Noise Engineering — Integra Solum

- Manual PDF

Manual / Docs

Noise Engineering Integra Solum for dense, hyper-complex percussion

Integra Solum is not a sound source or effect; it's a **dual rotating clock divider / trigger generator**. That makes it extremely good for building **intricate drum logic**, especially when you want:

- **polyrhythms**
- **offset repeating structures**
- **odd-length cycles**
- **rotating accents**
- **controlled randomness**
- **two interlocked but independent rhythmic systems**

Because it has **two sides of 8 trigger outputs**, independently clockable/resettable, you can treat it like **two related rhythmic brains** in one module.

What it does best for percussion

From the manual, the key features for rhythm design are:

- **Dual sections**, each with **8 trigger outputs**
- Shared or separate **clock** and **reset**

- **Shift** control rotates the order of outputs
- Three core modes:
- **/2N** = powers-of-two divisions
- **N** = sequence of eight
- **/2N+1** = odd-number divisions
- **Wack mode** adds randomized behavior
- One clock can **normal to both sides**, or you can clock each side independently

That means Integra Solum is ideal as a **master trigger architecture module** for percussion systems built from drum voices, LPGs, envelopes, burst generators, logic, and VCAs.

Big-picture strategy

For hyper-complex percussion, think of Integra Solum as generating **layers of rhythmic hierarchy**:

1. **Main pulse layer**
   foundational kick/snare timing
2. **Subdivision layer**
   hats, shakers, clicks, ghost hits
3. **Accent rotation layer**
   changing emphasis over time via Shift
4. **Odd-cycle layer**
   polymeter and phrase drift via /2N+1
5. **Random disruption layer**
   Wack mode for instability and variation

Use one side for **stable structure**, and the other for **destabilization**.

A strong approach is:

- **Side A** = your “grid authority”
- **Side B** = your “mutation engine”

Understanding the three modes musically

1. /2N mode: powers-of-two divisions

This is your classic clock division territory.

Use it for:

- kicks on slower divisions
- snares/claps on medium divisions
- regular hats from faster divisions
- structured anchors in otherwise chaotic patches

This mode is useful when you want **clarity inside complexity**. Even if everything else gets strange, /2N gives the ear recognizable points of reference.

Good uses

- Kick on one slow output
- Snare on a medium division
- Closed hat on a faster division
- Accent trigger for opening a VCA or changing decay on another drum voice

Why it helps with dense rhythm

Dense music only works if some layer remains comprehensible. /2N provides that backbone.

---

2. N mode: sequence of eight

This is one of the most useful modes for evolving percussion. The manual describes it as a **sequence of eight**, and the Shift control rotates which jack is considered the first in the cycle.

Think of it like an **8-step trigger scan** spread across the outputs.

Use it for:

- stepping through 8 percussion voices
- moving accents around a smaller set of voices
- creating phrase-based rhythmic movement
- manually “rotating” patterns without re-patching

Great patch idea

Patch several outputs from one side in N mode to:

- kick trigger
- snare trigger
- closed hat
- open hat
- rim
- clap
- FM percussion hit
- noise burst

Now Shift changes which output happens first, so the whole drum phrase rotates. This is a very fast way to get **pattern mutation** while keeping a coherent phrase shape.

Why it helps with complex time signatures

Because the sequence is ordered and rotatable, you can align or misalign it against another clock domain or reset interval to create: - 5-over-8 feeling - 7-against-4 phrase movement - evolving downbeat displacement

---

3. /2N+1 mode: odd-number divisions

This is the goldmine for **polyrhythms and weird meters**.

Odd divisions naturally resist lining up with standard 4/4 subdivisions. That gives you:

- 3-based accents over 4-based kicks
- 5-step recurring percussion lines
- 7-hit phrase overlays
- long non-repeating composite structures

Best uses

- toms
- metallic percussion
- accents
- ratchet enable signals
- modulation triggers for drum parameter changes

Why this mode matters

If you want “complicated patterns” that don’t feel like simple x0x repeats, odd-number divisions are essential.

A patch with: - Side A in **/2N** - Side B in **/2N+1**

already gets you far into **polymetric percussion architecture**.

Wack mode: where complexity becomes alive

Wack mode turns Integra Solum from a structured divider into a **controlled chaos percussion sequencer**.

The manual gives these behaviors:

Wack /2N

- probabilistic divide-by-two behavior
- 50% chance of generating a trigger at each step
- average density similar to /2N, but randomized

This is ideal for: - ghost hats - unstable accents - broken techno percussion - swung-feeling textures without actual swing

Wack N

- a **single random trigger** is generated at each step

This is excellent for: - rotating one-hit-per-step percussion choice - “which voice fires now?” behavior - sparse but animated phrase motion

If patched to multiple drum voices, this becomes a kind of **random voice allocator**.

Wack /2N+1

- all 8 outputs independently have a 50% chance of going high on each rising clock

This is the densest and wildest mode. It can generate: - flurries of percussion - clustered impacts - glitch bursts - chaotic fills

This is perfect for: - hats through VCAs - noise bursts - FM clicks - transient layers - fill sections - modulating drum parameters at high density

---

Best practical patch architectures

Patch 1: Stable core + unstable detail

A classic high-functioning drum patch.

Side A

- Clocked by your master clock
- Mode: **/2N**
- Use outputs for:
- kick
- snare
- clap
- open hat accents

Side B

- Same clock, normalled from Side A
- Mode: **/2N+1** or Wack **/2N+1**
- Use outputs for:
- closed hats
- shakers
- blips
- metallic percussion
- fills

Result

Side A gives groove stability; Side B generates tension and complexity.

Patch 2: Two independent clock domains for true polyrhythm

Since each side can be clocked independently, use this.

Side A

- Clock: straight 16th-note clock
- Mode: **N**
- Handles your “main kit”

Side B

- Clock: a different clock rate, such as:
- triplets
- dotted 8ths
- clock multiplied x5 or x7
- a manually divided odd-rate pulse
- Mode: **/2N+1**

Result

The two sides phase against each other and create **real polyrhythm**, not just rotated variation.

This is one of the strongest uses of the module.

Patch 3: Complex time signatures with strategic reset

The reset input is crucial.

Use resets to define phrase length. For example:

- Let Side A reset every **16 steps**
- Let Side B reset every **15** or **14** or **21** steps

Now your rhythmic pattern re-aligns only after a longer cycle. That gives the effect of: - shifting meters - non-repeating bar structures - evolving phrase accents

Example feel

- Side A suggests 4/4
- Side B implies 5/4 or 7/8
- Together, the system creates long-form rhythmic evolution

Because the reset can be patched independently, you can use another sequencer, logic source, or clock divider to determine phrase boundaries.

Patch 4: Rotating drum orchestration

Put one side in **N** mode and patch each output to a different percussion voice.

Then perform with **Shift**.

Since Shift rotates the outputs, the order of events moves across the kit: - kick appears in different phrase positions - snares displace - metallic hits slide forward - accents migrate

This is excellent for: - live improvisation - pattern mutation without menu diving - generating “same pattern, new groove” effects

If your voices have overlapping timbres, this can sound like a highly composed rhythmic rearrangement rather than randomization.

Patch 5: Trigger one voice, modulate another

Don’t use all outputs only as drum triggers. Some should control **parameters**.

Patch outputs to: - envelope triggers for pitch sweeps - accent VCAs - decay CV sample-and-hold clocks - filter ping triggers - wavefolder CV

gates - sample-rate or bit-depth change triggers - switch/select modules to swap percussion timbres

Example

- One output triggers a kick
- Another output, on a different division, triggers a short envelope that momentarily increases kick pitch or attack
- Another output opens a VCA on noise layered with the snare

Now the rhythm system is not just deciding **when** things happen, but also **what kind** of hit happens.

That's how you get truly intricate percussion.

How to make it feel dense without becoming mush

Dense rhythmic music needs separation. Use Integra Solum to distribute roles.

Assign outputs by function

Instead of patching 8 outputs to 8 random drums, organize them like:

- 2 outputs = structural hits
- 2 outputs = accents
- 2 outputs = ghost notes
- 2 outputs = parameter modulation triggers

This keeps complexity legible.

Use different decay lengths

Even if two triggers are dense, they'll read clearly if one voice is: - short click - medium snare - long metallic ring

Reserve Wack mode for upper layers

If everything is randomized, the groove dissolves. Usually: - stable low-end - unstable mids/highs works best.

Use reset as composition

A reset is not just utility; it is a phrase-defining tool. Resetting one side at unusual lengths creates "meter" out of pure trigger logic.

Specific advanced rhythmic ideas

1. Fake euclidean-like behavior

While Integra Solum is not a Euclidean sequencer, you can approximate distributed rhythmic variety by:

- using **N** mode for ordered stepping
- rotating with **Shift**
- resetting at odd lengths
- combining one side in **/2N+1**

The result is often similar to Euclidean percussion: evenly distributed but shifting accents.

2. Cross-accenting

Use one side to trigger hats regularly, and the other side to trigger accent envelopes that modulate: - hat decay - hat VCA level - hat filter cutoff

This creates a second rhythm over the first, giving a very “programmed” complex drum-machine feel.

3. Fill generator

Leave one side in stable mode, and switch the other into a Wack mode during fills.

For example: - normal groove: Side B in /2N+1 - fill: Side B in Wack /2N+1

This creates dramatic bursts of activity without losing the foundational pulse.

4. Long-form cycle design

Try this: - Side A reset every 16 clocks - Side B reset every 12 clocks - Side A in N - Side B in /2N+1

You’ll get recurring but slowly recombining patterns. This is great for: - IDM - broken beat - industrial - complex techno - generative percussion

5. Voice stealing by trigger overlap

Patch multiple outputs into a logic OR/mixer or trigger combiner before hitting a single voice. This can generate: - irregular retriggers - burst-like rolls - clustered impacts

Especially effective with: - noise snares - resonant pings - LPG bongos - FM blips

---

Performance techniques

Use Shift as a live composition control

Shift is one of the most musical controls on the module. During performance: - move Shift slowly for phrase evolution - snap it abruptly for fills and drops - rotate only one side while the other remains fixed

This creates the sensation of changing the beat “from the inside.”

Toggle Wack mode for variation

The Wack modes are powerful live tools. Use them like: - enter Wack for a breakdown - return to normal mode to re-lock the groove - place only one side in Wack for asymmetrical disruption

Separate the clocks live

If both sides normally share one clock, temporarily clock one side differently for: - burst sections - halftime/doubletime overlays - triplet percussion episodes

---

Concrete genre-oriented uses

For IDM / glitch

- Side A: N mode for phrase structure
- Side B: Wack /2N+1 for fragmented hats and clicks
- Route some outputs to sample-select, decay modulation, and pitch envelopes

For polyrhythmic techno

- Side A: /2N for kick/snare architecture
- Side B: /2N+1 for toms, rides, metallics
- Reset Side B on a different phrase boundary than Side A

For industrial / broken rhythm

- Side A: stable pulse
- Side B: Wack N or Wack /2N+1 into noisy percussive voices
- Use Shift aggressively

For “complex but danceable”

- Keep kick and main snare on the more predictable side
- Put all hats, claps, ghosts, percussion, and parameter accents on the experimental side

---

Good companion modules

Integra Solum becomes especially powerful with:

- **drum voices** or percussive synth voices
- **logic modules** for combining/diverting triggers
- **switches** for re-routing outputs
- **burst generators**
- **sample and hold**
- **VCAs**
- **envelope generators**
- **clock multipliers/dividers**
- **sequential switches**
- **accent VCAs**
- **mute/performance mixers**

If you have a logic module, combining Integra Solum outputs with AND/OR/XOR can push it into very advanced rhythmic territory.

Technical notes from the manual that matter

- Clock responds to a **rising edge around 3.3 V**
- Outputs update with about **70 µs latency**
- Trigger outputs go **0 V to 5 V**
- Trigger threshold is around **3.4 V**
- One clock input can **normal to both sides**
- Reset can be used independently per side or commonly

In practical terms, that means it should play nicely with most Eurorack trigger/gate destinations.

Best starting patch for your goal

If your goal is **densely rhythmic, hyper-complex percussion**, start here:

Recommended starter setup

Side A

- Master clock in
- Mode: **/2N**
- Outputs to:
- kick
- snare
- clap
- accent envelope

Side B

- Same clock, normalled
- Mode: **/2N+1**
- Outputs to:
- closed hat
- open hat
- shaker
- rim
- metallic percussion
- noise burst
- decay modulation trigger
- pitch envelope trigger

Then

- Rotate **Shift** on Side B
- Reset Side B at an odd phrase length
- Occasionally enter **Wack mode** on Side B

This gives you: - a dependable body - unstable upper percussion - phrase movement - odd-cycle behavior - enough complexity to stay interesting for a long time

---

Bottom line

Integra Solum excels at **architecting percussion ecosystems**, not just simple clock division. For hyper-complex rhythmic music, its strengths are:

- **two parallel rhythmic layers**
- **odd divisions**
- **rotating output order**
- **independent resets**
- **probabilistic Wack behaviors**
- **easy creation of long, non-repeating composite cycles**

If you want dense percussion, use it less like a “divider” and more like a **dual phrase engine**: - one side defines pulse, - the other side destabilizes it, - resets define meter, - Shift defines motion, - Wack defines chaos.

Generated With Eurorack Processor