Z3 Zero, Dual Zero Difference

The Z3 is a two channel zero difference trigger that can be CV controlled.

It is the same evaluating click-less switching control circuit used in our D-2, 4-Band Distortion module but with all the ins and outs accessible to make it as useful as possible in the context of a modular synthesizer.

The Z3 was mainly designed for audiorate wave-splicing together with a Switch but it can also be used to detect zero crossings of signals, to detect zero difference between two signals, to perform click free switching, to add jitter to signals, to synchronize signals and many other things. It is mainly designed to be used in combination with other modules such as the Switch module but can also be used on its own.

As the technical nature of this module can get a little abstract we thought we should start with some clear examples of its use.
There is a more detailed description further on in the manual for those that are interested.

Controls

A
A is one of the inputs to a comparator (B is the other). A is compared to B. If the voltage on A is higher than B then Out will go high if D is high or Out will go low if D is low. D is controlling what Out will be after A has become higher than B.
A is normalled to 0V if nothing is connected.

B
B is the other input to the comparator.
B is normalled to 0V if nothing is connected.

D
D is controlling what Out will be after A has become higher than B (a zero event).
D is a logic input but works with audio too. A signal 1V or more is seen as a logical high signal.

If nothing is connected to D the output is just controlled by A and B and will toggle from high to low or low to high every time it gets triggered by the comparator (free running mode).

Out
Out is a logic output where high is +5V and low is 0V.
It is meant as a logic control signal but can of cause be used as audio.
The LED next to the Out shows the status of the output signal.

In short you can compare two signals on A and B and when they are equal Out will toggle.
Or you can compare signals on A and B but Out will only switch to what you tell it to switch to by the signal in on D.
This sounds simple but can give you access to functions that are hard to achieve in other ways.

Device specs

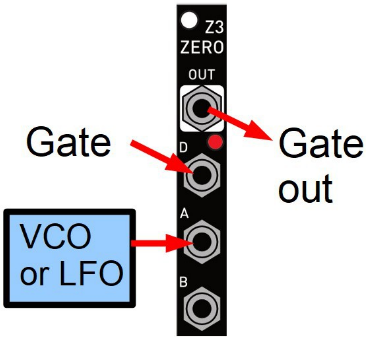
Module size:	2 hp wide, 29 mm deep with power connector.
Input impedance:	100 kohm
Out impedance:	1 kohm
Power requirements:	+12V. Max power consumption 10 mA
	-12V. Max power consumption 3 mA
	Connect the power cable with the red stripe towards the marking -12V on the board.
	The unit is protected for reverse power.

Use case examples

A few examples, just to get your imagination going.

Jitter generation (swing)

Jitter is when the rising and falling edges of a signal vary around the time where they are expected to be. This is what is called swing in sequencers. With the Z3 you can add that to any signal.



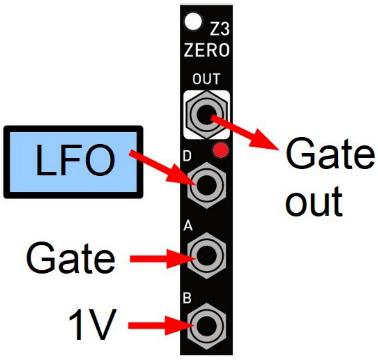
A simple example is for drums.
Connect as the picture and you can set the amount of jitter with the VCO or LFO. High frequency low jitter. If you lower the frequence a lot, the jitter will be so extreme so it starts skipping gate signals and you have a random variation in your gate rhythm.

An other example can be to connect it on the sync signal between two VCOs instead of the gate signal.
This will give you a very dirty sound out from the two jitter synced VCOs.

Or simply one VCO in to D and another VCO to A. Then use the Out signal as your sound source. Can end up as a very aggressive square sound.

Synchronizing

This is like the previous jitter example but syncing signals instead of making them unsynced.

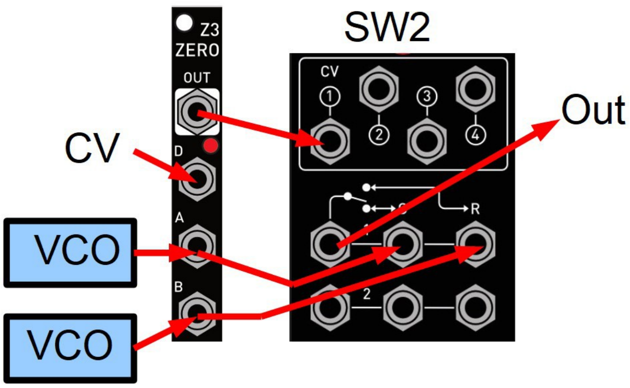


Again the drum example from before but LFO and Gate in has swapped places. This will also generate a varied drum pattern with the frequency of the LFO controlling how many gate signals that will be let through to Out. The difference from the previous example is that the gate out signals will be in perfect sync with the gate in on A.

The reason there is a 1V signal in to B is that B is normalled to 0V and a gate signal from a sequencer (or other logic signal) will not completely go down to 0V. It can actually be 0,1V and then the gate signal will never pass the detection threshold on B. So you need to feed the circuit with a voltage between 1V to 4V.

Wavesplicing, clickless switching

A way to switch audio or CV sources without any glitches in combination with the SW-1 Switch module.



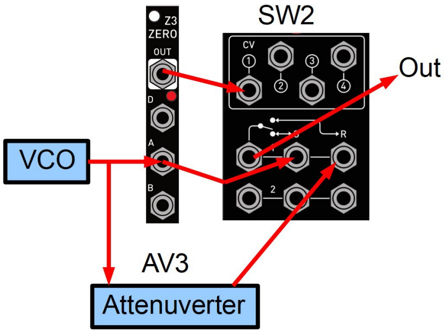
Start with D disconnected. Free running mode.
A and B listens to the two VCOs you are going to switch between.
When the Z3 detects that A and B are on the exact same voltage Out will toggle and a switch is made from one VCO to the other. This way you get no click in the switching and a seamless wave-splicing.

If you connect a gate signal to D you can control the switching with it. When the gate signal goes high the Z3 will wait till the two VCOs are at the same voltage and a switch will be made.

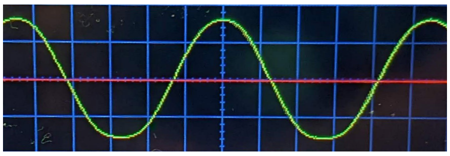
Try and connect an unsynced VCO to D for controlling when it shall switch. Could give an almost broken but not harsh sound, or all sorts of interesting sounds.

Switched rectification

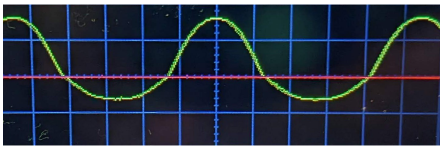
Using the zero crossing detection to switch in only one half period of a simple signal, like a sine wave. This means you can do half or full wave rectification and anything in between!



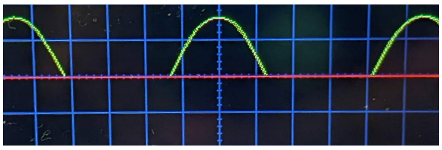
Connect a sine wave from the VCO as shown. Set the attenuverter to full on. You will have exactly the same sine wave out in this start position.



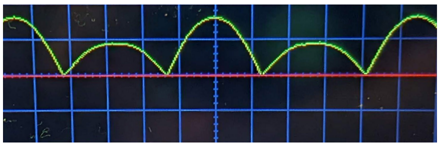
The Z3 is now detecting zero crossings and controlling the switch so the positive part of the wave comes through one side of the switch and the negative comes through the other side.



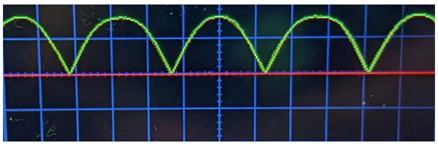
If you start turning the attenuverter only one half period of the wave will be affected.



Turning the attenuverter to zero will give you a perfect half wave rectified signal.



If you turn the attenuverter over to the negative side you will start getting a wave that looks almost full wave rectified.

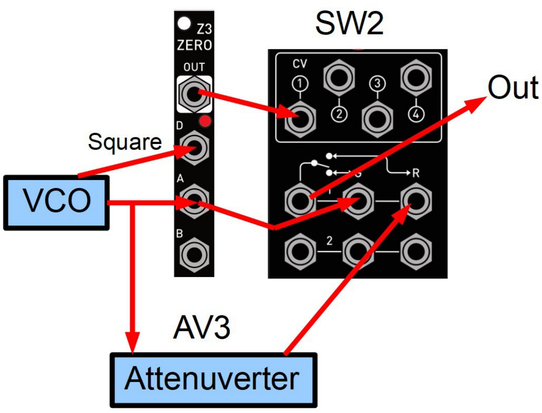


Finally a perfect full wave rectification with the attenuverter turned max to the negative side.

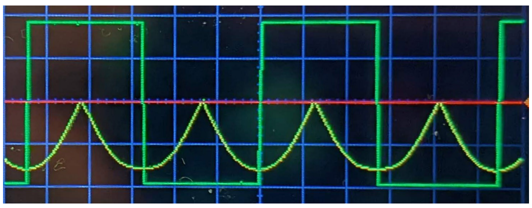
At this point you may think that something is wrong with your connections because your wave may be turned upside down!

The reason the wave may be upside down is that with nothing connected to D the Z3 is in free running mode and just toggles the state of Out everytime it detects a zero event. You have no control over if it started high or low the first time.

This is normally not a problem for audio as a wave sounds the same if it's upside down.



You can control this by connecting the squarewave from the same VCO to D. Now the switch will always turn on at a zero event when the square wave is high and it will turn off at a zero event when the square wave is low.



In the picture you can see the added squarewave.

As you can the rectified wave ended up being upside down.

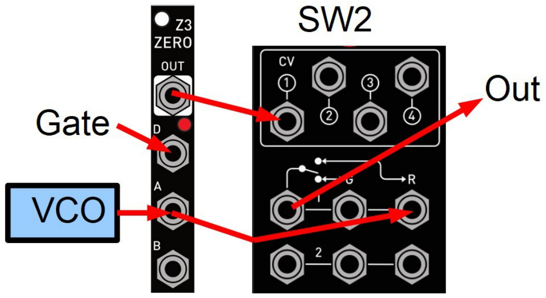
You can turn it the other way by simply swapping the two patch cords connected to G and R on the Switch.

If you would do something like this with a harmonically rich sound like a complex wavetable it can be good to have a simple wave connected to D to help the Z3 know how often it is allowed to react to the signals on A and B.

In general, if the result on Out gets too chaotic. Connect something to D that restricts it in some way.

Zero crossing switching

If you don't want to use a VCA to turn a sound on you can switch it on. To get a super fast tight note on and off. This is especially useful for bass sounds.

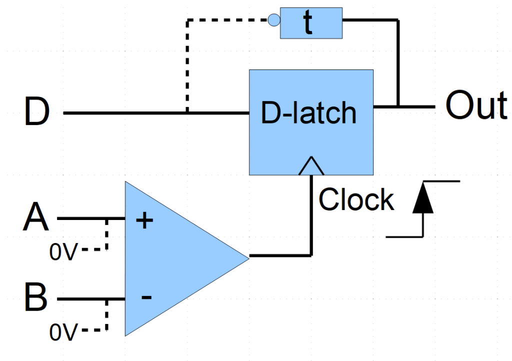


Very similar to the clickless switching example.

When the gate is set high the Z3 waits till the next zero crossing of the VCO signal. Then Out goes high and the sound is switched on without any click.
The same happens when the gate goes low. The Z3 waits for the next zero crossing before switching the sound off.

In depth description

The function of the Zero 2 is not what you would find in every synth so it may need a deeper explanation.



Here is a simplified block schematic of what's going on inside the module. The dotted lines are normalling when nothing is connected to the jack.

In the center of this there is a D-latch. You can see it as a one bit memory. It is a logic circuit.
If you put a high signal on D nothing happens, but if you feed it a rising edge on the Clock signal (internal signal) it will read what's on D and copy that to Out.
Out will then keep that high signal no matter what's going on at the D input until there is a Clock signal telling it to read a low signal from D.

The other part of the circuit is a comparator. Simply put, if the voltage on A is higher than B then A wins and the output from the comparator goes high. That's why there is a plus sign in the symbol.
If B is higher than A then B wins and the comparator output goes low.
The positive edges of the comparator output is what is clocking the D-latch.

The difference in this case compared to a normal comparator is that this one doesn't have any hysteresis. That means it will react more accurately when comparing A and B, which is key in fast accurate wave splicing.
The downside is that it will be more sensitive to noise on the signals on A and B. If it had been just a normal comparator without the D-latch you would have had ugly bursts going out, but as you have the D-latch here the Out can only change once to the high or low that you have put on the D input.
So the frequency in on D will be the maximum frequency on Out.
What you put on D can be anything from DC to a signal high above audio rate. You can run this circuit very fast if you like.

If you don't connect anything to D a time delay is normalled to the D input. This creates a time window when the D-latch isn't allowed to change the Out signal. It is a way to handle the fact that the comparator doesn't have the traditional hysteresis.
The Out will toggle from low to high or from high to low depending on what it was before at every clock event from the comparator.
In this mode the maximum frequency you can run from A or B to Out will be about 7 kHz.
That is enough for most use cases and will block most unwanted behavior in the comparator. If you find that it still reacts in a way you don't want you can add a signal to D to make it react less (or more).

Please check www.dpw.se for updates of the manual and demo videos.