DePitcher

Build #3, 2025-03-20

This module is intended to simulate inaccurately tuned voices of ancient analog synthesizers and it's long time frequency drifting as a result of thermal influences.

Features:

- 1 to 8 voices in mono, poly or unison mode
- controllable by monophonic or polyphonic pitch/gate signals
- using both monophonic and polyphonic outputs at same time
- voice assignment modes either "walk" or "smart"
- fixed or randomized detune up to +/- 100 cent (equally distributed to all voices)
- initial and final main detune and keyboard rate failure
- drifting time interval 1 to 9999 seconds
- start, restart and finish controllable by push buttons and CV

Table of contents

1 Function

1.1 Basics

Basically **DePitcher** is a CV processor. One could also name it “CV-Effect”, because it receives, processes and then transmits pitch and gate signals.

A block diagram at the end of this document may help you getting familiar with **DePitcher** module.

1.2 Signal inputs

DePitcher is equipped with both mono and poly input pairs (***pitch*** + ***gate***), but depending on ***M-P*** slider switch either mono or poly inputs will be used internally.

1.3 Voice assignment

Music synthesizers contain a distinct number of voices. Each voice can consist of one or more main VCOs, LFOs, VCFs and envelope generators. Ancient analog synthesizers created their sound with one, two, six, eight, 16 or more voices. Commonly each voice was built on a printed circuit board. Synth controllers had to manage signal flow from keyboard to the voices. Therefore different principles have been used. Two of these are:

- “***Walking*** mode”: Each time a key is pushed down, main controller tries to use next voice. If it is still in use by a held key, controller will look around in order to find an unused voice. If there isn’t any free voice, controller will use succeeding voice as planned before. So that still held note will be “killed” by a new one. Even if only one key is pressed down at a time, controller will use succeeding voice for each new starting note. This makes it possible playing several single notes on poly synthesizers, each with very long sounding sustain.
- “***Smart*** mode” works a bit smarter. First controller looks for earliest formerly played note among unused voices. If there isn’t any voice unused, active voice with earliest played note will be selected to play actual note.

Because in MONO mode mono outputs anyway just provide last played note, a real advantage of “smart mode” is only hear-able with poly output signals in **DePitcher**’s POLY mode.

In order to provide another variant ***SMART*** assignment mode is created as being more likely stupid in ***MONO*** input mode. In this case only virtual voice #1 is used

and gets sent to mono outputs as well as to poly output channel #1. Only in UNISON mode whole number of “voices” is used as set with synth voices knob.

Assignment mode also effects behavior of mono outputs. (more in 1.6.1.2)

1.4 Voice transforming

Though the module processes either mono or poly input signals, it can provide both monophonic and polyphonic signals at same time. Host polyphony may be equal to or differ from actually preset **DePitcher**’s voice number. Depending on input signals, as well as on polyphony and current module preset, only one or a distinct number of poly channels will provide pitch and gate values.

DePitcher is able to transform a distinct number *P* of input signals (representing polyphony) to another number *V* of output signals (representing ***synth***esizer ***voices***).

Case *P* < *V*:

> **DePitcher** will use every internal synth voice channel. But because Voltage poly connectors transmit only as much channels as predefined with polyphony, poly outputs will not provide signals of all voices. So all notes of synth channels above *I* will get missed at the poly outputs. Therefore the module is equipped with eight pairs of mono output jacks too.

Case *P* = *V:*

> There are as much input channels as output channels. **DePitcher** will assign each input note to one of the internal voice channels. No note will get lost.

Case *P* > *V*:

> All input signals will get distributed to a synth voice channel. Depending on number of played notes at same time and of selected assignment mode, some notes may get terminated by other notes.

1.5 Pitch processing

1.5.1 Fixed detune

Fixed detune of several oscillators is very popular together with unison in order to create fat sounds. **DePitcher** offers two variants:

- In “regular” mode ***detune*** value will be equally distributed to all synthesizer voices, except voice #1.

Example 1:

- ***synth voices***: 2
- ***fixed detune***: 12 ct (1 cent = 1/100 half tone)

resulting detune: voice 1 0 ct
voice 2 +12 ct

Example 2:

- ***synth voices***: 5
- ***fixed detune***: 12 ct

resulting detune: voice #1 0 ct
voice #2 +3 ct
voice #3 -6 ct
voice #4 +9 ct
voice #5 -12 ct

- In ***RANDOM*** mode all voices (excepted #1) get detuned by a value between null and preset detune value, when ***RNDMZ*** (= randomize) button is pushed. Value sign will be set randomly too.

1.5.2 Pitch drifting

Ancient analog synthesizers did have a massive problem: their pitch was not even constant. After switching power on, internal device temperature increased until it became rather constant. Semiconductor's characteristics are very sensitive against temperature changes. So devise tuning drifted until device interior reached a stable temperature.

That pitch drift manifests itself in two ways:

First, absolute pitch of the whole instrument was a little bit higher or lower than it should be. We call this pitch offset. To correct that offset the “Master Tune” knob can be used, which is commonly part of analog music synthesizer’s panel.

Second, keyboard factor could change. What does this mean? Normally tone frequency should get exactly doubled when a key is pressed just one octave higher. Resulting from inadequate adjustment or varying temperature this factor may differ from 2.

DePitcher tries to simulate both detuning effects, influenced by temperature changes.

1.5.3 Timer

DePitcher's timer can be set to an interval length from one to 9,999 seconds. After timer was started, output pitch values will get modified by an offset value. This value varies from ***initial offset*** at the beginning to ***final offset*** at the end.

1.6 Signal outputs

1.6.1 Mono outputs

1.6.1.1 Mono outputs in mono input mode

Mono output jacks deliver values from mono input jacks. But **DePitcher** offers a special feature: It can remember played notes. Even in mono input mode held keys stay in memory while other keys are played and hold. When a key gets released, it's pitch gets replaced by the pitch of it's preceding note (still held key). Number of memorized notes does not depend neither on polyphony nor on preset number of synth voices.

Basically it's the same with mono gate out signal. But **DePitcher** has another feature. Before we will talk about this feature, we must have a look at *Cherry Audio's Voltage Modular*.

On it's *CV SOURCES* panel there is a push button, which can be toggled between *SINGLE* and *MULTI*. When *MULTI* is on, *TRIG* output provides a trigger pulse each time a note is played, even when played legato. When toggle button is on *SINGLE*, no further trigger pulses will appear until all keys get released at least for a short moment.

GATE output beside *TRIG* out jack works similar. Regularly a gate signal appears any time a key is pushed down. In *SINGLE* mode *GATE* would stay ON, while at least one key is held down. This would prevent from restarting envelope generator for newly played notes. Therefore commonly MULTI is preferred mode. In *MULTI* mode *GATE* gets interrupted for a very short moment at same time as mentioned *TRIG* output is ON.

Let's get back to **DePitcher** and it's next feature. It's mono ***gate out***put can work in two different ways:

- With ***LEGATO*** button toggled off, ***gate out***put provides single gate pulses, each time a note is played, even when notes are played as legato and *Voltage SINGLE* button is on. There will also be gate pulses, when keys get released, but not when very last note ends.

- When ***LEGATO*** is toggled ON, mono gate output stays ON even when ***gate input*** receives shortly interrupted signals because *MULTI* is selected in *Voltage Modular*.

1.6.1.2 Mono outputs in poly input mode

Behavior of mono pitch output in poly input mode can be set to one of these modes:

- LAST: Pitch output provides pitch value of last played note. If there are several notes played at same time, pitch value with highest channel number gets processed as last played note.

- LOWEST: Lowest pitch value of all active voice channels gets sent to mono pitch output. If actually lowest note ends, **DePitcher** searches for lowest note again.

  Voice assignment mode also influences LOWEST mode:

  + In WALK mode every active note gets processed same way. So it may happen, that previously lowest note gets terminated by a new note. In this case **DePitcher** searches for lowest note again.

  + In SMART mode lowest notes get processed privileged. If all voice channels are already active, lowest note will be prevented from getting terminated by a new note. That simplifies playing live, because bass note from mono output can stay on all time while playing accords.

1.6.2.1 Poly outputs in mono input mode

Basically only output channel #1 provides signals, though number of voices can be higher than one. Depending on assignment mode, **DePitcher** uses voice after voice. Only signal from last played note from last assigned voice will be sent to poly output channel #1. Only exception is with UNISON. In this case as many output channels are active as number of voices is preset.

1.6.2.2 Poly outputs in poly input mode

Polyphony sets and also limits number of usable output channels. When DePitcher works with more voice channels, all channels higher than polyphony will stay silent. Independent of that limitation extra mono outputs provide signals from all voice channels.

2 Controls and connectors

Connect cables for pitch and gate to these sockets. You can use both mono and poly connectors at same time.

Module will only process signals matching switch position:

- M mono inputs
- P poly inputs

Use this knob to set number of synthesizer voices, DePitcher should emulate.

Range: 1 to 8 (green display)

When set to 0, number of voices is equal to polyphony, but limited to 8. (red display)

These square fields show actual state of every voice:

- light gray: unused
- dark gray: used, gate off
- cyan: used, gate on

With this slider switch you can chose voice assignment mode.

Here you can set length of emulation interval.

Range: 1 to 9,999 seconds (max 2h 46min 39s)

Red display shows actual duration of emulation.

When a trigger pulse is received or the button is pushed, emulation process starts or restarts with initial settings.

As long as emulation is running, this socket delivers a +5 volt gate signal.

With this knob one can detune synth voices in relation to voice #1 as described in chapter 1.5.1.

Range: -100 to +100 cent (1 ct = 1/100 half tone)

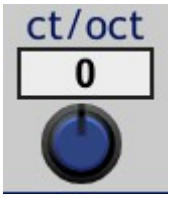
If you want to detune voices more naturally, first set maximal detune value. Then set RANDOM toggle button on. Finally click on the RNDMZ (= randomize) button as often as you want until you are satisfied with the result.

When you want DePitcher to go back to initial emulation state, click on this knob or send a trigger pulse to the input connector beside.

When you want DePitcher to go forwards to final emulation state, click on this knob or send a trigger pulse to the input connector beside.

Here you can adjust a common pitch offset for all voices.

Range: -1,200 to +1,200 ct (-1 to +1 oct)



This knob lets you adjust the keyboard scale factor fault.

Range: -100 to +100 ct/octave (-1 to +1 halftone/oct)

This output sends a pitch CV according **DePitcher** modes:

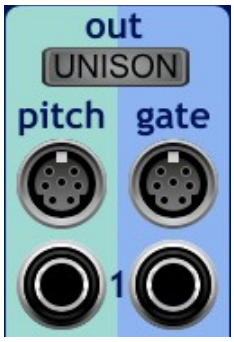
- MONO + WALK: step by step changing individual pitch value for every voice
- MONO + SMART: pitch value of voice #1
- POLY + LAST: pitch value of voice with last played note
- POLY + LOWEST: pitch value of actually lowest tone playing voice

Here **DePitcher** offers a gate signal according modes:

- MONO + /LEGATO: A (shortly interrupted) gate is provided each time when a note begins or ends or when input pitch changes rapidly from one sample to another one
- POLY + /LEGATO + LAST: Gate goes on when a single note begins. It gets re-triggered with every new note and stays ON as long as last played note is active.
- POLY + /LEGATO + LOWEST: Gate goes on when a single note begins. It gets re-triggered with every new note and stays ON as long as lowest note is active.

- MONO + LEGATO: Gate goes on when a single note begins and stays ON as long as at least one note is active
- POLY + LEGATO + LAST: Gate goes on when a single note begins. It stays ON without interruptions as long as last played note is active.
- POLY + LEGATO + LOWEST: Gate goes on when a single note begins. It stays ON without interruptions as long as one note is active.

This toggle button activates gates of all available voices.



On module front right side there are poly outputs as well as mono output pairs for each voice. Latter connectors allow to overcome Voltage channel limitation because of a lower polyphony.

When UNISON button is toggled on, each new note activates all voices. LEGATO button lets you play all voices legato. Positions of LAST-LOWEST or MONO-POLY switches do not have any effect for poly outputs.

3 Block diagram