Pre-release

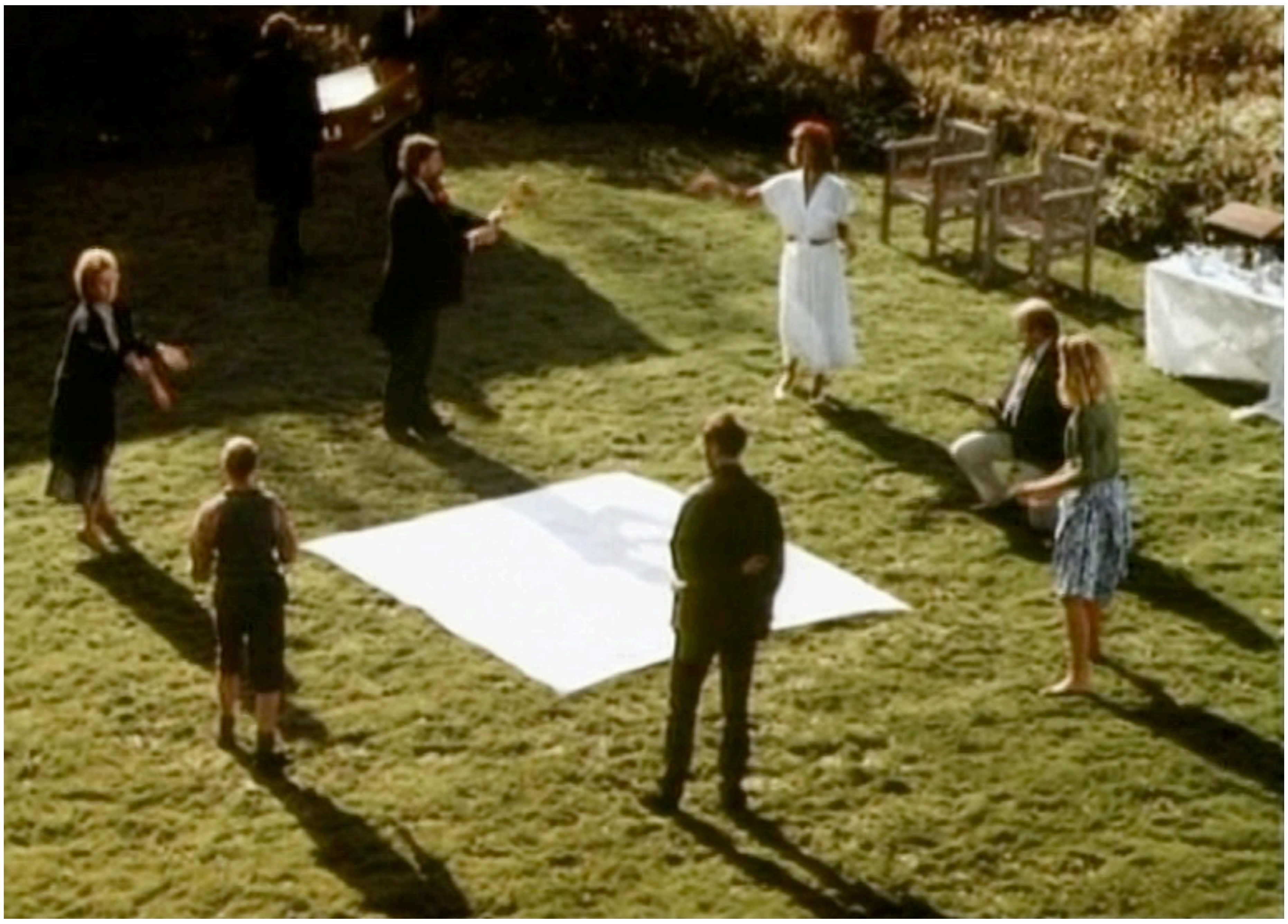
DMC-v0.7-beta

61d21a1

Dead Man's Catch v0.7-beta for the Mutable Instruments Peaks module

timchurches released this 2 days ago

Dead Man's Catch Version 0.7-beta



This is a beta release of Dead Man's Catch, which is modified firmware for the Mutable Instruments Peaks eurorack synth module, in the same way that Bees-in-the-Trees is modified firmware for the Mutable Instruments Braids eurorack synth module.

This release of Dead Man's Catch adds several capabilities to the standard Peaks firmware:

- it unlocks some hidden envelope modes which Olivier Gillet, the owner of and principal developer at Mutable Instruments, put into the Peaks firmware from the outset: a double-attack envelope, a repeating attach envelope, a looping envelope, and a Bouncing Ball envelope generator. The Bouncing Ball envelope emulates a ball which is thrown into the air and then allowed to fall back to earth and bounce several times - think of a basketball bouncing on a basketball court. The envelope output equates to the height of the ball at any instant.
- it adds a "randomised AD envelope" mode, in which the level (amplitude) and decay times of the envelope have random amounts added to them each time the envelope fires. The amount of randomness added is adjustable, separately for amplitude and decay time.
- four new internally-modulated LFO functions have been added. These behave like the standard LFO except each has its own internal folded sine-wave modulator or random value (sample and hold) modulator that modulates either the LFO frequency (in the FM LFO

function), or the LFO wave shape (in the wave shape-modulation LFO function). Yes, a frequency-modulated LFO, inside your Peaks! The two waveform-modulated LFOs also feature an additional waveform: an overdriven sine wave, which morphs from a pure sine to a square wave with rounded corners. Not only useful as an LFO, it also sounds nice at audio frequencies.

- dual PLL (phase-locked loop) oscillators (aka PLOs) have been added. This mode is based on the tap LFO mode and uses the same pattern predictor, but it has ben tweaked for use at audio rates. It will track the frequency of most oscillators running at audio rates (and LFOs too), and puts out a signal in unison, or one, two or three octaves below or above - yessir (or, maybe, Yasser), it's a sub-oscillator (or a super-oscillator)! Waveforms are selectable and can be modulated with an internal sine wave modulator. Note that this PLO is not guaranteed to track all possible signals, and it may glitch or lose the frequency lock, depending on the input. That can be good, or bad.
- it adds "randomised" bass and snare drums, in which the pitch and/or the level or decay time of the bass (kick) and snare drum models follows a random walk, with the stride of each step being adjustable from zero (no randomisation) to a fair bit.
- it adds dual Turing Machines, which are semi-random looping CV sequencers, modelled closely on the Music Thing Turing Machine.
- it adds a "ModSequencer", which works in a similar way to the DinSync MODSEQ module, and which is, under the bonnet, just a very simple extension to the existing Peaks mini-sequencer mode.
- dual "byte beat" noise generators have been added, with clock rate and two other parameters settable via the front panel pots. Eight different byte beat "equations" are available (it is easy to add more in later versions, or better equations substituted).
- the high hat model has been moved to its own function, rather than being available only as part of the snare drum as in the factory firmware. The high hat is now also adjustable. However, it runs at half the sample rate of the high hat in the factory firmware, and thus sounds somewhat different. The high hats offer randomisation, too, and mutual exclusivity, explained below, which facilitates their use as open and closed high hats with modules such as Grids which output an accent trigger synchronised with the main trigger output for each channel.
- the user interface for the four basic functions, as labelled on the panel, is unchanged, but the user interface for the alternate functions (of which there are many more than in the standard Peaks) has been changed substantially - there are now four alternative function "pages", explained in more detail below.

These instructions assume reasonable familiarity with Peaks. If in doubt, please consult the official Peaks manual.

Some Observations on this Release by its Author

Please note that this is a *beta* release, and in many ways must be considered as a test bed for additions and enhancements which might find their way into one or more production versions of Dead Man's Catch. In particular, the user interface for this version is still a work-in-progress, and although it is more usable than the user interface catastrophe of v0.5-beta, it is still far from ideal. However, it does not (or at least should not) crash or fail to work, and the new alternative function "paged" paradigm offers a lot more user-interface "space" in which to add more variations on function themes. I suspect this is the best that can be achieved given the Peaks hardware design, but further or alternative suggestions are welcome, and it is acknowledged that such overloading of a simple hardware user interface is far from ideal, and requires a chart of what the patterns of LEDs mean to be kept handy. Various more complicated schemes for solving the user interface conundrum in Dead Man's Catch have been mooted - they may or may not be implemented for the production version of the firmware. For now, you will just need to refer to the table below, very often. I suggest that you print it out.

Playing Dead Man's Catch

After installing Dead Man's Catch v0.7-beta on your Peaks module (see below for instructions), the added capabilities are accessible as via four "pages" of alternative functions. Each of these four pages can be accessed in sequence by a single long-press (1 second or more) on the FUNCTION button for one second. In the basic functions, only one of the LEDs labelled ENV, LFO, TAP or DRUM will be steadily lit at any one time. However, in each of the four alternative function pages, one LED will be steadily lit, and one or more of the other three FUNCTION LEDs will blink. The pattern of these blinking LEDs (not the pattern of blinks, but which LEDs are blinking) indicates which of the alternative functions for that page the module is currently in. Short-presses on the FUNCTION button will cycle through the alternative functions for that alternative function page, in the usual fashion. To move to the next alternative function page, long-press the FUNCTION button. The module remembers which function you were last on when you re-enter an alternative function page, although it forgets this when it is powered off. However, the last function page and function it was in (for each channel) *is* remembered before powering off, and it will come back up in those functions when next powered on.

All this becomes will become clearer after studying the table which follows.

In Dead Man's Catch v0.7-beta, the alternative function pages and the alternative functions on each page appear in the following order, indicated by the LED pattern shown in the table (◉ means a steadily-lit LED, ☀ means a flashing LED, and ◎ means an unlit LED, listed left-to-right below as the LEDs appear from top-to-bottom on the Peaks panel e.g. ◉ ◎ ◎ ◎ means only the ENV LED is lit steadily, ◉ ◎ ☀ ◎ means that the ENV LED is lit steadily and the TAP LEDs flashes, and so on). Note that functions marked with an asterisk* are unchanged from the standard Peaks firmware. All the rest are additional functions in Dead Man's Catch:

Function page	Function	LED pattern (from top to bottom)
Basic	Envelope*	◉ ◎ ◎ ◎
	LFO*	◎ ◉ ◎ ◎
	Tap LFO*	◎ ◎ ◉ ◎
	Bass/Snare Drums*	◎ ◎ ◎ ◉
Alternative ENV	Double attack envelope	◉ ☀ ◎ ◎
	Repeating attack envelope	◉ ◎ ☀ ◎
	Looping envelope	◉ ◎ ◎ ☀
	Randomised AD envelope	◉ ☀ ☀ ◎
	Bouncing ball envelope	◉ ◎ ☀ ☀
Alternative LFO	Frequency-modulated LFO (folded sine FM)	◎ ◉ ☀ ◎
	Frequency-modulated LFO (random FM)	☀ ◉ ☀ ◎
	Varying wave shape LFO (folded sine WS mod)	◎ ◉ ◎ ☀
	Varying wave shape LFO (random WS mod)	☀ ◉ ◎ ☀
	Phase-locked loop oscillator	◎ ◉ ☀ ☀

Alternative TAP	Mini-sequencer*	☀ ◎ ◉ ◎
	ModSequencer	◎ ☀ ◉ ◎
	Pulse Shaper	◎ ◎ ◉ ☀
	Pulse Randomiser	◎ ☀ ◉ ☀
	Turing Machine	☀ ◎ ◉ ☀
	Bytebeats	☀ ☀ ◉ ☀
Alternative DRUM	FM drum Generators*	☀ ◎ ◎ ◉
	Randomised bass and snare drum generators	◎ ◎ ☀ ◉
	Randomised high hat	◎ ☀ ◎ ◉

To return to the 4 basic functions, hold down the FUNCTION button for more than 1 second, one more time, and the function page will cycle back to the basic functions.

Controls for the additional alternative functions

Functions in the table above marked with an asterisk (*) are the same as in the standard Peaks firmware and behave exactly as documented in the official Peaks manual, with one exception: there is no high hat model accessed from the basic snare drum function at the extreme of the knobs 2 and 3 ranges. Instead, high hats have their own function.

Double attack envelope controls

In **TWIN** and **EXPERT** modes, the double attack envelope implements an ADSAR envelope, which behaves exactly like the ADSR envelope, except that an attack phase is repeated when the gate signal falls - that is, attack phases are triggered by **both** a rising gate **and** a falling gate signal. The four knobs, from top to bottom, set the duration of the attack phases (sic), the duration of the decay phase, the sustain level, and the duration of the release phase, respectively.

Like the ADSR envelope in the basic function mode, the attack phases use a quartic transfer function, and the decay and release phases have exponential responses. Quartic functions can help avoid unpleasant "clicks" when short envelope segment times are used - so you can have a punchy attack segment without (or without so much) audible clicking. They also have the advantage of reaching a target value in finite time, whereas exponential curves never quite get to the desired target value.

In **SPLIT** mode, the double attack envelope function actually implements a simple AD (attack-decay) envelope, just as **SPLIT** mode does in the basic ENV function. However, unlike the basic ENV function, in which the attack and decay curves of the AD envelope are quartic and exponential respectively, in the double attack envelope function in **SPLIT** mode, both the attack and decay phases of the AD envelope have a linear shape. The controls are otherwise the same as **SPLIT** mode in the basic ENV function, with the knobs controlling, from top to bottom:

- Channel 1 attack time
- Channel 1 decay time
- Channel 2 attack time
- Channel 2 decay time

Repeating attack envelope controls

In **TWIN** and **EXPERT** modes, the repeating attack envelope implements an ADSR envelope in which the attack and decay phases are repeated as soon as the decay phase reaches the sustain level, as long as the gate signal remains high. Thus, while the gate is high, there is no sustain "plateau" - rather, the attack-decay phases are repeated in a series of peaks, and the sustain control merely sets the level at which the attack phase is retriggered. There is no refractory period*. As soon as the gate signal falls, the release phase is entered. The attack and decay phases use quartic curves, and the release phase uses an exponential curve.

Apart from the fact that the sustain plateau is replaced by a repeating series of AD peaks, the repeating attack envelope behaves just like the basic ENV function ADSR envelope. The four knobs, from top to bottom, set the duration of the attack phases (sic), the duration of the decay phases (sic), the sustain level, and the duration of the release phase, respectively.

In **SPLIT** mode, the repeating attack envelope function implements a simple AD (attack-decay) envelope, just as **SPLIT** mode does in the basic ENV function, with quartic curves for both the attack and decay phases. However, the attack and decay repeats for as long as the gate signal remains high. The controls are otherwise the same as **SPLIT** mode in the basic ENV function, with the knobs controlling, from top to bottom:

- Channel 1 attack time
- Channel 1 decay time
- Channel 2 attack time
- Channel 2 decay time

Looping envelope controls

In **TWIN** and **EXPERT** modes, the looping envelope implements an ADR envelope in which the release phase immediately follows the decay phase, as soon as the sustain level is reached. There is no sustain "plateau" - the sustain control merely sets the level at which the envelope transitions into the release phase - an inflection point, in fact. However, as soon as the ADR cycle is complete, it automatically retriggers itself, *ad infinitum* regardless of whether the gate input is high or not. However, a rising gate (or trigger) signal will immediately reset the cycle back to the start of the attack phase, and can thus be used to synchronise with clocks etc.

Apart from the fact that the sustain control merely sets the inflection point between the decay and release phases, the controls for the looping envelope behave just like the controls for the basic ENV function ADSR envelope. The four knobs, from top to bottom, set the duration of the attack phases, the duration of the decay phase, the sustain inflection point, and the duration of the release phase, respectively. All phases use a quartic transfer function.

In **SPLIT** mode, the looping envelope function implements a simple AD (attack-decay) envelope, just as **SPLIT** mode does in the basic ENV function, with quartic curves for both the attack and decay phases. However, the attack and decay cycle repeats *ad infinitum*. A rising gate/trigger signal will immediately reset the cycle back to the start of the attack phase. The controls are otherwise the same as **SPLIT** mode in the basic ENV function, with the knobs controlling, from top to bottom:

- Channel 1 attack time
- Channel 1 decay time
- Channel 2 attack time
- Channel 2 decay time

Randomised AD envelope

In **TWIN**, **EXPERT** and **SPLIT** modes, the randomised AD envelope implements (gosh!) an AD (attack-decay) envelope in which the decay phase immediately follows the attack phase, as soon as the peak of the attack amplitude is reached. A rising gate (or trigger) signal will immediately reset the cycle back to the start of the attack phase, and can thus be used to synchronise with clocks etc. This AD behaviour is exactly the same as the basic ENV function in **SPLIT** mode.

However, in **TWIN** and **EXPERT** modes, there are also controls which set the amount of random variation in the amplitude and decay time of the envelope. in these modes, the four knobs control, from top to bottom:

- Attack time
- Decay time
- Amount of random variation in the envelope maximum amplitude (i.e. amplitude at the peak of the attack phase)
- Amount of random variation in the length of the decay phase.

The controls are otherwise the same as **SPLIT** mode in the basic ENV function, with the knobs controlling, from top to bottom:

- Channel 1 attack time
- Channel 1 decay time
- Channel 2 attack time
- Channel 2 decay time

Bouncing Ball controls

Controls for the Bouncing Ball envelope generator are as follows.

In **TWIN** and **EXPERT** modes, the knobs from top to bottom control:

- Gravity - increasing values **reduce** the acceleration due to gravity, so that twisting the knob clockwise simulates a bouncing ball on the Moon, or an asteroid, and turning it counter-clockwise simulates a bouncing ball on Kepler-452b.
- Bounce energy loss - how much energy is dissipated as heat each time the ball bounces. Increasing values **reduce** the energy loss, and at high values, the ball just bounces forever. Think of it as how hard a basketball is pumped up. At low values, the ball is soft and flat, and hardly bounces at all. Higher values equate to a progressively higher air pressure inside the ball, and thus a bouncier ball.
- Initial amplitude - the height from which the ball is thrown upwards - similar to throwing the ball upwards from progressively higher levels of the Eiffel Tower. Turning the control clockwise **reduces** the initial amplitude.
- Initial velocity - how hard the ball is thrown upwards.

In **SPLIT** mode, the knobs from top to bottom control:

- Channel 1 initial velocity **and** gravity
- Channel 1 bounce energy loss
- Channel 2 initial velocity **and** gravity
- Channel 2 bounce energy loss

Initial amplitude is set to minimum (i.e same as knob fully clockwise) and cannot be adjusted in **SPLIT** mode.

Frequency-modulated LFO (folded sine modulation)

This function is the same as the basic LFO function, except that there is an internal foldable sine wave modulator that can perform frequency modulation on the LFO frequency.

In **TWIN** and **EXPERT** modes, the knobs from top to bottom control:

- LFO base frequency
- LFO waveform/waveshape preset (sine, triangle, sawtooth, square, stepped triangle and random/noise in clockwise sequence)
- frequency of LFO frequency modulator
- depth of frequency modulation (zero depth at 12 o'clock position)

The depth of frequency modulation knob provides zero FM depth at the 12 o'clock position. Turning it counter-clockwise (left) from the 12 o'clock position provides increasing FM using a pure sine wave for modulation. Turning it clockwise (right) from 12 o'clock provides increasing depth of modulation with a folded sine wave, that has several "wiggles" in its waveform (wake up, Jeff!).

In **SPLIT** mode, the knobs from top to bottom control:

- channel 1 LFO base frequency
- channel 1 LFO waveform/waveshape preset (sine, triangle, sawtooth, square, stepped triangle and random/noise in clockwise sequence)
- channel 2 LFO base frequency
- channel 2 LFO waveform/waveshape preset (sine, triangle, sawtooth, square, stepped triangle and random/noise in clockwise sequence)

Note that the frequency modulation settings established in **TWIN** and **EXPERT** modes continue to apply in **SPLIT** mode.

Triggers or gates received on the trigger input will reset the phase of *both* the LFO and its internal modulator.

Frequency-modulated LFO (random modulation)

This is identical to the frequency-modulated LFO (folded sine modulation) mode described immediately above, except that random values are used to modulate the frequency, rather than a sine wave. The random values are periodically sampled at a rate set by the modulation frequency knob (knob 3) and that value is used for the duration of the sampling cycle. Turning the modulation depth (knob 4) counter-clockwise from 12 o/clock causes increasing depth of frequency modulation, with linear interpolation between random values, while turning it clockwise from 12 o'clock causes increasing depth of random frequency modulation, but with a smoother cosine interpolation between random modulation values.

Varying wave shape (waveshape-modulated) LFO (folded sine wave modulation)

This function is the same as the basic LFO function, except that there is an internal folded sine wave modulator that can perform waveshape modulation on the LFO waveform. The waveforms provided are also slightly different.

The standard and tap LFO modes, and the two FM LFO modes, offer these waveforms:

- folded/power-folded sine
- sawtooth
- triangle
- square
- stepped triangle
- unfiltered noise
- smoothed noise

However, the wave shape-modulated LFO modes, and the PLO mode (see below) off there waveforms:

- folded sine
- power-folded sine
- overdriven sine
- triangle/sawtooth/ramp
- square (with pulse-width settable)

In **TWIN** and **EXPERT** modes, the knobs from top to bottom control:

- LFO base frequency
- LFO waveform/waveshape preset (sine, triangle, sawtooth, square, stepped triangle and random/noise in clockwise sequence)
- frequency of wave shape modulator
- depth of wave shape modulation

The depth of wave shape modulation knob provides zero wave shape modulation depth at the 12 o'clock position. Turning it counter-clockwise (left) from the 12 o'clock position provides increasing wave shape modulation using a pure sine wave for modulation. Turning it clockwise (right) from 12 o'clock provides increasing depth of wave shape modulation with a folded sine wave, that has several "wiggles" in its waveform (stop sleeping Jeff!).

In **SPLIT** mode, the knobs from top to bottom control:

- channel 1 LFO base frequency
- channel 1 LFO waveform/waveshape preset (sine, triangle, sawtooth, square, stepped triangle and random/noise in clockwise sequence)
- channel 2 LFO base frequency
- channel 2 LFO waveform/waveshape preset (sine, triangle, sawtooth, square, stepped triangle and random/noise in clockwise sequence)

Note that the wave shape modulation settings established in **TWIN** and **EXPERT** modes continue to apply in **SPLIT** mode.

Triggers or gates received on the trigger input will reset the phase of *both* the LFO and its internal modulator.

Wave shape modulated LFO (random modulation)

This is identical to the wave shape-modulated LFO (folded sine modulation) mode described immediately above, except that random values are used to modulate the wave shape, rather than a sine wave. The random values are periodically sampled at a rate set by the modulation frequency knob (knob 3) and that value is used for the duration of the sampling cycle. Turning the modulation depth (knob 4) counter-clockwise from 12 o/clock causes increasing depth of wave shape modulation, with linear interpolation between random values, while turning it clockwise from 12 o'clock causes increasing depth of random wave shape modulation, but with a smoother cosine interpolation between random modulation values.

Phase-locked loop oscillator (PLO)

To use PLO mode, patch an external oscillator signal into the trigger 1 and/or trigger 2 inputs on your Peaks. Peaks will then attempt to lock on to and follow the frequency of the audio signals provided at these inputs. This is just like tap LFO mode, except that it is optimised for use at audio frequencies. Note that two, independent PLOs are provided, one on each channel of Peaks - you can feed different audio signals into each channel and it will independently track the frequency of each.

Note that although it does its best, it will not track every possible audio signal perfectly. Clean, regular waveforms straight out of an oscillator module work best - don't expect it to track music or a heavy filtered or modulated waveform. However, you can try it with you singing "oohs" and "aaahs" into a microphone. However, mostly you need to feed it with signal directly from the oscillator you want it to track, *before* you filter or otherwise mangle that oscillator's signal (send the oscillator signal through a "multiple" to Peaks and to your filter etc). With knob 1 on Peaks (set to PLO mode) at just past the 12 o'clock position, you should hear an audio signal on the corresponding output from Peaks at exactly the same frequency as the input signal. Now move knob 1 counter-clockwise slightly. The output signal will drop an octave, but will still track the frequency of the input signal. Yup, a sub-oscillator! It will go down 1, 2 , 3 or 4 octaves below the input signal. Turning knob 1 clockwise increases the frequency by 1, 2 or 3 octaves above that of the input signal.

Knob 2 sets the waveform - the same ones as listed for the wave shape-modulated LFO mode described above. Knob 3 sets the rate of the internal sine-wave wave shape modulator, and knob 4 sets the depth of the wave shape modulation. That's it!

Note that although PLO mode tracks frequency, it doest reproduce the *phase* of the signal it produces doesn't match the phase of the input signal, and thus you can use it to "thicken" or "fatten" the sound of a single oscillator in various way by mixing the original input signal with the output of PLO mode.

There is also some audible "slew" as the PLO tracks frequencies from one note to the next. This is because it takes a fraction of a second for the phase-lock to settle on the new frequency. The audible effect is similar to the somewhat wayward portamento function on a Minimoog or Roland System 100 synth from 1975, and is thus part of the charm of this mode.

More formally, in **TWIN** and **EXPERT** modes, the knobs from top to bottom control:

- frequency divider/multiplier (sub-oscillator from 4 octaves below, in octave steps to a super-oscillator 3 octaves above the frequency of the input signal)
- LFO waveform/waveshape preset
- frequency of internal wave shape modulator
- depth of wave shape modulation

In **SPLIT** mode, the knobs from top to bottom control:

- channel 1 sub-oscillator/super-oscillator frequency division/multiplication ratio (in octaves)
- channel 1 LFO waveform/waveshape preset
- channel 1 sub-oscillator/super-oscillator frequency division/multiplication ratio (in octaves)
- channel 2 LFO waveform/waveshape preset

Note that the wave shape modulation settings established in **TWIN** and **EXPERT** modes continue to apply in **SPLIT** mode.

Triggers or gates received on the trigger input will *NOT* reset the phase of *either* the LFO nor its internal modulator. The reason for this is that in PLO mode, the trigger inputs should be used to feed the module audio-rate oscillator signals whose frequency you want it to track.

Turing Machine controls

If you are unfamiliar with the way in which a hardware Music Thing Turing Machine works, it is strongly recommended that you read Tom Whitwell's explanation so that the following explanations make more sense. The Turing Machine implementation in Dead Man's Catch is not an exact recreation of Tom Whitwell's hardware Turing Machine module, but it is inspired by it.

Two independent Turing Machines are available, one on each of Peaks' two channels. Triggers received on the TRIG1 input advances the Turing Machine 1 shift register by one step (subject to the Clock Division ratio, see below), and triggers received on the TRIG2 input advances the Turing Machine 2 shift register. Each Turing Machine outputs a voltage on the OUT1 and OUT2 outputs, respectively. The value of the output voltage is indicated (approximately) by the brightness of the TRIG1 and TRIG2 illuminated push buttons.

In **EXPERT** mode, you can, of course, use a Turing Machine on just one of the Peaks two channels, and some other function on the other channel.

Controls for the Turing Machines are as follows.

In **TWIN** and **EXPERT** modes, the knobs from top to bottom control:

- Probability - sets the probability (*p*) that a bit in the LSB (least significant bit) position of the shift register will be flipped. The scaling on the potentiometer is from *p*=0 at the extreme counterclockwise position, up to *p*=0.125 at about the 4 o/clock position. Turning the knob all the way clockwise (to 5 o'clock) forces the probability to 1.0, (that is, every bit is flipped at every step), thus turning the shift register into a Moebius strip and effectively doubling its length (to a maximum of 64 steps). In general, low settings work best, and the shorter the length of the shift register, as set by the Length knob (the 3rd knob), the lower the Probability setting needs to be in order to maintain a given rate (strictly, expectation) of a bit being flipped in any arbitrary time period. Remember, the probability is tested at every step of the shift register, so if you only want a bit to be flipped every few cycles of the register, you will need a quite low per-shift probability. In general, low probabilities are the sweet spot.
- Span - sets the span (or range) of CV output by the Turing Machine, from a minimum of 0 volts to 0 volts (that is, no change at all) up to a span of about 0 to 5 volts. A low span compresses the range of the control voltage output. Note that the voltage is unquantised, and thus for best effect when used to control pitch of an oscillator, you really need to feed it through a voltage (note) quantiser (such as one of these, or these), or use it with an oscillator module that has built-in note quantisation, such as Braids (or Braids running Bees-in-the-Trees alternative firmware for even more quantisation options). Smaller spans create a more restrained, sombre feel to the sequence, while longer spans sound livelier and more bouncy, but it is easy to go too far with the span. Very short note spans can be effective when used with heavy timbral and/or colour modulation.
- Length - sets the length of the shift register, in 4 bit steps from 8 bits at the extreme anticlockwise position, to 32 bits at the extreme clockwise position. The length of the shift register in bits controls the number of steps before the pattern loops around and repeats (subject to random change as set by the Probability pot).
- Clock Division - sets the clock division ratio for the incoming trigger (clock), from a minimum of 1 at the extreme anti-clockwise position, to a maximum of 8 at the extreme clockwise position. Thus, when set to 1, the Turing Machine shift register will advance one step every time a trigger is received. If set to 2, then the shift register only advances every second trigger, and so on.

In **SPLIT** mode, the knobs from top to bottom control:

- Channel 1 Probability
- Channel 1 Span
- Channel 2 Probability
- Channel 2 Span

Note that in **SPLIT** mode, Length defaults to 8 and Clock Division to 1, but if you set these in **TWIN** or **EXPERT** modes, then whatever you set for Length and Clock Division will be retained in **SPLIT** mode.

ModSequencer

The ModSequencer alternative function works in a very similar way to the DinSync MODSEQ module. Under the bonnet, it is just a very simple extension to the existing Peaks mini-sequencer mode and works in a similar fashion. Thus, in **TWIN** and **EXPERT** modes, knobs 1 to 4 set the output voltages of steps 1 to 4 of a sequence. The active step is advanced by a trigger received on trigger input or 2 respectively (or by pashes on the trigger buttons). However, instead of there being just 4 steps, there are 8! The voltages output by steps 5 to 8 are actually the complements of the voltages output by steps 1 to 4, respectively. Thus, say that step 1 is set to output 6 volts (out of a possible range of about -8 volts to +8 volts), then step 5 will output -6 volts. If step 2 is set to output -1.5V, then step 6 will output about 1.5V, and so on.

In **SPLIT** mode, there are 4 steps in each sequence. Knobs 1 and 2 set the voltage output levels for steps 1 and 2, and the complement of the levels for steps 3 and 4, for channel 1. Knobs 3 and 4 set the output levels for steps 1 and 2, and the complements of the output levels for steps 3 and 4 of channel 2.

The trigger inputs behave the same way that they do in the mini-sequencer alternative mode. That is, as it says in the official peaks manual, in **TWIN** and **EXPERT** modes: "*Channel 1 is clocked by TRIG 1 and reset by TRIG 2. Channel 2 is clocked by TRIG 2. _" In **SPLIT** mode: "_There is no reset, and each channel has its clock.*"

Bytebeats

This function makes eight different "bytebeat" equation oscillators available in each of the two Peaks channels. For background information on bytebeats, see (this page) [http://canonical.org/~kragen/bytebeat/] by *kragen*. The sources and author attributions for the eight bytebeat models used are available in the source code.

In ***TWIN** and **EXPERT** modes, the knobs control the following parameters, from top to bottom:

- "pitch", sort of
- adjustable parameter 0 (zero). What this does depends on the equation in use and the settings of the other knobs...
- adjustable parameter 1 . What this does depends on the equation in use and the settings of the other knobs...
- equation selector - eight different equations are available

The trigger inputs (and trigger buttons) reset the internal counters for each byte beat equation back to zero.

Note that these bytebeat equations play "tunes", or sequences of tones, or noise, and thus "pitch" should be considered more an offset or transposition of the "tune": the pitch control affects the speed at which the time accumulator advances, and thus affects both loop speed and the pitch of the "notes" played.

Be warned, the output of these bytebeat equations can be very harsh - rather like a Merzbow composition! You will almost certainly want to filter their outputs through a low-pass filter to remove some of the high-frequency digital noise they can produce - but appropriately filtered, they can sound rather nice too. Use of formant filters or resonant band equalisers is particularly effective with them. There is a large degree of interaction between the settings for these bytebeat models, with many combinations of settings producing harsh grating noise, or complete silence, but there are islands of beauty, or at least interest, amidst all the noise and digital grit.

Tip: if channels 1 and 2 are panned left and right in the stereo image, then some interesting phasing and spatial effects can be achieved by slightly offsetting the internal clocks of the bytebeat models on each of the two channels. This can be done by using two finger to push the trigger buttons on your Peaks, at not quite the same time - maybe 50 or 100 milliseconds apart (or more - experiment!).

Randomised bass and snare drum generators

These are versions of the bass and snare drum models which are available as the normal DRUM functions in Peaks. The difference is that there are controls which set the degree of random variation in the drum sounds from beat-to-beat. As in the normal drums, channel 1 is a bass drum and channel 2 is a snare drum. However, the controls are a bit different.

In **TWIN**, **EXPERT** and **SPLIT** modes, knob 1 sets the pitch of the base drum, and knob 2 sets its decay time and its tone. For the snare drum, in **TWIN**, **EXPERT** and **SPLIT** modes, knob 1 sets pitch and knob 2 sets the "snap" of the snare (the high frequency content). In other words, knobs 1 and 2 control the same parameters as they do for the base and snare drums in **SPLIT** mode in the normal drum function.

However, for the randomised drums, in **TWIN** and **EXPERT** modes, knob 3 sets the degree of random variation in the pitch of the drum, and knob 4 sets the random variation in the amplitude of the "hit". For the base drum, this "hit amplitude" affects the length of the decay and the overall signal level (harder hits are louder). For the snare drum, the amplitude of the "hit" affects the tone, decay time of the snare noise, and the overall loudness.

If knobs 3 and 4 are set to their minimum positions (extreme counterclockwise), all randomisation is disabled. Increasing the settings on the knobs increases the degree of randomisation. The pitch is randomised using a simple first-order random walk, and the randomisation setting affects the length of each "stride". The "hit amplitude" randomisation is also a random walk, but the length of each step (up or down) is also random: knob 4 sets the scaling of these random step lengths.

Note that although the randomisation settings, as set by knobs 3 and 4 when in **TWIN** or **EXPERT** modes, cannot be altered when in **SPLIT** mode, the randomisation is still active. To adjust it, you need to drop out of **SPLIT** mode back into **TWIN** or **EXPERT** modes.

In summary, in **TWIN** and **EXPERT** modes, the knobs, from top to bottom, are mapped as follows:

- Drum pitch
- Drum tone/decay time
- Pitch randomisation
- Hit amplitude randomisation

In **SPLIT** mode, the controls are as follows:

- Bass drum pitch
- Bass drum tone/decay time
- Snare drum pitch
- Snare drum decay/"snap"

What is the purpose of these drum randomisation functions? Well, robotic or motorik drumming has its place (who can fail to like Can, Neu! or Stereolab), but sometimes it is nice to inject a bit of random variation into the drum sounds. Swing, provided by whatever is triggering the Peaks drums sounds, certainly helps, but these randomised drum functions go further than just random variation in timing: they add random variation to the nature of the drum sounds themselves. The same can be achieved by random or semi-random modulation of the parameters of a voltage-controlled drum module, but Peaks isn't voltage controlled, and thus these randomised drum functions are the next best thing.

Randomised High hats

This function implements a pair of high hats, one on each channel. The high hat model is almost the same as the one available within the snare drum function in the standard Peaks firmware (note: in Dead Man's Catch, the high hat model has been removed from the snare drum function). The main difference is that the hight hat model in Dead Man's Catch does not run at twice the sample rate, as it does in the factory firmware, and thus sounds a bit different (and there may be some clicks or pops evident - let me know if there are). However, in Dead Man's Catch, unlike the standard firmware, the high hat model is adjustable!

In **TWIN** and *EXPERT** modes, knob 1 controls the pitch (or maybe, the diameter?) of the high hat, and knob 2 controls the decay time (equating perhaps to the thickness of the high hat material?). Knobs 3 and 4 set the degree of randomisation of the "pitch" and decay time settings, using a similar random walk scheme as used for the Randomised Bass Drum and Randomised Snare Drum (see above).

In **SPLIT** mode, knobs 1 and 2 control the "pitch" and "decay" for channel 1, and knobs 3 and 4 control "pitch" and "decay" for channel 2. Note that randomisation is still active, but to change the randomisation settings, *TWIN** or **EXPERT** mode must be used.

The other feature of the high hats is that the high hat on channel 1 is deemed to be "closed", and the high hat on channel 2 is deemed to be "open" (whether the settings you have for the pitch and decay on each channel reflect these descriptions is up to you). The key thing is that a trigger or gate input on channel 2 will inhibit a simultaneous trigger or gate on channel 1 from firing the channel 1 high hat. This means that you can patch, say, the trigger from a Grids channel into input 1, and the accent from the same Grids channel into input 2, and whenever Grids emits an accent trigger simultaneously with the main trigger, only channel 2 of the peaks will fire - thus giving you open/closed high hats (or any variation on that theme). This is similar to the action of the "Exclusive" switch on the HexInverter Mutant Hihats module.

Installation

First of all, all of the **warnings** set out in the Bees-in-the-Trees documentation, here, also apply to Dead Man's Catch firmware for Peaks. *Please* read these warnings and pay heed to them. This is beta firmware, and in any case, no warranty or guarantee that it will work correctly is offered. If you are in doubt, do **not** install Dead Man's Catch!

If, after reading all the warnings, you still wish to proceed, then before attempting to install Dead Man's Catch, it is **strongly** recommended that you practice the firmware update procedure and establish that it works correctly by re-installing the official factory Peaks firmware. You can download the latest official peaks firmware from the Mutable Instruments official firmware repository (just scroll down the list to find the latest version of Peaks firmware, which will be version 1.0 or higher). Now just follow the instructions in the *Firmware update procedure* section of the official Peaks manual.

Once you have established that you are able to install the factory Peaks firmware successfully, just follow the same procedure but using the Dead Man's Catch firmware WAV file that you download below, rather than the audio firmware file for the official factory Peaks firmware. You will need to unzip the file before using it, using whatever unzip utility you usually use.

De-installation

If you want to revert to the official Peaks firmware, download the WAV file for the latest official release for Peaks from the Mutable Instruments official firmware repository, and follow the same firmware update procedure, as described in the *Firmware update procedure* section of the official Peaks manual, using the official Peaks firmware WAV file. Note that it is **strongly** recommended that you set your Peaks to standard ENV function in the base (non-alternative) mode, on **both** channels, before trying to re-install the official Peaks firmware. If you don't do that, it shouldn't matter, but it is a wise precaution.

Feedback

If you encounter any bugs, please report them to Tim Churches at tim.churches@gmail.com providing as much detail about how to reproduce the problem as possible. Buggy behaviour that can't be reproduced can't be fixed.

Future versions

More features will be added to Dead Man's Catch in due course, as time permits. There is no development or release timetable. Some ideas for what might be included can be found in the Mutated Mutables wiki. If you have a GitHub account, you may add your own suggestions and ideas to that wiki page, or you can email me at tim.churches@gmail.com, or discuss them on the Mutable Instruments forum, where they will be noted and recorded on the wiki page if they are a good idea and potentially feasible.

Feel free to fork or clone this repository, and git pull requests are welcome.

[*] Yes, that's a reference to Masters and Johnson.

Downloads

Dead-Mans-Catch-v0.7.0beta-2015-10-31.wav.zip 200 KB

Source code (zip)

Source code (tar.gz)