MIDIdocs

The V4L as a MIDI Processor

Link to Forum Discussion

The V4L can also be used to process incoming midi signals. All incoming MIDI can be forced-to-scale, and effects (like delays and LFOs) can be applied to selected channels. Since firmware version 85, with a configuration change, the processed notes can go out on the same channel they came in on.

Now would be a good time to flash your V4L with the latest firmware, using MIOS Studio. After flashing the firmware, enter the “store” command in MIOS Studio's command line. This will write all of the new parameters you'll be using in the configuration files on your V4L, so you can edit them as set out below.

To enable force-to-scale on incoming MIDI notes

1. Open the MIOS32 File browser built into MIOS Studio.
2. Edit the SESSIONS/DEF_V4L/MBSEQ_C.V4 file.
3. Find “LiveForceToScale”, and set it to 1. If it doesn't exist, add it and set it to 1. But it should exist, because you used the “store” command, right?

When In–>Out is enabled on the V4L, incoming MIDI notes will now be forced to the scale set on the V4L's force to scale setting for the currently selected V4L sequence (set the scale and root note using the V4L SCALE button). If you choose different force-to-scale settings for the two V4L sequences, you can do quick key changes on the incoming midi by simply switching between the two sequences on the V4L using the SEQ1 and SEQ2 buttons.

All incoming MIDI notes on channels 1-4 will be merged together and sent out on the MIDI channel of the currently selected V4L sequence. In this configuration, the V4L only forwards MIDI notes received on channels 1-4.

Preserving the Original Channel

Well, that's all very nice, but what if you don't want the notes to all be merged onto the same MIDI channel? What if you want the V4L to forward notes received on all 16 MIDI channels, and you want each note forced-to-scale and then sent out on the same channel it was received on?

Never fear! Since firmware version 85, TK has you covered.

1. Open the MIOS32 File browser built into MIOS Studio.
2. Edit the SESSIONS/DEF_V4L/MBSEQ_C.V4 file.
3. Find “MIDI_BUS_Channel 0”, and set it to 17 - so: “MIDI_BUS_Channel 0 17”. This will make BUS 0 of the V4L listen for incoming midi on all channels.
4. Set “MIDI_BUS_Channel 1”, “MIDI_BUS_Channel 2” and “MIDI_BUS_Channel 3” to 0. We don't need these busses to listen to channels anymore, since BUS 0 is listening on all channels.
5. Find “LiveKeepChannel” and set it to 1.
6. Save the MBSEQ_C.V4 file. These parameters should be in effect immediately once you save the file.

You're done.

You'll notice that the V4L doesn't just force the incoming MIDI to scale. It also can apply the ECHO, HUMANIZE, and LFO effects to incoming MIDI notes - so your V4L is now a MIDI effects processor. The V4L will apply effects only to the MIDI channels that are selected for SEQ1 and SEQ2. Some of the effects won't work if the V4L playback is stopped.

You can also record and play back sequences on the V4L while it is processing and forwarding incoming MIDI.

From:
https://www.midibox.org/dokuwiki/ - **MIDIbox**

Permanent link:
https://www.midibox.org/dokuwiki/doku.php?id=mididocs:v4l:midiprocessing&rev=1410747837

Last update: **2014/09/15 02:23**