



BEAT KNITTER

www.fairlyconfusing.net

Licence and technical information

Beat Knitter is a virtual instrument for Native Instruments Kontakt sampler, you need full version of Kontakt 5.6.6 or newer to run it.

You are licensed to use this device and samples which come with it, in the creation of a recorded or live sound performance, free or commercial, without paying any additional license fees or providing source attribution.

Please DO NOT: include provided scripts and samples in any music library or sample library; sell, repackage or re-distribute the samples or sampler programs.

USE AT YOUR OWN RISK! This device is provided ‘as is’ and there is no warranty of any kind.

The Knitter

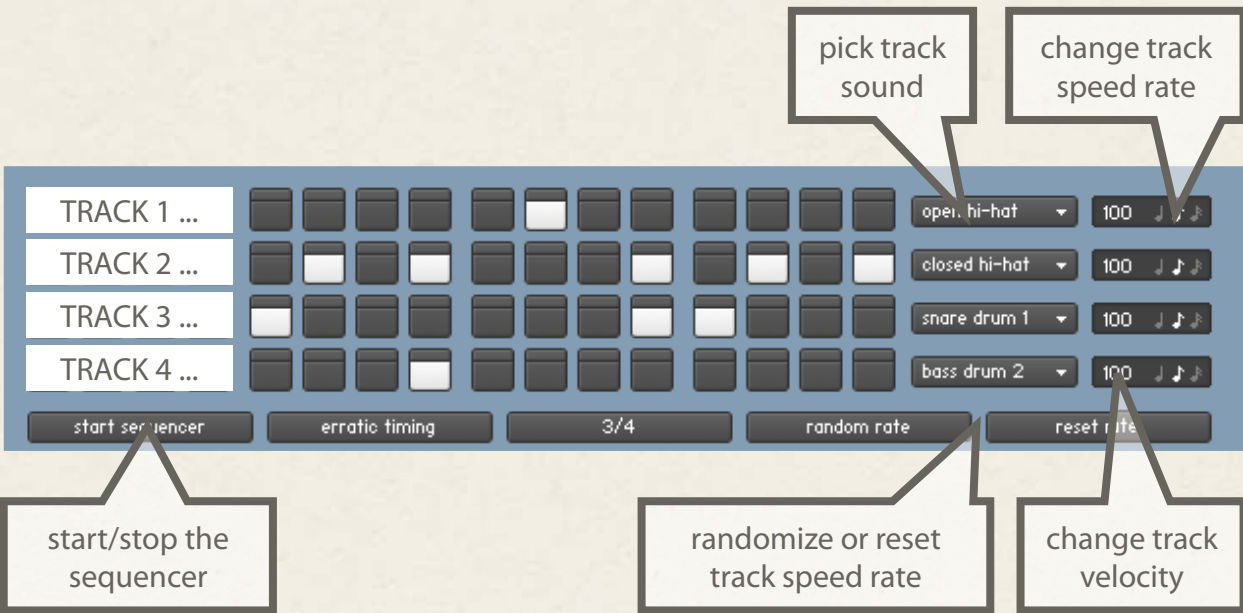
Beat Knitter is drum machine, sequencer and automated drum pattern generator.

The focus in this machine is on simplicity and experimenting with different random pattern generators. There is four track, sixteen step sequencer, which can be programmed by hand or automatically. The sound set is somewhat symbolic, but you can drag and drop current pattern into a MIDI file, or directly to a DAW to use with another drum machine or sample set.



Sequencer

This is a simple sequencer, you can assign different sound and one fixed velocity for each track. Tracks can be run at different speed rates, sixteenth note, eighth note or quarter note per step. 3/4 switch will simply trim off last four steps of each track, it has no impact on note generators or transformation functions – last four notes will simply remain hidden.



Pattern functions

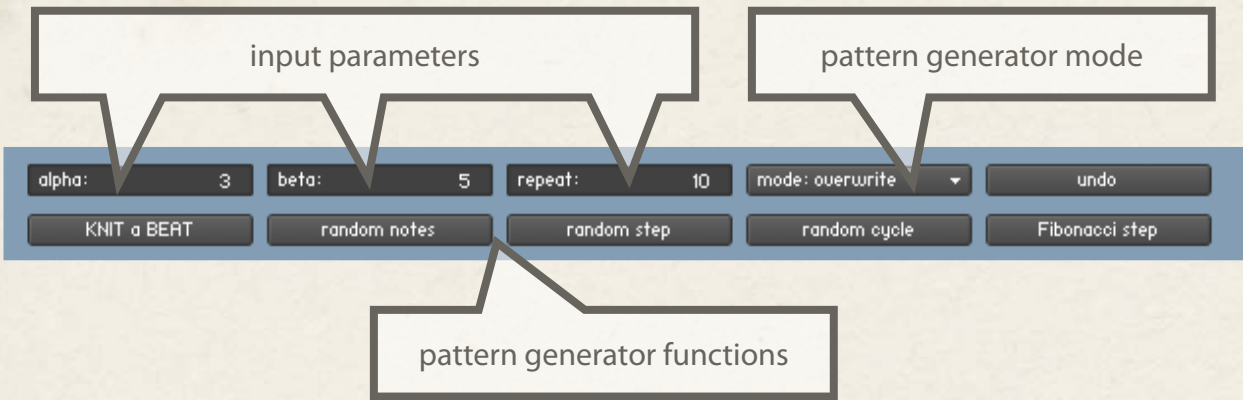
There are several functions, that can be used to generate or alter drum pattern. Three of input parameter can be used depending on function: alpha, beta and repeat. There is two-level undo button, press it to undo last operation, press it again to undo previous operation. Pressing it one more time will revert to result of last performed function.

Functions can apply to all tracks, only track 1 and 2 (when ‘write top sequences’ switch is off) or only tracks 3 and 4 (when ‘write bottom sequences’ switch is off). These limiting switches do not work on ‘twister’ function and undo function, which will always recall state of all four tracks.

Pattern generators

Pattern generators are functions adding new notes. They can work in four modes:

- **overwrite** (clear whole pattern and then add new notes),
- **add** (add new notes over existing pattern),
- **invert** (flip step state instead of adding a note),
- **clear** (remove notes instead of adding them to pattern).



KNIT a BEAT

parameters used: none

This is the main pattern generator. It attempts to write a consistent random beat, assuming default sound configuration (track 1: open hat, track 2: closed hat, track 3: snare, track 4: bass drum).

Random notes

parameters used: repeat

Simply add a random note. Repeat = how many times to repeat the function.

Random step

parameters used: alpha, beta

Start from beginning of track 1, add notes moving cursor forward by random step (alpha and beta stand for minimum and maximum of step length). By the end of track, cursor wraps around to next one. This function goes on until end of last track sequence is reached.

Example output for step range 2–3.



Random cycle

parameters used: alpha, beta

Similar to random step, but using the same step value for whole track. This function can randomly use two step cycle for a track, changing between two step values.

Example output for step range 2–3 (step 2 for first track and step sequence 2,3 for the other).

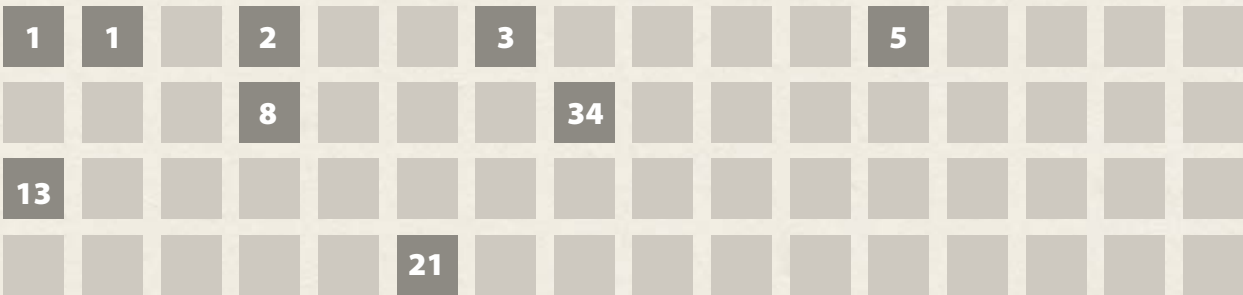


Fibonacci step

parameters used: alpha, beta, repeat

It's a small experiment. Using Fibonacci number sequence to determine cycle step (it's a sequence where a number is the sum of two preceding numbers in the sequence). Function is using alpha and beta as two starting numbers (these do not need to actually belong to Fibonacci sequence). Repeat determines, how many steps to take. Cursor wraps around sequencer range.

Example output for 1,1 as starting pair, with 9 notes generated. Step '34' has wrapped around.



Pattern transformations

Pattern transformation functions do not add new notes, but change pattern structure.

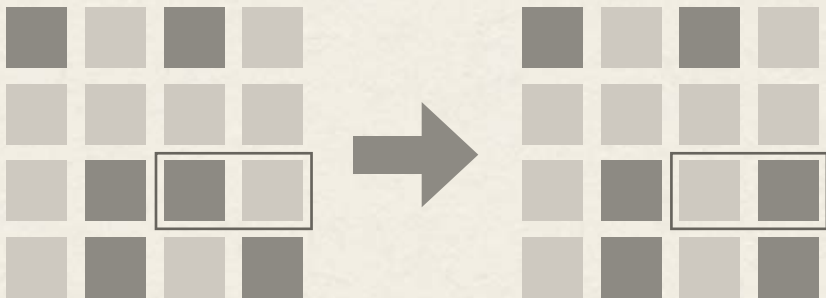


shuffle pattern

parameters used: repeat

This function will find random pair of neighbor steps with different content and swap them. It repeats defined number of times.

Example:

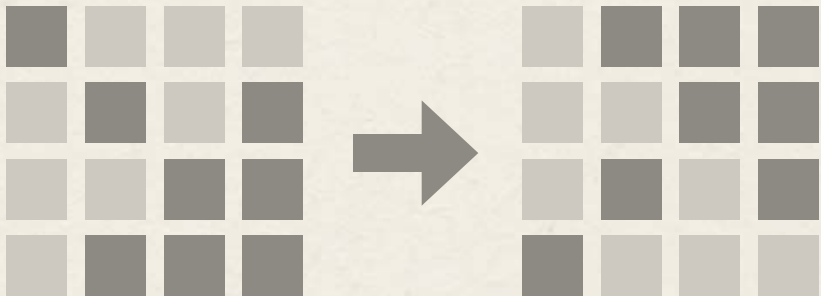


twister

parameters used: alpha

Splits pattern into 4x4 blocks and rotates them 40 degrees. When alpha = 2 all blocks are rotated clockwise, when alpha = 3 all blocks are rotated counter-clockwise. When alpha =1, rotation will be picked randomly (some blocks may remain as they were). This function ignores bottom/top switches, as it wouldn't make much sense then.

Example:

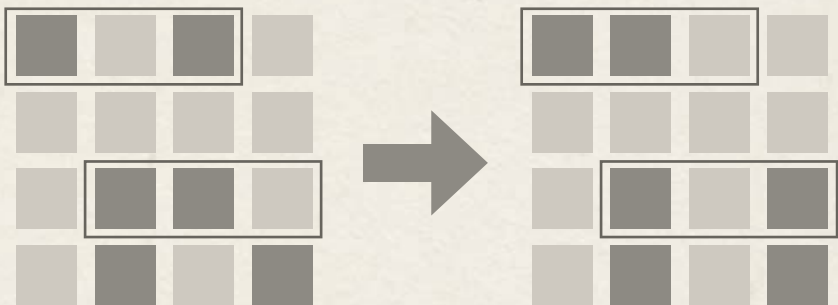


rearrange blocks

parameters used: alpha, beta, repeat

The function picks a random step string in a sequence and swaps it with another string positioned somewhere else in the sequencer grid. Alpha and beta stand for minimum and maximum length of the string (not more than 16 steps). It repeats defined number of times.

Example, 3 steps long block swap:

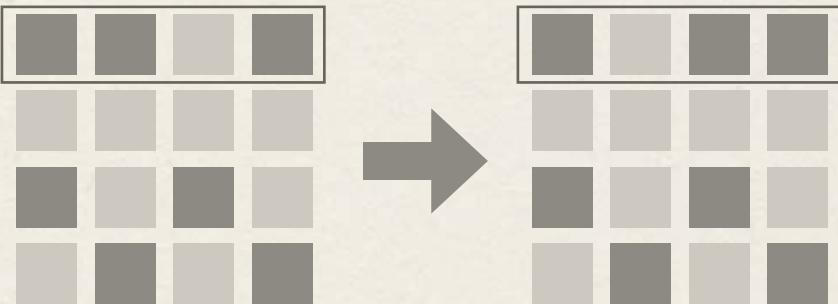


mirror blocks

parameters used: alpha, beta, repeat

The function picks a random step string in a sequence and mirrors it's contents. Alpha and beta stand for minimum and maximum length of the string (not more than 16 steps, not less than 2 steps). It repeats defined number of times.

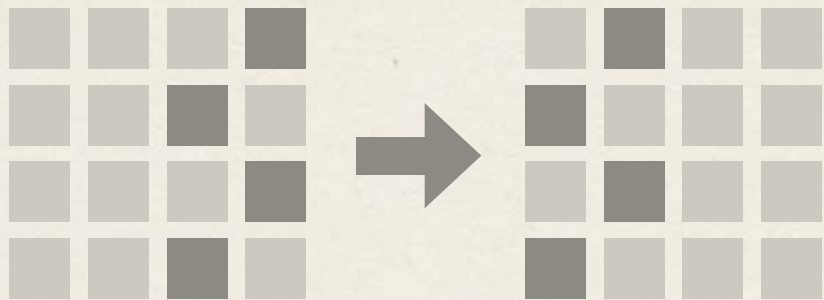
Example:



align
parameters used: none

Use to eliminate a gap at the beginning of sequence – make sure there is a note at first sequence step.

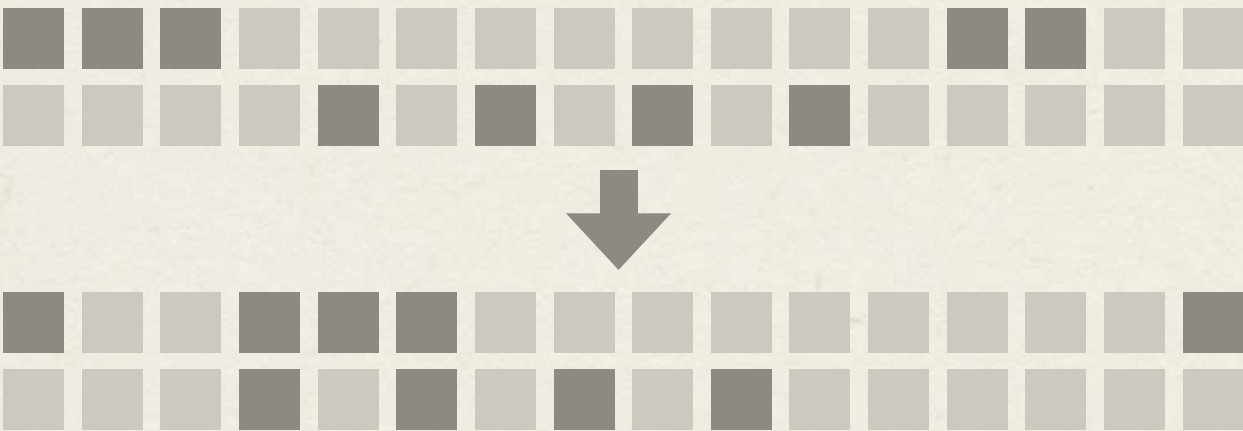
Example:



random offset
parameters used: none

Slides each track by random number of steps (contents wraps around track boundaries).

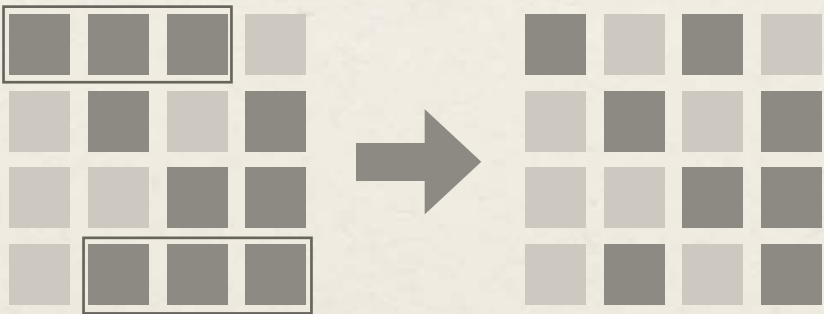
Example, first track +3, second track -1:



comb pattern
parameters used: none

Analyze all tracks, if there are three following ‘note on’ steps, set middle one ‘off’.

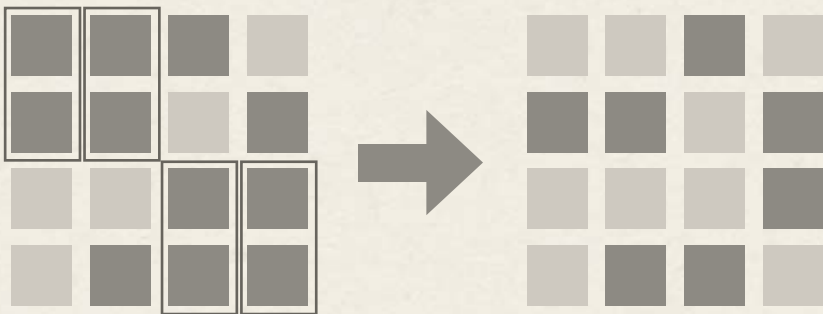
Example:



compact pattern
parameters used: none

Compares track 1 with track 2, if there are parallel ‘note on’ steps, it sets one of them ‘off’. The same procedure is repeated for tracks 3 and 4.

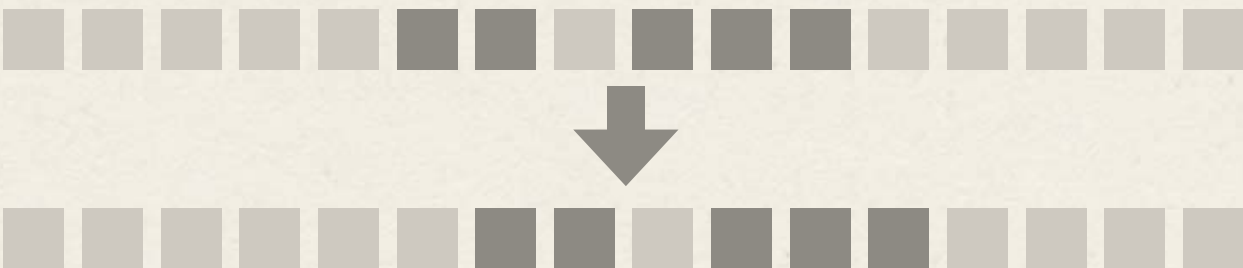
Example:



pull backward/push forward
parameters used: none

Slides each track one step forward of backward (contents wraps around track boundaries).

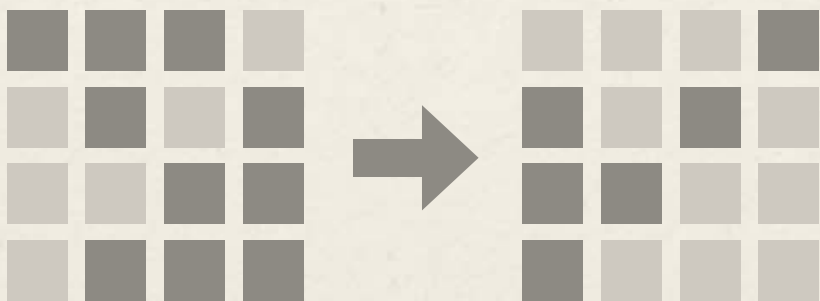
Example, push forward:



invert pattern
parameters used: none

Change ‘note on’ steps to empty steps and vice versa.

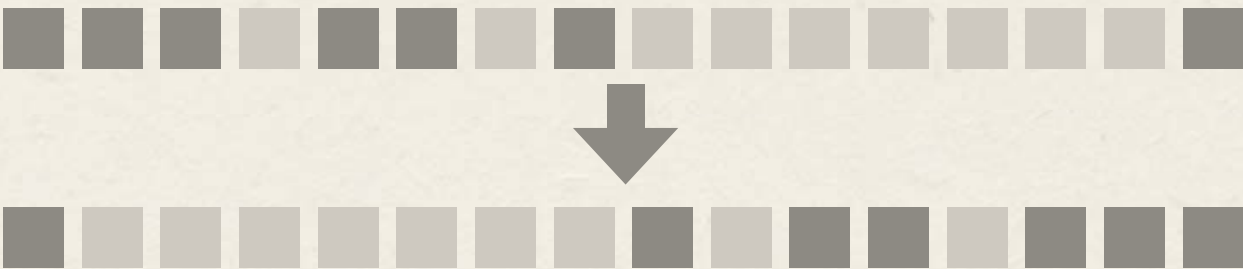
Example:



reverse pattern
parameters used: none

Puts track sequences in reversed order.

Example:



clear pattern
parameters used: none

Deletes all 'note on' steps.

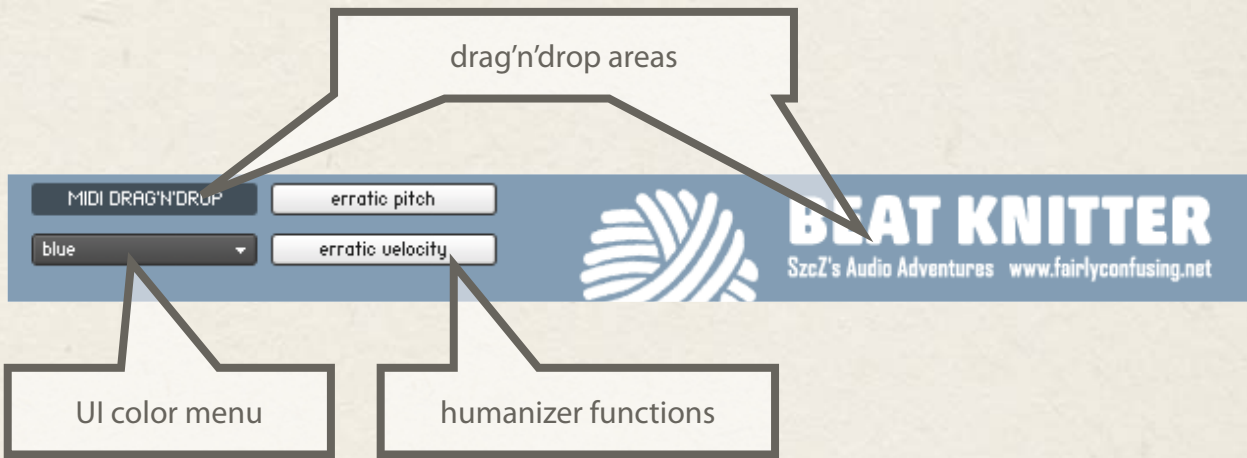
MIDI drag and drop

You can drag and drop programmed pattern directly to DAW. Click and hold on 'MIDI drag'n'drop' label or instrument's name. Note numbers will be assigned according to selected sound, as in GM standard.

Erratic timing/pitch/velocity switches can be used for 'humanizing' effect.

Erratic timing will make notes starting times not exactly spot on, this function affects only generated MIDI data. Erratic velocity will randomize notes velocity a bit. Erratic pitch will randomize played sound pitch a little bit, this function will only work on sounds played by sequencer (it does not affect generated MIDI data).

Finally, if blue doesn't suit you, there are several interface colors to pick from...



End of file

This document ends here. Have fun with Beat Knitter.