

I-Ching RND Plugin - User Manual

This plugin is a generative musical tool that combines randomization, hexagram-based sequencing, quantized pitch output, and selectable noise generation. It is designed for use on the Expert Sleepers disting NT platform and is highly configurable through a set of parameters.

This plugin is highly versatile and can be used for algorithmic melody generation, evolving pitch patterns, and noise-based modulation or percussion. Combine outputs for complex musical behavior or modulate parameters with CV for generative results.

Below is a detailed explanation of each parameter:

1. Clock In

- **Range:** 1 – 28
- **Default:** 1
- **Description:** Input channel for the master clock. This drives the hexagram sequence, clock divider, and synchronized outputs.

2. IntSeqTrigIn

- **Range:** 1 – 28
- **Default:** 2
- **Description:** Input trigger for advancing the integer sequence. It is separate from the main clock.

3. CV Out

- **Range:** 1 – 28
- **Default:** 13
- **Description:** Outputs a voltage-per-octave value derived directly from the current hexagram index (before quantization).

4. Quant Out

- **Range:** 1 – 28
- **Default:** 14
- **Description:** Outputs a quantized pitch value based on the hexagram index, processed through the selected scale, root, and transpose.

5. IntSeq Out

- **Range:** 1 – 28
- **Default:** 15
- **Description:** Outputs the pitch of the selected integer sequence, quantized to the active scale.

6. Noise Out

- **Range:** 1 – 28
- **Default:** 16
- **Description:** Outputs the selected type of noise (White, Pink, Brown, or Blue).

7. Clock Thru Out

- **Range:** 1 – 28
- **Default:** 17
- **Description:** Passes through the raw incoming clock signal.

8. Clock Div Out

- **Range:** 1 – 28
- **Default:** 18
- **Description:** Outputs a clock pulse every N clocks, where N is set by the Clock Div parameter.

9. Scale

- **Range:** 0 – 133 (scale names shown)
- **Default:** 0
- **Description:** Selects the musical scale used for quantization (e.g. Major, Minor, Pentatonic, etc.).

10. Root

- **Range:** 0 – 11
- **Default:** 0
- **Description:** Sets the root note (C=0, C#=1, ..., B=11) for the selected scale.

11. Transpose

- **Range:** -24 – 24 (semitones)
- **Default:** 0
- **Description:** Transposes the output pitch up or down by the specified number of semitones. QuantOut and IntSeqOut are influenced by this, IntSeqOut can behave weird.

12. MaskRot (Mask Rotate)

- **Range:** 0 – 15
- **Default:** 0
- **Description:** Rotates the internal mask used in quantization to create alternative pitch patterns.

13. IntSeq

- **Range:** 0 – 10 (Sequence name will be shown)
- **Default:** 0
- **Description:** Selects the integer sequence (e.g. PI, vanEck,ssdn,dress) used for the IntSeq Out.

14. IntSeqMod

- **Range:** 1 – 32
- **Default:** 1
- **Description:** Applies modulo to the integer sequence output, limiting its range. A value of 1 disables the modulo.

15. IntSeqStart

- **Range:** 0 – 126
- **Default:** 0
- **Description:** Index into the selected sequence from which playback starts.

16. IntSeqLen

- **Range:** 1 – 128
- **Default:** 16
- **Description:** Number of steps played from the starting index before the sequence loops.

17. IntSeqDir (Direction)

- **Range:** 0 (Forward), 1 (Ping-Pong)
- **Default:** 0
- **Description:** Sequence direction: forward through steps or ping-pong (back and forth).

18. IntSeqStride

- **Range:** 1 – 16
- **Default:** 1
- **Description:** Stride determines the number of indices skipped per step. Higher values create sparser or non-linear patterns.

19. Noise Type

- **Range:** 0 – 3
- **Default:** 0
- **Options:** White, Pink, Brown, Blue
- **Description:** Selects the type of noise output:
 - **White:** Flat spectrum
 - **Pink:** Warmer, 1/f filtered
 - **Brown:** Deep, integrated noise
 - **Blue:** High-passed, sharp noise

20. Clock Div

- **Range:** 2 – 512
- **Default:** 2
- **Description:** Number of incoming clock pulses before one pulse is sent out via Clock Div Out. Useful for creating slower rhythms.

Created by: Fabian Jodok Romualdo MARTINEZ

I-Ching_RND.cpp is a random value generator for the Disting NT module.

It generates random values based on the I Ching hexagrams and their associated values.

There is a quantized output for the generated values, which can be used in various musical applications and quantized with a variety of scales.

It has a built-in integer sequence generator that can be used to generate sequences of random values.

It has a noise output that can be used to generate random values based on noise. The noise type can be selected from a variety of options.

Every 64 clocks, you will have seen every hexagram exactly once, in a random order.

No repeats until all have been used.

After all 64, the order is reshuffled and the process repeats.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.