

# Noise Engineering – Integra Solum

---

- [Manual PDF](#)
- 

[Manual PDF / Documentation](#)

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

---

Integra Solum is not a pitch sequencer in the traditional sense, but it is very useful for creating **melodic structure indirectly**. It excels at generating **timed trigger patterns** that can drive sequencers, sample-and-holds, envelope generators, switches, quantizers, and logic modules. In other words: it is a **melody organizer, rhythmized note selector, and variation source**.

### What the module does musically

---

Integra Solum is a:

- **dual rotating clock divider**
- with **16 trigger outputs** total
- split into **two independent sections of 8 outputs**
- with:
  - independent or normalized **clock inputs**
  - independent or normalized **reset**
  - **Shift/Offset** rotation per side
  - three main output behaviors:
    - **/2N** = powers-of-two divisions
    - **N** = sequence of eight

- $/2N+1$  = odd divisions
- extra **Wack mode** random behaviors

This means the module gives you multiple related trigger streams from one clock, and those streams can be used to produce melodic events at different rates and in different orders.

---

## Important technical details from the manual

---

- **Clock input threshold:** responds to a rising edge around **3.3 V**
- **Output latency:** about **70  $\mu$ s**
- **Trigger output voltage:** approximately **3.4 V trigger**, outputs range **0 V to 5 V**
- **Reset behavior:** pressing Reset pauses clock processing; on release, reset occurs on the first clock after release
- **Shift/Offset:** rotates the output assignment, so output 2 can become the first step, etc.
- **Inputs are normalized:** one clock can drive both sides unless separate clocks are patched

These details matter because the module is reliable as a rhythmic brain for sequencing other pitch-producing modules.

---

## How it creates melody

---

By itself, Integra Solum does **not output pitch CV**. To make melody, pair it with modules that do one or more of the following:

- **generate pitch CV**
- sequencer
- quantizer
- random voltage source
- sample-and-hold
- **respond to triggers**
- sequential switch

- envelope
- burst generator
- trigger-to-CV modules
- **choose among notes**
- switch
- addressed switch
- quantizer scale memory
- precision adder with stepped voltages

Integra Solum supplies the **when** and **which subdivision**, and another module supplies the **what note**.

---

## Best melodic uses

---

### 1. Triggering a pitch sequencer at different divisions

---

This is the most direct use.

#### Patch idea

- Master clock -> Integra Solum Clock
- One Integra Solum output -> sequencer clock input
- Sequencer pitch CV -> oscillator 1V/oct
- Sequencer gate/trigger or another Integra Solum output -> envelope -> VCA

#### Why it works

Different outputs represent different clock relationships. If you clock the sequencer from one output and articulate notes from another, you get: - phrase offsets - syncopated note advances - repeating but asymmetrical melodic loops

## Musical result

- slower note changes over faster articulation
- polyrhythmic melody phrasing
- melodic cycles that drift against drum clocks

## Best modes

- $/2N$  for stable melodic phrases with traditional divisions
  - $/2N+1$  for longer non-even phrase structures
  - $N$  for stepped note-order motion across 8 outputs
- 

## 2. Driving a sample-and-hold into a quantizer

---

This is one of the strongest melodic applications.

### Patch idea

- Noise or slow chaotic CV -> sample-and-hold input
- Integra Solum output -> sample-and-hold trigger
- sample-and-hold output -> quantizer
- quantizer output -> oscillator pitch
- another Integra Solum output -> envelope/VCA trigger

### Why it works

Each trigger output can sample new voltages at different times. By changing mode and Shift, you reshape **when a new note is chosen**.

### Musical result

- evolving melodies from one clock source
- related note changes across two voices
- controlled randomness when quantized to a scale

## **Especially good with Wack mode**

Wack mode makes this excellent for semi-random melodic lines: - **Wack / 2N**: probabilistic note updates - **Wack N**: one random trigger each step, useful for sparse note selection - **Wack / 2N+1**: dense randomized triggering for active generative melodies

---

## **3. Using the two sides as “pitch rhythm” and “articulation rhythm”**

---

Since there are two divider sections, you can split melodic duties.

### **Patch idea**

**Left side** - clocks a sequencer or sample-and-hold for pitch changes

**Right side** - triggers envelopes, accents, or filter pings

### **Why it works**

A melody is not just notes; it is also: - when notes change - when notes sound - when accents happen

Integra Solum lets those be related but not identical.

### **Musical result**

- repeated notes with shifting articulation
- one pitch held while multiple rhythmic triggers fire
- accents landing on different note changes each cycle

This is especially powerful if: - one side uses **/2N** - the other uses **/2N+1** - both are fed from the same normalized clock

That creates long-form phrase interplay.

---

## 4. Rotating note selection with a sequential switch

---

The **Shift** control is very valuable for melody.

### Patch idea

- 4–8 fixed voltages or sequence rows -> sequential switch inputs
- Integra Solum triggers -> switch advance or select behavior
- switch output -> quantizer -> oscillator pitch

or

- Multiple Integra Solum outputs -> trigger different stages of a sequential switch
- Switch selects among several pitch sources:
  - tuned voltages
  - sequence rows
  - offsets
  - random sources

### Why it works

Shift rotates which output occurs “first” in the cycle. That is effectively **re-ordering event priority** without repatching.

### Musical result

- same note set, new melodic order
  - transposed-feeling phrase movement without changing pitches
  - evolving ostinatos
-

## 5. Building call-and-response melodies with the two sections

---

Because both sides can run independently or from one normalized clock, they work well as a two-voice melodic system.

### Patch idea

- Same master clock to both sides
- Left side -> quantized random melody voice A
- Right side -> separate sequencer or transposed voice B
- Use different modes on each side

### Example

- Side A in **N mode**
- Side B in **/ $2N+1$  mode**

### Musical result

- one voice behaves more regularly
- one voice behaves more oddly spaced
- together they create counterpoint-like phrasing

You can also use reset strategically so both voices periodically realign.

---

## Mode-by-mode melodic applications

---

### / $2N$ mode

---

This is the classic clock divider behavior: powers of two.

## Melodic use

- phrase-length control
- occasional note changes
- stable bassline stepping
- long drones with periodic pitch motion

## Good for

- bass melodies
- root note changes
- transposition clocks
- structured arpeggio resets

## Pair with

- 8-step sequencer
- quantizer
- precision adder
- clocked switch

---

## N mode

---

The manual describes this as a **sequence of eight**.

## Melodic use

This is useful when you want the eight outputs to behave more like a stepped progression through positions rather than just subdivisions.

## Good for

- scanning through note sources
- addressing a switch
- routing one step to one event
- pseudo-sequencing using trigger destinations

## Example

Patch each output to trigger a different fixed voltage source or envelope/VCA path. This can create note-by-note melodic construction even without a normal sequencer.

---

## **/2N+1 mode**

---

Odd-number divisions.

### **Melodic use**

This is where phrases become less square and more interesting.

### **Good for**

- phrase lengths that take longer to repeat
- melodic syncopation
- cross-rhythm against 4/4
- evolving generative lines

## Example

Use odd divisions to trigger pitch changes while even divisions trigger note articulation. The melody will “walk around” the barline before repeating.

---

## **Wack mode for melody**

---

Wack mode turns Integra Solum into a strong generative melody source.

## **Wack /2N**

---

probabilistic divide by two; 50% chance a trigger generates at each step

## **Musical use**

Use this to: - sometimes advance a sequencer - sometimes resample pitch  
- create holes in a melodic line

Result: - familiar timing density, but with variation

---

## **Wack N**

---

a single random trigger is generated at each step

## **Musical use**

This is excellent for: - selecting one note source among eight - triggering one of eight melodic events - animating a switch matrix

Result: - one note/event at a time, but unpredictably chosen

This can be incredibly musical if those eight destinations are: - eight fixed tuned voltages - eight transposition amounts - eight different sequencer rows - eight envelope-decay settings for note character

---

## **Wack /2N+1**

---

all 8 outputs act independently, each with 50% chance of going high per input clock

## **Musical use**

This is the most chaotic.

Use it for: - dense generative note changes - layered melodic trigger clouds  
- probabilistic harmonies if multiple voices are involved

Best when tamed by: - quantizers - slew - attenuators - logic - sample-and-hold

---

# Concrete melodic patch recipes

---

## Patch 1: Generative lead melody

---

**Goal:** a constantly changing but scale-locked lead line

- Master clock -> Integra Solum clock
- Integra Solum output -> sample-and-hold trigger
- Smooth random CV -> sample-and-hold input
- sample-and-hold output -> quantizer
- quantizer output -> oscillator pitch
- Another Integra Solum output -> envelope trigger
- Envelope -> VCA CV
- Oscillator -> VCA -> mixer

### Suggested settings

- Use **/2N+1** or **Wack /2N**
- Adjust **Shift** until the note-change timing feels musical

### Result

A melodic line with recurring internal timing but non-obvious phrasing.

---

## Patch 2: Bassline with accent structure

---

**Goal:** stable bass notes with dynamic groove

- Integra Solum left side output -> sequencer clock
- Sequencer CV -> bass oscillator
- Integra Solum right side output -> envelope trigger
- Another right-side output -> accent envelope or filter ping
- Reset both sides from bar reset

## Suggested settings

- Left side: /2N
- Right side: /2N+1

## Result

Bass notes move predictably, but accents and articulation stay alive.

---

## Patch 3: Eight-note melodic selector

---

**Goal:** create melody from eight fixed notes

- Tune 8 voltage sources or use a switchable voltage bank
- Feed them into a sequential switch or addressed switch
- Integra Solum in **N** or **Wack N**
- Outputs drive switch selection/advance or directly trigger note lanes
- Switch output -> quantizer if needed -> oscillator

## Result

A melody made from a custom note palette, with re-ordering via Shift.

---

## Patch 4: Two-voice canon / counterpoint

---

**Goal:** related melodies from one clock

- One clock patched to one Clock input so it normals to both sides
- Side A output -> sequencer A clock
- Side B output -> sequencer B clock
- Each sequencer -> separate oscillator
- Shared reset every 1 or 2 bars

## Suggested settings

- Side A: / $2N$
- Side B: / $2N+1$  or  $N$

## Result

Two voices sharing timing DNA but phrased differently.

---

## Patch 5: Arpeggio reshuffler

---

**Goal:** non-linear arpeggiation

- Arpeggiator or stepped CV source -> quantizer
- Integra Solum output -> sample-and-hold trigger or sequencer advance
- Use **Shift** to rotate trigger order
- Use another output to reset the arpeggiator/sequencer occasionally

## Result

The same note pool produces different melodic contours depending on trigger ordering.

---

## How to use reset musically

---

Reset is very important for melody because it controls phrase boundaries.

Use reset to: - realign both divider sections at the start of a bar - force a repeating melodic form - create “long cycle, short resolution” structures - periodically bring generative patches back home

Since the reset can be applied independently or normalized, you can choose whether both sides: - loop together - or drift and then reunite

That makes the module very useful for **structured generative melody**.

---

## Best companion modules for melodic work

---

Integra Solum pairs especially well with:

- **quantizers**
  - to turn random/stepped CV into scales
  - **sample-and-hold**
  - for clocked note generation
  - **sequential switches**
  - to choose among note sources
  - **CV sequencers**
  - to create note order from triggers
  - **precision adders**
  - for transposition layers
  - **logic modules**
  - to combine trigger streams into more complex melodic clocks
  - **envelopes and VCAs**
  - for shaping articulation independently from pitch changes
  - **clock multipliers/dividers**
  - to create broader rhythmic ecosystems
- 

## Performance advice

---

### Use Shift as a melodic arranger

---

Instead of thinking of Shift as only “rotation,” think of it as: - phrase re-indexing - note-priority shifting - groove displacement

A small shift can make the same patch sound like a different composition.

## Use one side for note change, one side for note emphasis

---

This is one of the easiest ways to get musical, non-flat results.

## Use Wack mode sparingly at first

---

For melody, Wack mode is best when there is also: - quantization - reset - some stable reference rhythm

Otherwise it can become too diffuse.

## Reset every few bars

---

This keeps generative patches from losing coherence.

---

# Summary

---

Integra Solum is best understood as a **melodic timing and structure generator** rather than a direct melody source. It can create melodic components by:

- clocking sequencers at different divisions
- triggering sample-and-hold for quantized random notes
- selecting among note sources
- separating pitch-change timing from articulation timing
- creating two interrelated melodic voices
- adding controlled randomness with Wack mode
- using Shift to reorder phrase behavior

If you combine it with a **quantizer, sample-and-hold, sequencer, or sequential switch**, it becomes a powerful tool for: - generative melodies - bassline variation - polymetric arpeggios - counterpoint-like dual voice motion - evolving melodic structures with strong rhythmic identity