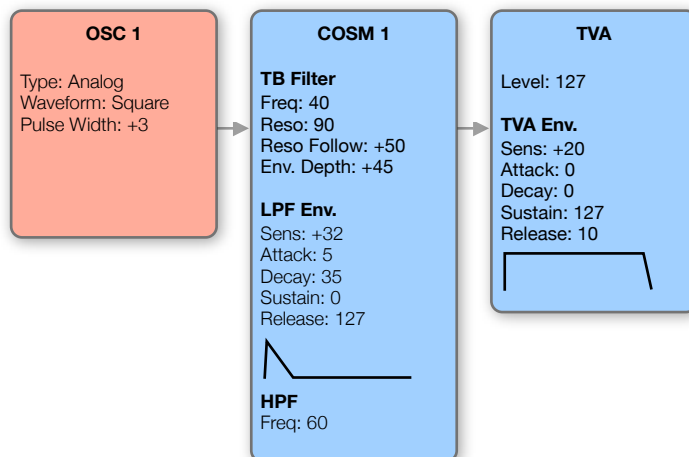
TB Filter

It shouldn't be a secret to you that the original -18 dB/octave resonant low-pass filter developed for the TB-303 Bassline synthesizer made in early 1980's by Roland is fairly considered to be among the most great-sounding filters ever. This magic device possesses a unique and easily recognizable character, and it's fantastic to have it's great sound faithfully modeled in a modern Roland's breakthrough - the V-Synth, that is.

But in order to closely emulate the original TB-303 sound, you have to take into account some of it's other unique properties and recreate them on the V-Synth accordingly:

- ▶ TB-303's envelope modulates not only the filter cutoff but also the resonance, and on the V-Synth's TB filter the *Reso Follow* parameter recreates this.
- ▶ The envelope has a slight attack which gives the sound a little more squelchiness. On the V-Synth this corresponds to LPF envelope *Attack* value of around 5..8.
- ▶ TB-303 has an "Accent" parameter which increased filter cutoff and resonance modulation depth by the envelope generator. On the V-Synth you should set up the filter envelope *Sens* to around +24..32.
- ▶ The saw oscillator is pretty much a generic saw (V-Synth's saw works just fine), but the TB's square waveform has a slightly wider pulse width than a perfect square which corresponds to a *Pulse Width* of about 3..4 on the V-Synth.
- ▶ The bottom lows of the V-Synth's oscillators should be removed to make it sound closer to the TB-303. This can be done by setting the HPF's *Cutoff* to 60-70.

To create a simple square-based TB-303 acid sound, follow the structure below.



Never enough of those juicy 303 sounds

Just switch the patch to mono mode, then turn on legato, portamento (with a *Time* setting of about 10..15), set portamento mode to legato, and you are ready to create very convincing TB-303 basslines albeit with accents and slides. 047 Classic TB

Parameter Tweak Map

Oscillator

Tab	Description	Most significant parameters
OSC Type	Oscillator type and general settings	Waveform, Wave Gain, Impact, Sub-oscillator range/off, Sub-oscillator level (<i>Analog</i>) Waveform, Wave Gain, Start Offset, Robot Voice, Playback Mode (<i>PCM</i>)
Pitch	Pitch and pitch envelope settings	Pitch, Coarse tune, Fine tune, Pitch KF, Pitch LFO Depth, envelope: Curve, Sens, Depth, Attack, Decay, Release, Sustain
Pulse Width (<i>Analog</i>)	Pulse width (Detune) and pulse width envelope settings	Pulse Width, Pulse Width KF, Pulse Width LFO Depth, envelope: Curve, Sens, Depth, Attack, Decay, Release, Sustain
Fat/Detune (<i>Analog</i>)	Fat/Detune and Fat/Detune envelope settings	Fat/Detune, Fat/Detune KF, Fat/Detune LFO Depth, envelope: Curve, Sens, Depth, Attack, Decay, Release, Sustain
Time (<i>PCM</i>)	Time and time envelope settings	Time, Time KF, Time LFO Depth, envelope: Curve, Sens, Depth, Attack, Decay, Release, Sustain
Formant (<i>PCM</i>)	Formant and formant envelope settings	Formant, Formant KF, Formant LFO Depth, envelope: Curve, Sens, Depth, Attack, Decay, Release, Sustain
TVA	Oscillator amplifier and amplifier envelope settings	Level, Level LFO Depth, Pan LFO Depth, envelope: Sens, Depth, Attack, Decay, Release, Sustain
LFO	LFO settings	Waveform, Rate, Key Sync, Offset

COSM

Tab	Description	Most significant parameters
Cutoff (TVF, Dynamic TVF)	Filter cutoff frequency and cutoff envelope settings	Freq, Freq KF, LFO Depth, Env. Depth, envelope: Curve, Sens, Attack, Decay, Release, Sustain
Reso (TVF, Dynamic TVF)	Filter resonance and resonance envelope settings	Reso, Reso KF, LFO Depth, Env. Depth, envelope: Sens, Attack, Decay, Release, Sustain

Tab	Description	Most significant parameters
LPF (TB Filter)	Filter/resonance and filter/resonance envelope settings	Freq, Freq KF, Reso, Reso Follow, Freq LFO Depth, Envelope Depth, envelope: Curve, Sens, Attack, Decay, Release, Sustain
HPF (TB Filter)	High-pass filter and envelope settings	Freq, Freq KF, LFO Depth, Env. Depth, envelope: Curve, Sens, Attack, Decay, Release, Sustain
Width (SBF1/2)	Side-band filter width and width envelope settings	Width, Width KF, Dyn Env, Width LFO Depth, envelope: Sens, Envelope Depth, Attack, Decay, Release, Sustain
Detune (SBF1/2)	Side-band filter detune and detune envelope settings	Detune, Detune KF, Detune LFO Depth, envelope: Sens, Envelope Depth, Attack, Decay, Release, Sustain
LFO	LFO settings	Waveform, Rate, Key Sync, Offset

TVA

Tab	Description	Most significant parameters
TVA	Amplifier and amplifier envelope settings	Level, Level LFO Depth, Pan LFO Depth, envelope: Sens, Depth, Attack, Decay, Release, Sustain
LFO	LFO settings	Waveform, Rate, Key Sync, Offset

Credits

Acknowledgments

I would like to thank my wife Victoria and daughter Miroslava for giving me bright days of happiness which greatly inspire to create.

Kudos and sincere respect to Roland Corporation (especially the whole MI team) for their ever-flowing innovations and fantastic instruments.

Special thanks to (in strict alphabetic order):

- ▶ Vyacheslav Cherkashin
- ▶ Apple Computer
- ▶ Audiofile Engineering
- ▶ Igor and Olga Pavlovs
- ▶ Stephane Pigeon
- ▶ Gordon Reid
- ▶ Hans-Jörg Sheffler
- ▶ Howard Scarr
- ▶ Nathan Sheldon
- ▶ Roland Ukraine

Book Information

The V-Synth TweakBook has been created on Mac OS X Tiger using the following software:

- ▶ *Pages* for drawings and document composition
- ▶ *Grapher* for waveform graphics
- ▶ *Wave Editor* for sample editing and analysis

