

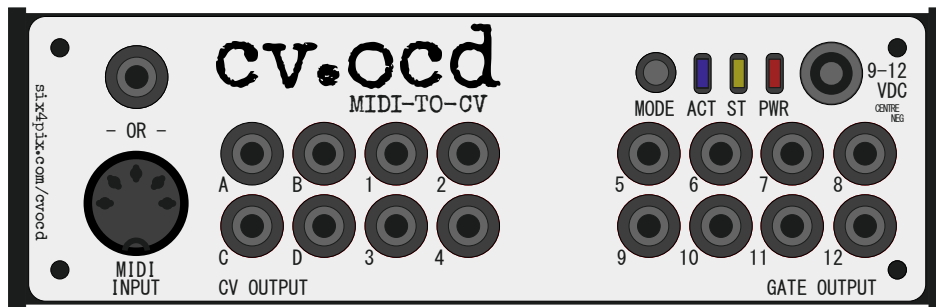
CV.OCD USER MANUAL

Introduction

CV.OCD is a device which converts MIDI signals to the Control Voltage (CV) and Trigger/Gate signals which are used to control traditional analog synthesizers.

As well as doing this useful bit of “plumbing”, CV.OCD is designed to open up some creative options that go beyond what a simple MIDI-CV converter can usually do. If you get a bit OCD about your CV this could be the box for you!

CV.OCD has four assignable “continuous” analog CV outputs and twelve assignable gate/trigger outputs.

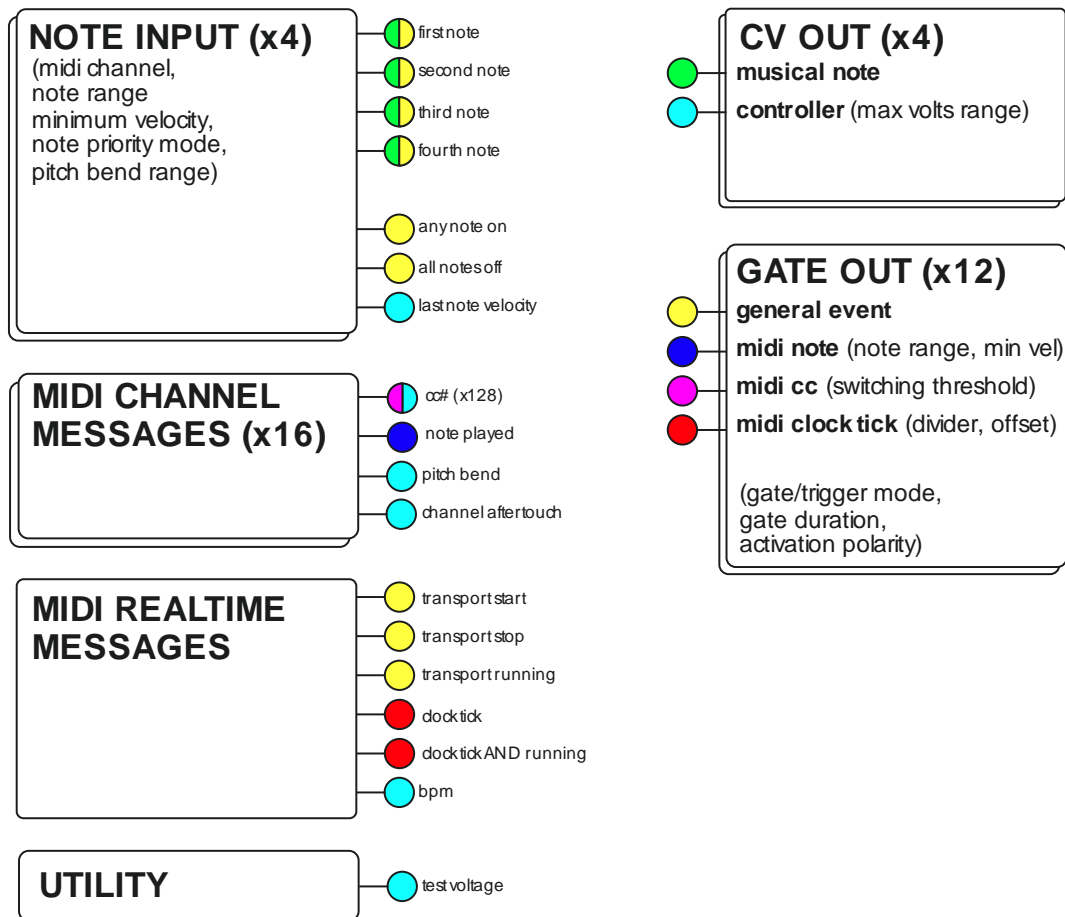


Straight out of the box, the outputs are assigned as follows

CVA <i>Channel 1 note pitch</i>	CVB <i>Channel 1 note velocity</i>	GATE1 <i>Channel 1 note gate</i>	GATE2 <i>Channel 1 note trigger</i>
CVC <i>Channel 2 note pitch</i>	CVD <i>Channel 2 note velocity</i>	GATE3 <i>Channel 2 note gate</i>	GATE4 <i>Channel 2 note trigger</i>

GATE5 <i>Trigger: Chan 10, note C4</i>	GATE6 <i>Trigger: Chan 10, note D4</i>	GATE7 <i>Trigger: Chan 10, note E4</i>	GATE8 <i>Trigger: Chan 10, note F4</i>
GATE9 <i>Trigger: Chan 10, velocity >= 100</i>	GATE10 <i>Clock: 4th notes</i>	GATE11 <i>Clock: 8th notes (e.g. Volca)</i>	GATE12 <i>Clock: 16th Notes</i>

These are just to get you started (and they should work well with the default synth and drum mappings of Novation’s Circuit sequencer / groove box) however there are many ways to assign MIDI notes and events to these outputs as shown in the diagram below...



The MIDI sources that are available for mapping are shown on the left hand side of the picture, and CV.OCD's outputs are shown on the right side.

The little round coloured "flags" indicate an input or an output and the colour indicate the type of information. Provided that the colour of an output flag matches one of the colours on a source flag, you can map the signal from the source to the output.

The text in brackets beside an output flag shows what additional configuration parameters (such as a note number) can be provided when you set up this mapping.

For example, a MIDI note (blue flag) can be directly mapped to a gate output, and that MIDI pitch bend (cyan flag) can be mapped to a CV output. A MIDI continuous controller (CC#) can be mapped to either a CV output as a controller (cyan flag) or to a gate output (magenta flag)

The Hardware

Connectors

CV.OCD has the following connections

- A socket for a 2.1mm barrel connector with a negative pin (as commonly used for guitar effect pedals). This socket accepts a DC voltage of between 9 and 12V. The device will not power up if a plug of the wrong polarity is inserted (although it will not be damaged)
- A standard 5-pin MIDI input socket
- A 3.5mm stereo jack socket for MIDI input. Using a stereo jack cable, you can connect a device which uses a 3.5mm jack MIDI output (such as Novation Circuit or Arturia Beatstep Pro). This is an alternative to using the 5-pin socket. You should not try to use both inputs at the same time. They are parallel connections and will not merge MIDI data. Trying to use both at the same time may even damage your sending device.
- Four 3.5mm jack sockets, labelled A through D, which output analog control voltage signals

- Twelve 3.5mm jack sockets, labelled 1 through 12, which output digital gate signals

LEDs

CV.OCD has three LED indicators:

- The BLUE LED indicates MIDI activity
- The YELLOW LED usually indicates presence of a MIDI clock and blinks on each beat. Some other actions use this LED to show special statuses. These will be described in this text.
- The RED LED shows that the device is powered

Button

CV.OCD is very configurable, but all configuration changes must be made over MIDI. In fact, there is only one button on the box, and it has the following functions:

Press and release the button to reset the outputs. This will

- clear all held MIDI notes
- turn off all gates (see note)
- set all CV outputs to zero (see note)
- reset note cycle modes to the first note
- reset clock dividers
- The yellow LED blinks once when the reset is registered.
- a CV mapped to a pitch bend will go to it half-voltage position (no bend) on reset.
- A CV mapped to a fixed voltage will not reset. An “all notes off” gate will be triggered.

Press and hold the button for about 2 seconds to save the current configuration. The configuration will be restored next time CV.OCD is switched on. The yellow LED blinks a second time once the save takes place. Note that patches loaded via Sysex are always saved. The button is only needed if you want to save the current state after making “live” changes using NRPN messages.

The button also has a special use at power up so that new firmware can be uploaded. This process will be described later in this manual

The Configuration Web Page

Introduction

CV.OCD uses a web page to set up configuration parameters so that a MIDI Sysex file can be downloaded and saved to your computer. You can send this file to the CV.OCD to set the new configuration and you can save the file for later use (you can build up a library of Sysex files for different CV.OCD configurations). CV.OCD stores the last patch when power is off, so you don't need to upload the Sysex file each time you power up.

The configuration web page is accessible at the following URL

six4pix.com/cvocd/patch.asp

This manual show screen shots of the CV.OCD SYSEX configurator at various points

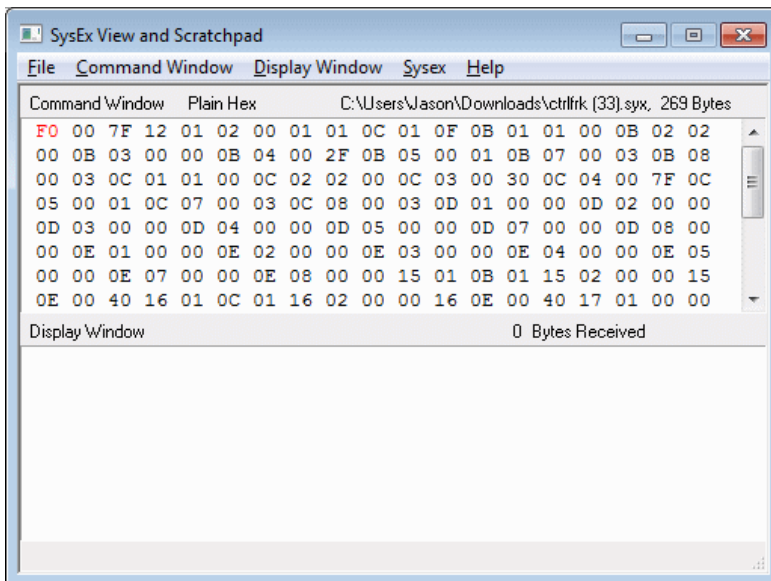
Downloading Sysex

The page lets you create a CV.OCD patch by making selections in the drop-down lists. You can then press the submit button at the bottom of the page and download a SYSEX file containing your settings. The submit button is labelled “I WANT YOUR SYSEX” (shown below)

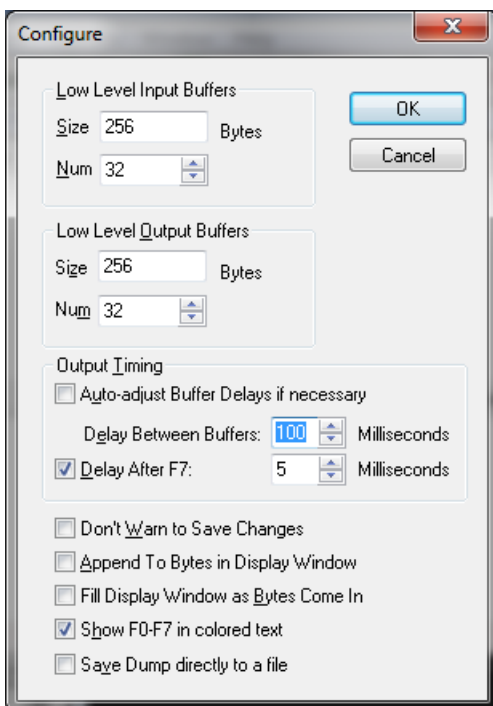
The screenshot shows a web form with several dropdown menus. The first dropdown is labeled 'Gate.12' and has 'disable' selected. The second dropdown is labeled '1st priority note' and has '1st priority note' selected. The third dropdown is labeled '(default)' and has '(default)' selected. The fourth dropdown is labeled '0 C-1' and has '0 C-1' selected. The fifth dropdown is labeled '127 G9' and has '127 G9' selected. Below these dropdowns is a submit button with the text 'I WANT YOUR SYSEX!' highlighted by a red box.

Sending a patch to CV.OCD

Use a SYSEX tool (such as MIDI-OX on Windows or SysexLibrarian on Mac) to send this file to your CV.OCD and patch it with your new settings.



Set up your Sysex tool to have a delay between SysEx messages. This is how I configure MIDI.OX



Viewing or editing an existing patch

The configuration web page allows you to open an existing patch which you already have as a SysEx file. Use the buttons at the top of the form to select then upload the file:



Layout of the configuration page

The function for each control will be explained in the following sections, however at a very basic level the configuration page has the following sections

- At the top are the controls that allow you to load an existing Sysex file into the page
- Below this are Global settings which define some optional standard values for other mappings (such as the MIDI channel).
- The next section defines the four “note inputs” which listen to incoming MIDI notes and deal with prioritising them and assembling them into chords which can be mapped to CV and Gates outputs
- The next section defines the source of the voltages for each of the four CV outputs
- The next section defines the source of the status for each of the twelve gate outputs

The Note Inputs

Introduction

In the diagram near the start of the manual you can see that a musical note (green flag) can only be mapped to a CV output if it has come through a “Note Input”. So we can’t simply map a MIDI note to a CV output... what is this note input and why is it needed?

OK, let’s imagine that we want to play a monosynth over MIDI. If we play notes one at a time this works just fine - the synth plays the note we press. However, what if we press more than one note at a time?

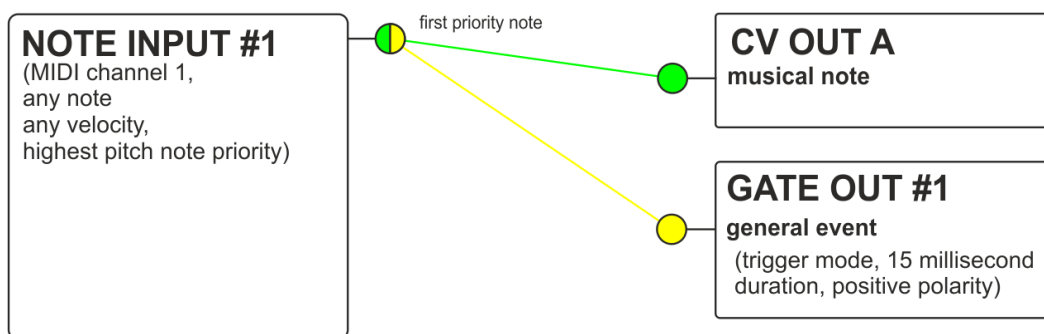
A monosynth played via MIDI must decide to play just one of the notes you are holding, and when you release that note it may (depending on what synth you have) start playing one of the other notes that is also held. This “prioritisation” of the notes is a useful feature and can be used to get some nice fluid playing like trills or alternating lead and bass notes.

CV.OCD must make the same decision, so something needs to “listen” for the MIDI notes and decide which note to send to the CV output. This is where “Note Inputs” come into play... basically they prioritise the notes and decide which is the note that should be mapped to a CV out. For polyphonic modes there might be multiple CV outputs playing the notes, so this is why the Note Input is a separate thing to the CV output itself.

Basic note mapping

As a simple example, let’s say we have a monosynth that we want to control with MIDI notes on MIDI channel #1, using highest pitch note priority. We want to accept any notes at any velocity and we want the note pitch to go to CV output A, and gate to go to gate output #1.

Based on our diagram, we’d have a situation like the following



We need to set up two mappings - firstly to get the first priority note from note input #1 and output it on CV A, and secondly to trigger gate #1 when the note is played.

To actually do this we would configure the CV.OCD using the configuration tool to make a SysEx file that we can then load into it. For example

Globals (TIP: hover mouse over field names for a description)

Default Channel	Default Trigger
Chan. 1	15ms

Note Inputs

	Enable	Channel	Mode	Min.Note	Max.Note	Min.Vel.	Pitch Bend
Input 1	ENABLE	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input 2	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input 3	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input 4	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3

CV outputs

CV Source	Note input	Transpose	Chan	CC#	Range
CV.A	Note input #1	(none)	(default)	Mod Wheel	5V
CV.B	(disable)	First note pitch	(default)	Mod Wheel	5V
CV.C	(disable)	First note pitch	(default)	Mod Wheel	5V
CV.D	(disable)	First note pitch	(default)	Mod Wheel	5V

Gate Outputs

Source	Note input event	Chan	Min.Note	Max.Note	Min.Vel.	CC#	Switch@	Ck.Rate	Trk.Ofs	Trig
Gate.1	Note input #1	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(gate)
Gate.2	(disable)	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(trigger)
Gate.3	(disable)	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(trigger)

Monophonic note priority modes

Each note input has three note priority modes which are used for monophonic assignments. These are

- **Last note priority** – The most recently played note takes precedence. If multiple notes are held down and the currently playing note is released, the next most recent note plays.
- **Highest note priority** – The highest pitched note takes precedence. If multiple notes are held down and the currently playing note is released, the next highest note plays.
- **Lowest note priority** – The lowest pitched note takes precedence. If multiple notes are held down and the currently playing note is released, the next lowest note plays.

Polyphonic note mapping modes

You may have noticed on the diagram that each of CV.OCD's Note Inputs has little "flags" for the first note, but also for the second, third and fourth notes.

These outputs are used in the polyphonic "chord" and "cycle" modes where a single note input can control up to four CV and four gate outputs.

- In **Chord** modes we can map up to four held notes at the same time, so we could use four monosynths together to play a chord of up to four notes. We can map the first, second, third and fourth notes to their own CV and Gate outputs
- In **Cycle** modes, the operation is similar except that consecutive notes that are received over MIDI are assigned cyclically to the first, second, third and fourth notes (even for the same note being received multiple times). You can return to the first note position using the button on the front of the unit (this also resets other settings)

Each Chord and Cycle mode has options for 2, 3 and 4 notes (depending on how many outputs you are using)

Here is an example of how you can map four CV/gate output pairs to play a 4 note chord across four monosynths or voices.

Note Inputs

	Enable	Channel	Mode	Min Note	Max Note	Min Vel	Pitch Bend
Input 1	ENABLE	(default)	4 note chord	0 C-1	127 G9	(any)	+/-3
Input 2	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input 3	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input 4	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3

CV outputs

	CV Source	Note input	Transpose	Chan	CC#	Range
CV.A	Note input #1	First note pitch	(none)	(default)	Mod Wheel	5V
CV.B	Note input #1	Second note pitch	(none)	(default)	Mod Wheel	5V
CV.C	Note input #1	Third note pitch	(none)	(default)	Mod Wheel	5V
CV.D	Note input #1	Fourth note pitch	(none)	(default)	Mod Wheel	5V

Gate Outputs

	Source	Note input event	Chan	Min Note	Max Note	Min Vel	CC#	Switch@	Ck Rate	Tk Ofc	Trig
Gate.1	Note input #1	First note on	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(gate)
Gate.2	Note input #1	Second note on	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(gate)
Gate.3	Note input #1	Third note on	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(gate)
Gate.4	Note input #1	Fourth note on	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(gate)

Pitch bend

MIDI pitch bend is applied automatically to notes that are mapped via a Note Input. You can use the configurator to specify the pitch bend range used by a given note input, for example

Note Inputs

	Enable	Channel	Note Priority	Min.Note	Max.Note	Min Vel.	Pitch Bend
Input 1	ENABLE	(default)	Lowest pitch	0 C-1	127 G9	(any)	+/-3
Input 2	(disable)	(default)	Lowest pitch	0 C-1	127 G9	(any)	+/-3

If applying pitch bend would take a note outside of the range of a CV output, the output will “plateau” at the highest or lowest allowable value.

You can also use pitch bend directly to control a CV output. This is explained further below.

Multiple MIDI channels

CV.OCD allows you to control each of the four CV outputs (and associated gates) using different MIDI channels. This allows you to run four monosynths on their own channels.

To do this you’ll need to use a Note Input for each MIDI channel. The following image shows how this can be configured

Note Inputs

	Enable	Channel	Mode	Min Note	Max Note	Min Vel	Pitch Bend
Input.1	ENABLE	Chan.1	Lowest note priority	0 C-1	127 G9	(any)	+/-3
Input.2	ENABLE	Chan.2	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input.3	ENABLE	Chan.3	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input.4	ENABLE	Chan.4	Highest note priority	0 C-1	127 G9	(any)	+/-3

CV outputs

	CV Source	Note input	Transpose	Chan	CC#	Range
CV.A	Note input #1	First note pitch	(none)	(default)	Mod Wheel	5V
CV.B	Note input #2	First note pitch	(none)	(default)	Mod Wheel	5V
CV.C	Note input #3	First note pitch	(none)	(default)	Mod Wheel	5V
CV.D	Note input #4	First note pitch	(none)	(default)	Mod Wheel	5V

Gate Outputs

	Source	Note input event	Chan	Min.Note	Max.Note	Min Vel.	CC#
Gate.1	Note input #1	First note on	(default)	0 C-1	127 G9	(any)	Mod V
Gate.2	Note input #2	First note on	(default)	0 C-1	127 G9	(any)	Mod V
Gate.3	Note input #3	First note on	(default)	0 C-1	127 G9	(any)	Mod V
Gate.4	Note input #4	First note on	(default)	0 C-1	127 G9	(any)	Mod V
Gate.5	(disable)	First note on	(default)	0 C-1	127 G9	(any)	Mod V

Every note input has its own note priority setting, so you can have different note prioritisation used on each channel.

Keyboard splits

CV.OCD allows more than one Note Input to be assigned to the same MIDI input channel. This allows you to set up keyboard splits. For example, let's say we want notes below C3 to play a bass synth and notes above to play a lead synth, and both synths should play with highest note priority.

Here is a patch which could do this

Note Inputs

	Enable	Channel	Mode	Min.Note	Max.Note	Min Vel.	Pitch Bend
Input.1	ENABLE	Chan.1	Highest note priority	0 C-1	47 B2	(any)	+/-3
Input.2	ENABLE	Chan.1	Highest note priority	48 C3	127 G9	(any)	+/-3
Input.3	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3
Input.4	(disable)	(default)	Highest note priority	0 C-1	127 G9	(any)	+/-3

CV outputs

	CV Source	Note input	Transpose	Chan	CC#	Range
CV.A	Note input #1	First note pitch	(none)	(default)	Mod Wheel	5V
CV.B	Note input #2	First note pitch	(none)	(default)	Mod Wheel	5V
CV.C	(disable)	First note pitch	(none)	(default)	Mod Wheel	5V
CV.D	(disable)	First note pitch	(none)	(default)	Mod Wheel	5V

Gate Outputs

	Source	Note input event	Chan	Min.Note	Max.Note	Min Vel.	CC#
Gate.1	Note input #1	First note on	(default)	0 C-1	127 G9	(any)	Mod
Gate.2	Note input #2	First note on	(default)	0 C-1	127 G9	(any)	Mod
Gate.3	(disable)	First note on	(default)	0 C-1	127 G9	(any)	Mod

The key ranges assigned to Note Inputs can also overlap if you like - this enables some interesting patching and playing styles - for example you could have a bass synth mirror the lowest note of a chord played across three lead synths. Loads of fun to be had!

CV Outputs

Note pitch CV

When CV.OCD is sending out a musical pitch CV, it uses a 1 Volt/Octave standard and has a range of 8 Octaves. Middle C (MIDI note 60) is set at 3 volts, so the output range is approximately MIDI note 24 (C1) to 120 (C9)

- C1 (MIDI 24) = 0V
- C2 (MIDI 36) = 1V
- C3 (MIDI 48) = 2V
- C4 (MIDI 60) = 3V
- C5 (MIDI 72) = 4V
- C6 (MIDI 84) = 5V
- C7 (MIDI 96) = 6V
- C8 (MIDI 108) = 7V
- C9 (MIDI 120) = 8V

Notes which lie outside of this range will be transposed up or down by whole octaves to bring them into the playable range.

If you really want out of range notes to be muted (rather than transposed into range), you can restrict the note range of the Note Input that is playing the CV output.

If you want to use a different octave mapping you can use the transpose function of the Note Input. For example if you want MIDI note 0 (C-1) to map to 0 volts, you can set a CV transpose of +2 octaves.

Notes which are pushed out of the playable range by pitch bend will “plateau” at the limit.

Other CV sources

CV.OCD’s four analog CV outputs can output signals other than musical note pitches. You can also use them to output

- Velocity of the most recently played note (Specify note input)
- Continuous controller (CC#) value (Specify MIDI channel and CC number)
- Pitch bend (Specify MIDI channel)
- Channel aftertouch (Specify MIDI channel)
- A CV which is linked to MIDI clock BPM (0-255 bpm range)
- Fixed voltages for calibration etc.

For each of these you can specify the full CV voltage range from 1V up to 8V. Additional parameters needed for a mapping are listed above. For example to map a CC to a CV output we need to give the CC number and the MIDI channel

CV outputs

	CV Source	Note input	Transpose	Chan	CC#	Range
CV.A	MIDI CC	1st Priority Note	+3oct	Chan.1	cc#10	5V
CV.B	(45-61)	4th Priority Note	+2oct	(45-61)	Mod Wheel	0V

BPM To CV mode maps a BPM of 0-255 BPM over the full voltage range. The output is updated once per beat, so updates are slow at low BPM and rapid changes in BPM will result in a “stepping” of the CV. If the MIDI clock stops, this does not reset the CV (the button can be used)

A Fixed Voltage can be output by selecting the required voltage in the Range drop-down. The ability to output a fixed voltage can be useful for calibrating the pitch CV input of a synth.

Gate Outputs

General operation

CV.OCD has twelve assignable gate outputs. Each of these outputs can be switched ON and OFF in response to information coming in from MIDI.

- When the gate output is ON, it outputs 5 Volts
- When the gate output is OFF, it outputs 0 Volts

Depending how a gate output is mapped, it will switch ON and OFF when a corresponding event happens on the MIDI input. There are three modes available

- **Gate Mode** keeps the output switched ON as long as the corresponding condition is true and turns it off when the condition is over. For example we can set a gate to be ON for as long as the MIDI clock is running.
- **Trigger Mode** only sets the output ON when the corresponding condition first becomes true. The output then turns off after a specific period of time (up to 125 milliseconds). The time period can be selected directly, or you can use the default trigger period set up in the global settings for the patch.
- **Retrigger Mode** is rather like Gate mode in that it keeps the output switched ON as long as the corresponding condition is true and turns it off when the condition is over. However each time the condition is signalled, CV.OCD makes sure that the output is turned off for a very short period before it going back on again. For example, this allows for playing styles where you want to control the sustain of a note by holding down a key, but also have an attack envelope which triggers each time a new key is pressed, even if you are playing “legato” (only releasing a key after the next is pressed)

These three modes are summarised with the picture below, which shows how three incoming note events would trigger the gate output in each case.



Select **(gate)** in the Trig setting for gate mode and **(retrig)** for retrigger mode. For Trigger mode you can select a specific time in milliseconds or just select **(trigger)** to use the default trigger period defined in the global settings.



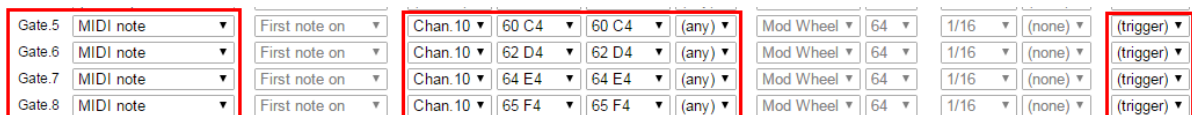
The Trigger period in global settings applies only to **(trigger)** mode. In retrig mode the “low” time is very short (~1ms) and just enough to give a rising edge to trigger your synth gear.

MIDI Note based triggers

When playing pitched notes, gates would be mapped via the Note input, as described above. However, sometimes it is useful to tie a gate output to a specific MIDI note directly. One obvious example is for triggering drum sounds.

Here is an example of mapping MIDI notes (60, 62, 64, 65 on channel 10) to gate outputs 5,6,7,8 respectively

In these cases, where we are not worried about creating a pitch-based CV but rather just listening for a specific MIDI note, we can bypass the Note Inputs completely and simply tie the gate output to the MIDI note. We can do this like so...



We can assign a single note or a range of notes to a gate output. We can also specify a velocity threshold so that the gate fires only when the MIDI note is above that velocity. This can be useful for bringing in additional gates for harder hits, or maybe for adding an accent trigger which fires when any individual drum trigger is above a certain velocity threshold. For example

Gate.5	MIDI note	First note on	Chan. 10	60 C4	60 C4	(any)	Mod Wheel	64	1/16	(none)	(trigger)
Gate.6	MIDI note	First note on	Chan. 10	62 D4	62 D4	(any)	Mod Wheel	64	1/16	(none)	(trigger)
Gate.7	MIDI note	First note on	Chan. 10	64 E4	64 E4	(any)	Mod Wheel	64	1/16	(none)	(trigger)
Gate.8	MIDI note	First note on	Chan. 10	65 F4	65 F4	(any)	Mod Wheel	64	1/16	(none)	(trigger)
Gate.9	MIDI note	First note on	Chan. 10	60 C4	65 F4	100	Mod Wheel	64	1/16	(none)	(trigger)

CC based gates and triggers

A gate output can be tied to a specific MIDI continuous controller (CC) value such that the gate is activated when the CC value goes above or below the threshold

For example, we can set a gate to activate when CC#10 is above the value 64

Gate Outputs

	Source	Note input event	Chan	Min.Note	Max.Note	Min.Vel.	CC#	Switch@
Gate.1	CC above threshold	No notes	Chan. 1	0 C-1	127 G9	(any)	cc#10	64
Gate.2	(disabled)	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64

We can also set a gate to trigger when a CC is below a threshold.

In all cases the gate cannot trigger until we receive the current value of the CC from MIDI (i.e. when we first connect CV.OCD we don't know the current CC value until it changes)

MIDI transport based gates and triggers

MIDI has the concept of a "play" mode (or "transport"). When the transport is running, this means that pattern based devices and sequencers would be playing notes in time with the MIDI clock. The transport can also be stopped.

MIDI controls playback with three messages:

- **START** - This message is usually sent by a master controller device when a slave device should reset its play position to the beginning of a sequence and should start playing in time with MIDI clock ticks. When this message is received, CV.OCD triggers any gate with a "Transport restart" or "Transport running" condition.
- **CONTINUE** - This message is usually sent by a master controller device when a slave sequencer pick up playing from its current position. When this message is received, CV.OCD triggers any gate with a "Transport running" condition.
- **STOP** - This message stops playback at the current position. When this message is received, CV.OCD triggers any gate with a "Transport stop" condition.

It should be noted that the MIDI clock tick messages (which define the BPM) continue to be sent by the master device even when the transport is stopped.

MIDI clock based gates and triggers

A very useful application of CV.OCD is to output clock pulses when MIDI clock tick messages are received.

A MIDI clock master defines the BPM by sending out 24 of these tick messages for every "beat" (quarter note) so at 120bpm, 48 ticks are received every second.

CV.OCD can trigger its gate outputs in response to these tick messages. The outputs can then be used to synchronise external gear with MIDI.

Most usefully, you can specify the rate at which the gate output will fire, as a division of the tempo. CV.OCD can output a clock at the following rates:

- 1/32nd note - 3 ticks
- 1/16 note triplet - 4 ticks
- 1/16 note - 6 ticks
- 1/8 note triplet - 8 ticks

- 1/16 dotted note - 9 ticks
- 1/8 note - 12 ticks
- 1/4 note triplet - 16 ticks
- 1/8 dotted note - 18 ticks
- 1/4 note (beat) - 24 ticks
- 1/2 note triplet - 32 ticks
- 1/4 dotted note - 36 ticks
- 1/2 note - 48 ticks
- 1/2 dotted note - 72 ticks
- whole note - 96 ticks
- 24ppqn – one pulse every MIDI tick

When you output the clock, you need to use a trigger duration that is short enough so that there is an OFF time between the clock pulses, but long enough to register on the device being controlled. A good choice is usually 10 milliseconds or so.

Here is an example of how a beat clock (1/16th) can be output to gate 1, with gates 2,3,4 outputting divided clocks at half, quartet and eighth speeds.

Gate Outputs

	Source	Note input event	Chan	Min.Note	Max.Note	Min.Vel.	CC#	Switch@	Ck.Rate	Tk.Ofs	Trig
Gate.1	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	15ms
Gate.2	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/8	(none)	15ms
Gate.3	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/4	(none)	15ms
Gate.4	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/2	(none)	15ms
Gate.5	(disable)	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(trig)

Another option is to “offset” the clock. This means to add a lag to the timing of the clock so it triggers later than it should (by a number of ticks that you specify). Remember that each tick is 1/24th of a beat (use the list above to know how many ticks there are in each clock division)

For example lets create four clocks that trigger once per beat but are each staggered by a quarter of a beat. 1 beat is 24 ticks, so each output should be offset from the previous one by a quarter of this (i.e 6 ticks)

Gate Outputs

	Source	Note input event	Chan	Min.Note	Max.Note	Min.Vel.	CC#	Switch@	Ck.Rate	Tk.Ofs	Trig
Gate.1	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	15ms
Gate.2	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	+6	15ms
Gate.3	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	+12	15ms
Gate.4	Clock Tick	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	+18	15ms
Gate.5	(disable)	No notes	(default)	0 C-1	127 G9	(any)	Mod Wheel	64	1/16	(none)	(trig)

Firmware Updates

The firmware of CV.OCD can be updated through the MIDI port using special Sysex files

To put the device into firmware update mode you need to hold the button while powering the device on. Both the yellow and blue LEDs should light solidly.

Send the SysEx file to CV.OCD using a MIDI SysEx program (such as MIDI-OX on Windows). Make sure that there is no other MIDI traffic on the connection at the time of the upload, and configure your Sysex program to place a delay of 100ms or so between each SysEx buffer. The upload should complete within a minute or two.

When the upload is completed, the yellow and blue LEDs should blink alternately and rapidly. You need to remove power and apply it again to boot up CV.OCD with the new firmware

If there is an error during the upload, the blue LED is off and the yellow LED is blinking. Check your settings (and make sure there is no other MIDI activity) before trying again.

Note that an incomplete firmware upload may leave the device in an unusable state, but you should still be able to access the firmware upload mode to reload firmware.