

# Noise Engineering – Integra Funkitus

---

- [Manual PDF](#)
- 

[Manual PDF / Documentation](#)

## Noise Engineering Integra Funkitus: using it for melodic components

---

Integra Funkitus is **not a pitch/CV sequencer**. It is a **4-channel rhythm modifier** that takes up to four gate streams and transforms them using probability or logic. So its role in a melodic patch is to create the **timing structure** that drives melody: when notes happen, which voice fires, when accents occur, and how several melodic lanes interact.

### What the module does

---

- **4 gate inputs**, triggered by signals over **2 V**
- **4 gate outputs** at **6 V**
- **3 modes**
- **Trigger**
- **Gate**
- **Logic**
- **4 CV inputs** for the modification knobs
- **Burn** control to combine all inputs into all outputs for instant fills/dense activity

# Why it matters melodically

---

Melody in modular is usually built from: 1. **Pitch CV source** – sequencer, quantizer, sample-and-hold, keyboard, random CV 2. **Gate/trigger structure** – determines when notes are played 3. **Envelope/VCA or LPG** – shapes the note 4. Optional **switching / logic / accents / transposition**

Integra Funkitus handles step 2 extremely well, and indirectly influences phrasing so strongly that it can make a simple pitch source feel musical and evolving.

---

## How the three modes support melody

---

### 1. Trigger mode: probabilistic note generation

---

In **Trigger mode**, each knob sets the probability that an incoming **rising edge** passes to the output. The **falling edge always happens**.

#### Musical use

This is ideal when your melodic voice expects **clean triggers** to fire: - envelopes - function generators - trigger inputs on sequential switches - sample-and-hold clocks - quantizer sample/advance inputs

#### Melodic applications

- Create **probabilistic note omissions** from a steady clock or sequence
- Turn a rigid sequence into a more human or generative melody
- Derive several related melodic rhythms from one master pattern
- Clock a sample-and-hold irregularly so new pitches occur only on selected beats

## Example

- Input 1: steady 16th-note trigger pattern
- Output 1 in Trigger mode: send to envelope for a pluck voice
- Pitch source: 8-step sequencer or quantized random CV

Result: same pitch sequence, but notes only occur when IF allows triggers through, creating melodic syncopation.

---

## 2. Gate mode: changing note lengths and phrasing

---

In **Gate mode**, probability applies to **both rising and falling edges**. This means gates can stay high longer when a falling edge is skipped.

### Musical use

This is especially good for melodic lines where **duration matters**, not just note onset.

### Melodic applications

- Produce **variable note lengths**
- Create **legato-like phrases**
- Hold VCAs, LPGs, or envelopes open longer for sustained notes
- Feed outputs to sequencers that advance on gate transitions for unusual phrase timing
- Drive slew/portamento or envelope retrigger behavior differently than Trigger mode

## Example

- Input 2: regular 8th-note gate pattern
- Output 2 in Gate mode: to envelope gate input of a bass voice
- Same pitch CV on every step, or a quantized sequence

Result: some notes become tied together, producing longer tones and more melodic contour without touching pitch.

---

### 3. Logic mode: combining rhythmic sources into melodic structures

---

In **Logic mode**, each knob determines which input channels are combined into the corresponding output. Fully counterclockwise, the output matches its own input; as you turn the knob, other parts are added/combined; fully clockwise acts as a **mute**.

The manual describes this as generalized logic combinations for four inputs.

#### Musical use

This is the most compositionally powerful mode for melody because it lets you create **derived rhythmic lanes** from multiple sources.

#### Melodic applications

- Make one pitch source play several different **interlocking note rhythms**
- Use different outputs to trigger:
  - separate melodic voices
  - octave layers
  - harmonized voices
  - transposition events
  - ratchets or ornamentation
- Build call-and-response phrasing by combining or isolating rhythmic streams
- Generate accent, sustain, and melody clocks from the same source material

## Example

Inputs: - In 1: sparse kick-like rhythm - In 2: syncopated snare-like rhythm - In 3: dense hi-hat pattern - In 4: occasional fill trigger

Outputs: - Out 1: pluck melody trigger - Out 2: bass trigger - Out 3: sample-and-hold clock for pitch changes - Out 4: trigger for transposition, sequential switch advance, or accent envelope

Result: percussive logic becomes **melodic counterpoint**.

---

# Best ways to use Integra Funkitus in melodic patches

---

## A. Generate melodies from a single pitch source

---

Use one pitch sequence and multiple rhythm outputs.

### Patch

- One CV sequencer or quantized random source -> mult to several oscillators / voices
- Four rhythmic sources -> IF inputs
- IF outputs -> separate envelopes / VCAs for different voices

### Result

Each voice shares related pitch material, but rhythm makes them feel like independent melodic parts.

This is great for: - canon-like textures - layered arps - bass + lead + high pluck arrangements

---

## B. Use IF to clock sample-and-hold for pitch creation

---

One of the strongest melodic uses is to place IF between a clock source and a **sample-and-hold**.

### Patch

- Random CV or slow modulation -> S&H input
- Rhythm source(s) -> IF input(s)
- IF output -> S&H clock
- S&H output -> quantizer -> oscillator pitch

### Result

IF decides **when a new note is chosen**.

Different modes produce different melodic feels: - **Trigger mode**: skipped notes, sparse melodies - **Gate mode**: occasional held notes / fewer pitch changes - **Logic mode**: structured, pattern-derived melodic movement

---

## C. Drive a sequential switch for melodic rearrangement

---

If you have a sequential switch, IF can create non-obvious note orderings.

### Patch

- Several fixed voltages / sequence rows / chord tones -> switch inputs
- IF output -> switch advance
- Switch output -> quantizer or oscillator pitch

### Result

Instead of sequencing pitch directly, you sequence **selection events**. IF turns rhythms into note-order changes.

This is especially effective when: - one IF output drives note trigger - another IF output drives switch advance - another IF output drives octave jump or transposition

---

## D. Create harmony by splitting rhythmic roles

---

Use IF outputs for different harmonic functions.

### Patch concept

- Output 1 -> root note voice
- Output 2 -> fifth or upper harmony voice
- Output 3 -> transposition trigger
- Output 4 -> ornament voice or echo voice

If all voices share a quantizer or tonal source, the rhythmic separation from IF creates melodic interplay with minimal patch complexity.

---

## E. Use Burn as a melodic fill generator

---

The **Burn** function combines all inputs into all outputs.

### Musical effect

For melody, this can act like: - a fill button - phrase climax - burst of ornamentation - temporary densification of an arpeggio or sequence

### Patch idea

Normally: - sparse IF outputs trigger a few selective notes

During Burn: - all outputs get much denser activity - can trigger extra melodic layers, pitch changes, or transposition events

This is useful for transitions between: - verse/chorus - pattern A/B - buildup/drop

---

# Concrete melodic patch recipes

---

## 1. Probabilistic arpeggio

---

### You need

- clock
- gate sequencer or trigger pattern source
- Integra Funkitus
- sample-and-hold or sequencer
- quantizer
- oscillator
- envelope + VCA/LPG

### Patch

- Clocked trigger pattern -> IF input 1
- IF in Trigger mode
- IF output 1 -> S&H clock and/or envelope trigger
- Random CV or stepped CV -> S&H input
- S&H -> quantizer -> oscillator pitch

### What happens

Not every clock produces a note, so the melody becomes selectively sparse and musical.

---

## 2. Legato bassline generator

---

### Patch

- Regular gate pattern -> IF input 1
- Set IF to Gate mode

- IF output 1 -> envelope gate of bass voice
- Sequencer or quantized CV -> oscillator pitch

## What happens

Some note-offs are skipped, creating tied or extended notes. The bassline feels more alive and less grid-locked.

---

## 3. Interlocking duet from drum rhythms

---

### Patch

- 4 drum trigger patterns -> IF inputs 1–4
- IF in Logic mode
- Output 1 -> envelope of low melodic voice
- Output 2 -> envelope of high melodic voice
- Shared quantized pitch source, or two related pitch sources

## What happens

Drum-derived logic creates two interwoven melodic lines. Great for techno, IDM, electro, or generative patches.

---

## 4. Melody plus transposition lane

---

### Patch

- Main rhythm -> IF input 1
- Accent rhythm -> IF input 2
- Fill rhythm -> IF input 3
- IF in Logic or Trigger mode
- Output 1 -> melodic envelope trigger
- Output 2 -> trigger precision adder/transposition switch
- Output 3 -> reset/advance sequencer or trigger auxiliary ornament voice

## What happens

The melody itself and its harmonic movement become rhythmically linked but not identical.

---

## 5. Multi-voice canon patch

---

### Patch

- Feed four related clocks or trigger streams into IF
- Send outputs 1–4 to four envelopes controlling four voices
- Pitch source options:
  - same quantized CV to all voices
  - same CV with different octave offsets
  - one sequencer row plus precision adders

### What happens

You get a tightly related but rhythmically diverging ensemble from a small amount of pitch material.

---

## CV control for evolving melodic timing

---

Each channel has a **CV input for the modification knob**, and when patched the knob becomes an attenuator.

This is extremely useful for melodic movement because you can animate: - trigger probability - gate probability - logic selection amount

### Good CV sources

---

- slow LFO
- stepped random

- envelope from another phrase
- sequencer row
- pressure/touch controller
- manual offset from a performance module

## Musical results

---

- melody density changes over time
- note duration breathes
- logic combinations morph between sections
- one static pitch sequence feels composed and arranged

---

## Strengths in a melodic system

---

Integra Funkitus is best thought of as a **melodic phrasing engine**, not a note generator.

It is especially strong when paired with: - quantizers - sample-and-hold - CV sequencers - switches - precision adders - envelopes/LPGs - multi-voice oscillator setups

It excels at: - creating note events - removing note events - extending note lengths - deriving multiple related melodic lines - turning percussion logic into melodic logic

---

## Limitations

---

Based on the manual: - It does **not generate pitch CV** - It does **not quantize** - It does **not store sequences** - It works on **gate/rhythm structure only**

So if you're asking whether it can make melody by itself: **not directly**.

If you're asking whether it can make a melodic system far more expressive: **absolutely**.

---

# Best overall melodic use case

---

A very effective setup is:

1. Several clocks or rhythm patterns into IF
2. IF outputs used to:
3. trigger envelopes
4. clock sample-and-hold
5. advance switches
6. trigger transpositions
7. Quantizer keeps all voltages musical
8. One or more oscillators turn those events into notes

That gives you melody with: - variation - phrasing - rests - sustain differences - structural fills - inter-voice interplay

In short, **Integra Funkitus is a rhythm intelligence module for melodic patching.**

[Generated With Eurorack Processor](#)