

Behringer – 173

- [Manual PDF](#)
-

[Manual PDF](#)

Behringer System 100 173 QUAD GATE/MULTIPLES

Using it to create melodic components in a Eurorack system

The **173 Quad Gate/Multiples** is not itself a sound source or pitch sequencer. It's a **utility module** that helps **route, distribute, and condition signals** that are essential for building melodic patches.

In practice, this module helps you make melodies by:

- distributing **pitch CV**
 - distributing **gate/triggers**
 - selectively **passing or blocking signals**
 - creating **performance-routing structures**
 - splitting one melodic control source to several destinations
-

What this module does

From the manual, the 173 contains two functional sections:

1. Quad Gate section

There are **4 gate channels**, each with:

- **GATE IN**
- **GATE OUT**
- **GATE CV**
- non-inverting input (**active high**)
- inverting input (**active low**)

This means each channel can act like a **voltage-controlled signal gate**. A signal enters **GATE IN**, and whether it appears at **GATE OUT** depends on the CV applied to the gate control.

This is very useful for melodic patching because it lets you decide **when a CV or signal is allowed through**.

2. Multiples section

There are **6 rows of passive multiples**, each row with **4 parallel jacks**:

- A, B, C, D are all connected together within a row

This allows one signal to be copied to multiple destinations. Since these are **passive multiples**, they are best for many CV/audio duties, but precision pitch distribution can sometimes depend on the receiving modules.

Important technical points from the manual

Gate section

- CV control has:
 - **non-inverting threshold: > +3 V**
 - **inverting threshold: < +1.5 V**
 - Max control input: **10 V p-p**

So if you feed a standard Eurorack gate or trigger into the control input, the gate will open/close reliably.

Multiples section

- 6 sets of 4 parallel jacks
- passive
- works with CV and audio signals

Size / power

- 16 HP
 - 40 mA on +12 V
 - 40 mA on -12 V
-

How it helps create melodic music

Melodic patches usually need three things:

1. **Pitch CV**
2. **Gate/trigger timing**
3. **Modulation/control over when notes happen**

The 173 is especially useful for items 2 and 3, and secondarily for routing pitch CV.

Best melodic uses

1. Split one pitch sequence to multiple destinations

Use a row of the **MULTIPLES** to send one pitch CV to several modules at once.

Example

Send one sequencer's pitch CV to:

- oscillator 1 pitch input
- oscillator 2 pitch input
- quantizer input or transpose input
- CV recorder / precision adder / scope

Why this is melodic

This lets you build:

- layered unisons
- octaves
- interval harmonies
- doubled melodies
- parallel voices

Patch

- Sequencer pitch out → **173 Multiple Row 1, jack A**
- Row 1 jack B → VCO 1 1V/Oct
- Row 1 jack C → VCO 2 1V/Oct
- Row 1 jack D → another melodic destination

Note

Because the multiples are passive, if you are distributing very precise **1V/Oct pitch**, check tuning. In many systems it works fine, but buffered multiples are more accurate for critical pitch tracking across multiple oscillators.

2. Split one gate pattern to several envelopes or voices

Use the **MULTIPLES** to distribute a gate signal.

Example

One sequencer gate goes to:

- envelope for voice 1
- envelope for voice 2
- clock/reset input elsewhere
- gate-control CV input on one of the 173 gate channels

Why this is melodic

This lets multiple voices articulate together, which is useful for:

- bass + lead doubling
- chords from multiple oscillators
- synchronized accents

Patch

- Sequencer gate out → **173 Multiple Row 2 A**
- Row 2 B → envelope 1 gate in
- Row 2 C → envelope 2 gate in
- Row 2 D → gate logic / trigger processor

3. Use the gate channels to allow notes through only at selected times

This is where the 173 gets musically interesting.

A gate channel can be used to **pass or block a melodic CV stream**.

If you send pitch CV, stepped modulation, or another melodic control signal into **GATE IN**, and control the gate opening with another rhythm or logic signal into **GATE CV**, you can create **selective melodies**.

Example

You have: - a running pitch sequence - a second rhythmic gate pattern

The second pattern opens the gate only on some steps.

Result

Only selected notes pass through, producing: - sparse melodies - rhythmic note omission - evolving motifs

Patch

- Pitch sequencer CV → **Gate 1 IN**
- Rhythm gate pattern → **Gate 1 CV** non-inverting
- **Gate 1 OUT** → oscillator pitch input or quantizer input

Musical effect

This can create: - broken sequences - rhythmic filtering of melodies - alternating note access - pseudo-generative pitch phrases

4. Gate a modulation source before it reaches pitch

Another strong melodic use is to gate a modulation source that affects pitch.

Example

Instead of sending a continuous random CV into your oscillator pitch or quantizer, use the 173 to only let that CV pass at certain moments.

Patch

- Random stepped CV → **Gate 2 IN**
- Clock division or pattern gate → **Gate 2 CV**
- **Gate 2 OUT** → quantizer input
- Quantizer output → VCO pitch

Result

You get notes only when the gate opens, so the random source becomes a more structured melodic source.

This is especially good for: - generative melodies - probabilistic lines - rhythmic pitch changes

5. Alternate between two melodic layers

While the 173 does not mix or switch between two sources directly, you can use multiple gate channels to control separate melodic streams and combine them downstream if you have another utility module.

Concept

- Melody A goes through Gate 1
- Melody B goes through Gate 2
- Different gate patterns open each path
- Downstream module selects, sums, or voices them separately

Use

This is effective for: - call-and-response - verse/chorus CV structures - alternating arpeggio lines - counterpoint between two oscillators

6. Build phrase structure with transposition routing

Use the **MULTIPLES** to distribute a transposition CV to several places, and use **gate channels** to let that transposition or melodic modifier affect only certain sections.

Example

- Base sequence plays constantly

- A transposition CV is allowed through only every 8 bars or on selected steps

Patch idea

- Transposition CV source → **Gate 3 IN**
- Phrase trigger → **Gate 3 CV**
- **Gate 3 OUT** → precision adder transpose input
- Sequencer pitch CV also goes to precision adder
- Precision adder out → oscillator pitch

Result

The melody changes register or key only when the phrase gate opens.

7. Create melodic accents by routing envelopes selectively

A melody is not only pitch; articulation shapes melody perception. The 173 can route gates to determine when envelopes happen.

Example

- Main sequence gate goes through a multiple
- One branch triggers the normal VCA envelope
- Another branch controls the 173 gate that allows a second accent envelope or timbral modulation to pass

Result

Some notes become accented, brighter, or more dramatic.

This creates: - melodic emphasis - phrase punctuation - dynamic contour

8. Derive harmony from one melodic source

The multiple section is especially helpful here.

Patch concept

- Main pitch CV → multiple
- Copy 1 → VCO 1
- Copy 2 → precision adder with offset
- Copy 3 → second voice or filter tracking
- Copy 4 → quantizer auxiliary path

If another utility adds intervals, the 173 serves as the central distribution hub.

Result

You can derive: - octaves - fifths - drone-related harmonies - parallel intervals

9. Use gate channels as rhythmic melodic masks

Think of each gate as a **mask** over a CV source.

Sources to mask

- sequencer pitch CV
- keyboard CV
- joystick CV
- sample-and-hold CV
- slow LFO into quantizer

Controllers for the mask

- clock divisions

- trigger sequencer
- manual gate button
- comparator output
- logic module

Musical outcome

This creates: - conditional melodies - polyrhythmic note appearance - repeating but varied lines - motif fragmentation

Concrete patch ideas

Patch 1: Sparse techno lead

Goal

Turn a dense sequence into a punchier melodic line.

Patch

- 8-step sequencer pitch CV → **Gate 1 IN**
- Trigger sequencer pattern → **Gate 1 CV** non-inverting
- **Gate 1 OUT** → quantizer → VCO 1V/Oct
- Main sequencer gate → envelope → VCA
- Optional: same pitch CV also sent via multiple to a second oscillator

Result

The pitch changes only on selected rhythmic moments, making the line more syncopated and hook-like.

Patch 2: Generative melody selector

Goal

Use randomness but keep it musical.

Patch

- Stepped random CV → **Gate 2 IN**
- Euclidean trigger pattern → **Gate 2 CV**
- **Gate 2 OUT** → quantizer input
- Quantizer output → oscillator pitch
- Clock also triggers envelope/VCA

Result

Random values become a structured melody determined by the rhythm pattern.

Patch 3: One sequence, two oscillators, one harmony

Goal

Create a thicker melodic voice.

Patch

- Sequencer pitch out → **Multiple row 1**
- Row 1 B → VCO 1 1V/Oct
- Row 1 C → VCO 2 1V/Oct
- Row 1 D → precision adder or tuner reference
- Sequencer gate out → **Multiple row 2**
- Row 2 B/C → envelopes for both voices

Result

Two oscillators track the same melody. Detune one slightly or offset one by an interval elsewhere for harmony.

Patch 4: Phrase-based transposition

Goal

Make a melody evolve across longer sections.

Patch

- Main pitch sequence → VCO through usual path
- Slow CV or keyboard transpose voltage → **Gate 3 IN**
- Phrase trigger every 16 steps → **Gate 3 CV**
- **Gate 3 OUT** → transpose input of sequencer/adder/quantizer

Result

The melody shifts only at phrase boundaries.

Patch 5: Inverted gate logic for rests

The manual notes the gate CV supports **inverting (active low)** behavior.

Use

This is useful when you want the signal to pass **except when a control gate is present**, effectively carving out rests or avoiding overlaps.

Patch

- Pitch CV or modulation CV → **Gate 4 IN**
- A “mute” gate pattern → **Gate 4 CV** inverting input

- **Gate 4 OUT** → melodic destination

Result

Whenever that mute pattern goes high, the path closes, creating rests or withheld notes.

This is great for: - anti-accent patterns - syncopated silences - phrase breathing

How this works with other typical modules

The 173 becomes much more musically useful when paired with:

Sequencers

Use it to split and condition: - pitch CV - gate lanes - reset pulses

Quantizers

Send gated random/LFO/sequenced voltages into a quantizer to turn utility gating into melody generation.

Oscillators

Distribute one melodic CV to several VCOs for layered voices.

Envelopes and VCAs

Use multiples for shared timing, and gate channels for selective articulation.

Sample & hold / random modules

Gate their output before quantization to create controlled generative lines.

Clock dividers / logic

Use divided clocks and logic outputs to decide when notes are allowed through.

Precision adders / transposers

Multiples distribute the source; gates decide when transposition enters the phrase.

Limitations to keep in mind

1. Passive multiples are not always ideal for precision pitch

For exact melodic tuning over several oscillators, a **buffered multiple** may be better.

2. This module does not generate pitches by itself

It needs external: - sequencers - random CV - keyboards - LFOs - quantizers
- oscillators

3. Gate section is best thought of as a utility router

It won't replace a dedicated sequential switch, logic processor, or precision CV processor, but it is very useful for controlling **when melodic voltages are allowed through**.

Best musical roles for the 173

If your goal is melody, this module is best used as:

- a **pitch CV distributor**
 - a **gate distributor**
 - a **rhythmic mask for melodic CV**
 - a **phrase control utility**
 - a **voice-doubling hub**
 - a **generative melody conditioner**
-

Summary

The **Behringer 173 Quad Gate/Multiples** helps create melodic components not by generating melodies directly, but by making melodic systems more flexible and playable.

Its strongest melodic uses are:

- splitting one pitch line to several voices
- splitting gates to multiple envelopes or voices
- rhythmically allowing or blocking pitch/modulation signals
- structuring generative melodies
- creating phrase-level melodic variation
- building layered harmonic patches from one source

If you want, I can also turn this into: 1. a **set of example patches using specific Behringer System 100 modules**, or

2. a **“melody recipe”** guide showing exact cable connections for a full voice patch.

Generated With Eurorack Processor