

Acid Rain – Constellation

- [Manual PDF](#)
-

[Constellation Firmware V1.1 Manual \(PDF\)](#)

Using Acid Rain Constellation to Create Full-Length Eurorack Songs

Context

One common challenge for eurorack modular musicians is turning a cool groove, beat, or riff into a full-length performance or song with structure, movement, and "story." The Acid Rain Constellation, covered in the attached manual, is a powerful rhythm and pattern generator that excels at this task.

Below is an analysis focused on songwriting and arrangement with the Constellation, including patching suggestions and integration tips for working with other modules to take your modular jams into structured compositions.

Key Features that Enable Song Structure

1. Pattern Morphing and Complexity

Constellation gives you 8 channels, each with 8 independent patterns, and the ability to combine these patterns with logical operations (AND/OR/

XOR), ratchets, bursts, chance/probability, Euclidean distribution, rotation, per-pattern and per-channel clocks, and more.

Songwriting Use: - **Variety Over Time:** Program several variations of rhythms, fills, and sparse/dense sections per channel. - **Dynamic Shifts:** Use bursts/ratchets/probability to fade between busy and minimal sections.

2. Bank and Slot Management (Scenes/Song Sections)

You can save up to 20 slots per bank, and 999 banks. Each slot stores the entire state of patterns, including mute states, clock divisions, etc.

Songwriting Use: - **Sections as Presets:** Treat each save slot as a verse, chorus, drop, breakdown, fill, or transition, and sequence through them either live or via CV control. - **Manual or Automatable Changes:** Use the “Live Mode” to jump between these “sections” on the fly or use CV to automate progression (see #4).

3. Mute and Live Mode for Variation

Both channels and individual patterns can be muted or unmuted instantly, momentarily, or permanently via the mute and live modes.

Songwriting Use: - **Adding/Dropping Layers:** Mute/unmute drums, fills, or other elements to create interest, introduce breakdowns, or build-ups. - **Performance Gestures:** Use momentary muting for fills, or toggle full mutes for dramatic transitions.

4. CV Modulation and Slot Switching

Assign CV inputs to control pattern parameters, mutes, or even to load entire save slots. CV can come from slow LFOs, sequencers, random sources, or manual controllers.

Songwriting Use: - **Automated Scene Progression:** Use a sequencer or stepped voltage source (e.g., Malekko Voltage Block, Varigate) to switch Constellation patterns/slots in time with your arrangement. - **Expressive**

Variation: Use LFOs or envelopes to modulate chance, burst, ratchet, or mute for evolving rhythms.

5. Sync and Reset Control

Constellation can be tightly clocked and reset to the rest of your system, ensuring all rhythmical changes and new scenes start in time.

Songwriting Use: - **Perfectly-Timed Transitions:** Ensure that resets and pattern/slot changes always land on the downbeat. - **Tight Integration:** Sync to or drive external sequencers, drum modules, or MIDI interfaces.

Patching and Arrangement Tips

1. Song Sections:

- Design multiple rhythm/scene variations (slots) for different parts (intro, verse, chorus, etc.).
- Map a controller (e.g., Fader bank, Planar, Pressure Points) or sequencer to select slots—build your song's journey.

2. Evolving Patterns:

- Assign LFOs or stepped random generators (e.g., Wobblebug) to modulate probability, rotation, burst, or ratchet for slow evolution in place.
- Fade between busier and sparser versions of a groove using pattern mutes, probability, or logic changes.

3. Accent and Dynamics:

- Use different logic types (AND/OR/XOR) to generate fills and accentuations by combining patterns, especially for key moments like drop-ins or turnarounds.
- Trigger “fill” patterns via manual mute, CV, or probability for occasional variation.

4. Automated Mutes and Fills:

- Use gates or triggers from clocks, envelopes, or sequencers to mute/unmute (via CV, assigned to mute parameter) for transitions, breakdowns, and “one-off” fills.

5. Structural Automation:

- Use a basic step-sequencer (e.g., Doepfer A-155, Varigate, Marbles) to

send CV/gate signals to Constellation's "Load" input, controlling when new sections or variations load.

6. Clock Manipulation:

- Change divisions/multiplications per channel to shift time signatures or create polyrhythms for breakdowns or bridges.
- Modulate clock swing for "groove" transitions (usable for breakdowns or more "live" feel moments).

7. Collaboration with Other Modules:

- Use Constellation's outputs to trigger drum modules, envelopes, sample players (e.g., Erica Sample Drum, Tiptop One), or even melodic sequencers that respond to triggers.
 - Use its main clock/reset outputs to keep all sequencers/drum machines in sync—making it the rhythmic "brain."
 - Process outputs through logic, switches, or gate mixers for further complexity.
-

Example Song Structure Workflow

1. Preparation:

2. Program Slot 1 with a basic groove (intro).
3. Program Slot 2 for a full groove (verse).
4. Slot 3 adds fills or denser patterns (chorus).
5. Slot 4 is a breakdown (minimal patterns, higher probability/mutation).
6. Slot 5 is a build-up (slowly increasing density or ratchets).
7. Slot 6 is the drop (everything in, max density).

8. Arrangement:

9. Sequence slot changes using MIDI-to-CV interface, manual slot loads in Live Mode, or a step sequencer.
 10. Use CV/LFOs for parameter morphing within sections.
 11. Use mutes for subtle performance tweaks or live improvisation.
-

In Summary

The Acid Rain Constellation is ideal for modular musicians seeking to create not just loops, but entire structured performances with dynamic

changes, automation, recallable scenes, and deep integration with other eurorack modules. By leveraging its pattern storage, slot-based workflow, powerful CV assignability, and scene/song section concepts, it becomes the heart of a modular “performance brain” capable of complex, evolving full-length songs.

[Generated With Eurorack Processor](#)