Fancyyyyy — Rung Divisions V2 Clock Divider + Shift Register

- Manual PDF

Manual PDF

Fancyyyyy Rung Divisions for dense, hyper-complex percussion

Rung Divisions is not a drum voice or effect by itself. It is a **rhythmic brain**: a clock divider, dual gate bus combiner, universal shift register, noise/data mangler, and stepped CV source. That makes it extremely good for building **dense percussion ecosystems** with:

- layered polyrhythms
- shifting accents
- non-repeating trigger grids
- pseudo-random fills
- direction-reversing sequences
- variable loop lengths
- clocked CV for pitch, decay, timbre, and sample selection

If your goal is **complicated percussion**, this module is excellent because it can generate both:

1. **interrelated gate patterns** via the divider and bus system
2. **correlated modulation** via the 1-bit, 3-bit, and 8-bit outputs

So instead of just making random triggers, it makes **families of rhythms** that feel related.

What parts matter most for percussion

From the manual, these are the key functions:

- **Clock divisions /2 through /8**
- **Two OR buses: Bus1 and Bus2**
- **Bus1 clocks the shift register**
- **Direction input/button** reverses the pattern
- **Length control/CV** changes loop point
- **Chance control/CV** determines how much the pattern loops vs accepts new data
- **Data input** can be external, noise, or other rhythmic sources
- **1-bit output** = rhythmic gate stream related to the register
- **3-bit and 8-bit outputs** = stepped CV outputs for modulation

This means you can separate patching into two domains:

- **Trigger architecture**: `/2–/8`, Bus1, Bus2, Reset
- **Percussion animation**: 1-bit, 3-bit, 8-bit, Direction CV, Length CV, Chance CV, Data

---

Core idea: use it as a master percussion logic engine

A strong way to think about Rung Divisions:

- **Bus outputs** create your main trigger layers
- **Shift register outputs** animate your drum parameters
- **Length, Direction, and Chance** create phrase evolution
- **Reset** imposes larger-form structure

This is where the “dense but musical” feel comes from: chaos under a controlled pulse framework.

Best uses for hyper-complex percussion

1. Build two interacting rhythm buses

The two bus outputs are OR-combined mixtures of selected clock divisions.

Each division switch can send that division to:

- **Bus1**
- **center/off**
- **Bus2**

Because Bus1 and Bus2 are independent mixes, you can make two simultaneous rhythmic streams.

Good strategy

Use:

- **Bus1** for your “main grid”
- **Bus2** for “accents, ghost hits, fills, ratchets, or resets”

Example bus assignment

Try:

- **Bus1**: `/2`, `/5`, `/7`
- **Bus2**: `/3`, `/4`, `/8`

This immediately creates conflicting periodicities. Since prime divisions drift against even divisions, the result feels alive and asymmetrical.

The manual specifically notes that **prime divisions behave like interference patterns** against non-prime pulses. That is gold for percussion.

Patch idea

- Bus1 → kick trigger logic / main sequencer clock / trigger sequencer advance
- Bus2 → snare, hats, accents, burst generator, or envelope retriggers

This yields a groove where all events are derived from one source but do not line up in a simple bar.

---

2. Use Bus1 as your “timing skeleton” and Bus2 as your ornament layer

Because **Bus1 clocks the universal shift register**, anything on Bus1 influences the timing resolution of the evolving CV patterns.

So if Bus1 is sparse and Bus2 is dense: - your CV pattern changes slowly - accents/fills happen quickly on top

If Bus1 is dense and Bus2 is sparse: - your CV changes constantly - major accents happen less often

Practical examples

- **Bus1 sparse**: `/4 + /7`
- **Bus2 dense**: `/2 + /3 + /5`

Result: - the shift register advances on an unusual but slower pulse structure - Bus2 can trigger fast hats, metallic percussion, and extra envelope strikes

This is a good recipe for **complex but readable percussion**.

---

3. Exploit the reset input for phrase structure and odd-meter feel

The manual notes: - Reset sets all counts to 0 - external reset can create off-kilter patterns - `/8` can sync all outputs to a `/7` count making things syncopated

This means reset is not just a utility; it is a **composition tool**.

Use reset to imply complex time signatures

Feed reset from a slower periodic source that doesn't align with the main clock: - every 5 beats - every 7 beats - every 9 or 11 pulses - every phrase from another sequencer

This creates perceived meters like: - 5 over 4 - 7 over 8 - 11-pulse phrases - shifting downbeats

Patch idea

- steady master clock into Clock
- another trigger sequencer or divider into Reset every 13 pulses
- Bus1 drives kick/clap logic
- Bus2 drives hats and percussion voices

Now the pattern continuously re-anchors in a different place, which feels like **advanced meter modulation**.

Using the shift register for percussion control

The real magic is not just the divider. It's the **shift register being clocked by Bus1**.

That gives you stepped CV and gate patterns that evolve according to your rhythm bus.

4. Use the 1-bit output as a derived percussion trigger

The **1-bit output** is a gate output derived from the first bit of the register and keeps clock pulse width.

This is perfect for: - rimshots - ghost snares - FM percussion strikes - LPG plucks - sample playback triggers

Because it's tied to the shift register rather than directly to the clock divider, it feels more patterned and less mechanical.

Patch

- 1-bit → closed hat
- Bus2 → open hat
- /4 direct out → kick
- /7 direct out → snare accent

This already gives four related but non-identical trigger streams.

---

5. Use 3-bit and 8-bit outputs to modulate drum parameters

The **3-bit** and **8-bit** outputs are stepped CVs. The manual describes them as reverse encoded and useful for contrapuntal movement.

That means they are ideal for percussion modulation:

Modulate:

- drum pitch
- decay time
- LPG response
- wavefolder amount
- FM amount
- filter cutoff

- sample select
- bit depth / crush amount
- distortion drive
- VCA accent amount
- noise color or bandpass frequency

Practical assignments

- **3-bit output** → kick pitch or tom tuning
  Gives a smaller set of musically recurring values
- **8-bit output** → sample select or hi-hat decay
  Gives more variation and more “sequence-like” motion

The 3-bit output tends to feel more constrained and motif-like.
The 8-bit output tends to feel more detailed and erratic.

For percussion, that usually means: - **3-bit for body** - **8-bit for detail**

Three high-value patch strategies

Strategy A: Polyrhythmic drum matrix

Goal

Multiple percussion voices with related but offset rhythmic roles.

Patch

- Master clock → Clock input
- Set switches:
- Bus1: `/2`, `/5`, `/7`
- Bus2: `/3`, `/4`, `/8`
- Bus1 → kick sequencer clock / LPG pluck voice
- Bus2 → hi-hat or metallic voice

- 1-bit → clap trigger
- 3-bit → kick pitch CV
- 8-bit → hat decay or sample select
- Noise out or external gate pattern → Data input
- Chance around 10–11 o'clock
- Length at 5 or 6
- Occasionally trigger Direction manually or from another slow source

Result

A highly interdependent rhythm system with: - stable low-end pulse - shifting upper percussion - evolving accents - pseudo-looping phrase logic

---

Strategy B: Broken-meter percussion engine

Goal

Grooves that imply changing time signatures and asymmetrical bars.

Patch

- Fast clock → Clock
- Slow odd divider or trigger sequencer → Reset
- Bus1: `/3`, `/5`
- Bus2: `/2`, `/7`
- Bus1 clocks the shift register
- Data input from `/6` or Noise
- Chance near fully clockwise for looping, then back off slightly
- Length CV modulated slowly by a triangle LFO or stepped random
- Direction input triggered every 8, 13, or 16 pulses

Result

You get: - repeating but unstable loop lengths - reversals that make phrases “fold back” - recurring rhythmic cells that don’t land on normal 4/4 boundaries

This is excellent for: - IDM - broken techno - electro-acoustic percussion - advanced polyrhythmic live sets

---

Strategy C: Intelligent fills and accents

Goal

Keep a steady beat, but let Rung Divisions generate intricate fill behavior.

Patch

- Main sequencer handles kick/snare basics
- Feed the same clock into Rung Divisions
- Bus2 → accent or extra percussion triggers
- 1-bit → flam/ghost hit voice
- 3-bit → accent VCA CV
- 8-bit → decay time on a percussion voice
- Reset from bar start
- Direction triggered only at end-of-bar or end-of-phrase
- Chance under CV from envelope follower or slow random source

Result

Instead of replacing your drum pattern, Rung Divisions acts like a **hyperactive percussion assistant**, injecting: - ghost notes - fills - unstable accents - phrase-end mutations

Very musical if you want complexity without losing groove.

How to create denser rhythms without losing punch

Hyper-complex percussion often turns to mush if everything triggers everything. To keep it punchy:

1. Reserve one layer for stability

Keep one output dedicated to a predictable role: - `/4` for kick - Bus1 for master pulse - reset every phrase

This gives the ear an anchor.

2. Use complexity in upper layers

Use Bus2, 1-bit, and CV outputs for: - hats - clicks - metallics - modular noise percussion - accents

Dense detail belongs in the top and midrange more than in the sub layer.

3. Modulate decay instead of only adding triggers

A great trick: - use a stable trigger pattern - use 3-bit or 8-bit to vary decay, timbre, or pitch

This sounds intricate but remains groove-coherent.

4. Use chance near the edge of looping

The manual notes: - fully clockwise = pattern loops - fully counterclockwise = data comes from XOR of data input and loop point - middle = noisy interference

For percussion, the sweet spots are often: - **fully clockwise** for locked polymeter - **just below fully clockwise** for slowly mutating loop - **midway** for unstable fill behavior

That “almost-looping” zone is usually where the best advanced rhythms happen.

Best modulation techniques for percussion

Length CV = evolving phrase length

The **length** parameter changes the loop point of the shift register.

This is one of the strongest controls for nonstandard meter.

Good uses

- slow LFO into Length CV for changing bar length
- sample-and-hold into Length CV for different phrase lengths every cycle
- manual performance control for sudden contractions/expansions

Percussion result

- 5-step hat pattern over 4-beat kick
- 7-step accent cycle
- abrupt jumps from short motifs to long ones

This is a direct route to **complex time signature feel**.

Direction CV = rhythmic reversal

A gate into **Direction** reverses pattern read direction.

This is especially powerful for percussion because the same bit pattern suddenly gets read in reverse.

Use it for

- phrase-end turnaround
- mirrored fills
- reverse-feel accents
- “drummer switched hands” effect

Patch ideas

- trigger Direction every 8 bars
- trigger Direction from a rare prime division
- send Bus2-derived pulses through logic before Direction for semi-predictable flips

This creates a lot of perceived intelligence in the rhythm.

Chance CV = controlled mutation

Chance is the control between: - looped repetition - new external data - noisy interference

For percussion, this is your **fill density** and **mutation amount** control.

Good modulation sources

- slow random voltage
- envelope from kick or master accent
- bar-end trigger converted to CV
- pressure/touch controller
- another sequencer lane

Results

- stable groove with occasional mutations

- more fills during louder sections
- less repetition in breakdowns or transitions

Use external data creatively

The **Data input** is crucial. According to the manual, it can be any signal crossing 1V, and it is XOR'd, making the register inherently unstable when data is present.

That's excellent for percussion.

Best data sources for drum complexity

1. Noise output

Patch the module's **Noise output** into **Data**.

This gives: - unstable, granular, pseudo-random trigger/cv generation - good for hats, glitches, granular accents

2. One of the divider outputs

Patch `/6`, `/7`, or `/8` into Data.

This creates cyclical but offset pattern injection.

3. Another drum trigger stream

Use your snare or hat trigger as Data.

Then the shift register evolves in response to what your beat is already doing.

4. Comparator/audio signal

Use a square wave, pulse train, or comparator-derived rhythm from another oscillator.

Now the percussion engine becomes audio-coupled and can produce very lively burst structures.

Advanced percussion patch recipes

1. Prime interference hats

Patch

- Clock → Clock input
- Bus2 = `/2 + /5 + /7`
- Bus2 → hi-hat trigger
- 8-bit → hi-hat decay or filter cutoff
- Data = Noise
- Chance = around noon
- Length = 5

Sound

Shifting metallic hats with non-Euclidean-feeling clusters.

---

2. Looping tom canon

Patch

- Bus1 = `/3 + /4`
- Bus1 clocks shift register

- 1-bit → tom trigger
- 3-bit → tom pitch
- Direction toggled every 12 or 16 pulses
- Length around 6–8
- Chance near clockwise

Sound

Repeating but reversing tom melodies/rhythms, like an evolving tribal canon.

---

3. Kick with unstable accent architecture

Patch

- `/4` direct out → kick trigger
- 3-bit → kick accent VCA
- 8-bit → kick pitch envelope amount
- Bus2 → extra click layer trigger
- Reset every 15 or 17 pulses

Sound

Stable body with constantly shifting attack/weight, good for broken techno.

---

4. Snare fill generator

Patch

- Main snare on your usual sequencer
- 1-bit from Rung Divisions → secondary snare/clap
- Bus2 → short envelope into noise burst voice
- Chance CV modulated by phrase envelope
- Direction trigger at phrase end

Sound

Human-like snare doubles, drags, and fills.

---

5. Percussive FM cluster machine

Patch

Use external percussive voice: - sine or triangle oscillator - exponential/ linear FM - VCA/LPG - short envelope

Then: - Bus1 or 1-bit → trigger envelope - 3-bit → oscillator pitch - 8-bit → FM depth - Bus2 → second envelope for click/noise layer - Noise → Data

Sound

A complex family of tuned blips, zaps, claves, toms, and metallic hits.

If you want unique, punchy, and percussive results

Since Rung Divisions is not the sound source, “punch” comes from what it controls.

Pair it with:

- LPGs for woody percussion
- snappy envelopes and VCAs
- analog drum modules with CV over pitch/decay
- FM voices
- noise + filter + VCA patches
- sample players with CV-selectable slices

Great targets for its CV outputs

Kick

- 3-bit to pitch amount
- 8-bit to decay or drive

Snare

- 3-bit to noise/filter balance
- 8-bit to snap amount or bandpass center

Hats

- 8-bit to decay/open-closed continuum
- 3-bit to metallic oscillator pitch or filter cutoff

Percussive synth voice

- 3-bit to oscillator pitch
- 8-bit to wavefolder or FM amount
- 1-bit to strike/trigger

Performance tips

1. Manually flip Direction

This is one of the best live moves on the module. Reversing the pattern mid-performance can sound like: - rhythmic inversion - fill reversal - phrase foldback - stroboscopic groove shift

2. Sweep Chance near full clockwise

That area transitions from: - locked loop to - slowly mutating loop

This is a high-value performance gesture.

3. Move Length while monitoring one voice

Length changes are very audible in percussion. They create: - phrase shortening - displaced accents - odd-meter feel - abrupt reorganization

4. Reassign bus switches live

Sending divisions between Bus1 and Bus2 is like re-orchestrating your drummer in real time.

For instance: - move `/5` from Bus2 to Bus1 - suddenly the whole CV sequence changes because Bus1 clocks the register

That's a big structural move.

A practical full-system patch

"Hyper-complex modular drummer"

Voices

- Kick module
- Snare/noise patch
- Hat/metallic FM patch
- Perc synth voice or sampler

Rung Divisions patch

- Master clock → Clock
- Reset from slower sequencer lane every 11 or 13 pulses
- Noise → Data
- Bus1: `/2`, `/5`, `/7`

- Bus2: `/3`, `/4`, `/8`

Routing

- `/4` direct → Kick
- Bus2 → Hat trigger
- 1-bit → Snare accent or clap
- 3-bit → Kick pitch / perc voice pitch
- 8-bit → Hat decay / sample select / snare filter
- Direction triggered by a slow rare pulse
- Length CV from sample-and-hold
- Chance CV from slow triangle or random

What happens

- kick remains somewhat grounded
- hats and accents form dense moving layers
- CV outputs create evolving timbre and pitch correlations
- phrase length shifts prevent obvious repetition
- reset keeps the entire thing musically recoverable

This is exactly the kind of architecture that produces **high-density percussion with long-form coherence**.

Best mindset for this module

Don’t treat Rung Divisions as “random percussion.”

Treat it as:

- a **polyrhythm generator**
- a **phrase mutator**
- a **trigger-to-modulation translator**
- a **controlled instability engine**

The secret is to patch it so one part defines form and another part defines detail.

A good split is:

- **Form**: direct clock divisions + reset
- **Detail**: Bus outputs + 1-bit + 3-bit + 8-bit + chance/direction/ length modulation

That way your percussion is: - dense - weird - asymmetrical - lively - but still danceable or intelligible

---

Quick starting settings for percussion

Locked polymetric groove

- Data: `/7`
- Chance: fully clockwise or just under
- Length: 5 or 7
- Bus1: `/2 + /5`
- Bus2: `/3 + /4 + /8`

Mutating broken beat

- Data: Noise
- Chance: noon to 2 o'clock
- Length: modulated slowly
- Direction: triggered occasionally
- Reset: every odd phrase

Controlled fill generator

- Bus1 sparse
- Bus2 dense
- Chance high
- 1-bit to ghost hits

- 8-bit to decay/sample select

Final recommendation

For your goal—**densely rhythmic, hyper-complex percussion with polyrhythms and complicated patterns**—Rung Divisions is best used as the **central rhythmic logic module** that distributes:

- clocks
- derived triggers
- evolving stepped modulation
- reversible phrase structure

If you pair it with a few percussion voices that accept CV over **pitch, decay, timbre, accent, or sample selection**, it can generate extraordinarily rich and sophisticated drum music.

The strongest features for your use case are:

1. **prime clock divisions mixed on buses**
2. **Bus1 clocking the shift register**
3. **Length modulation for odd phrase lengths**
4. **Direction reversals for mirrored patterns**
5. **Chance control for mutation vs loop**
6. **3-bit and 8-bit outputs modulating percussion character**

If you want, I can also give you:

- a **10-patch cookbook** specifically for techno/IDM/drum-and-bass
- a **small-system patch plan**
- or a **pairing guide** for which kinds of drum modules work best with Rung Divisions.

Generated With Eurorack Processor