RoboticBean CREATIVE

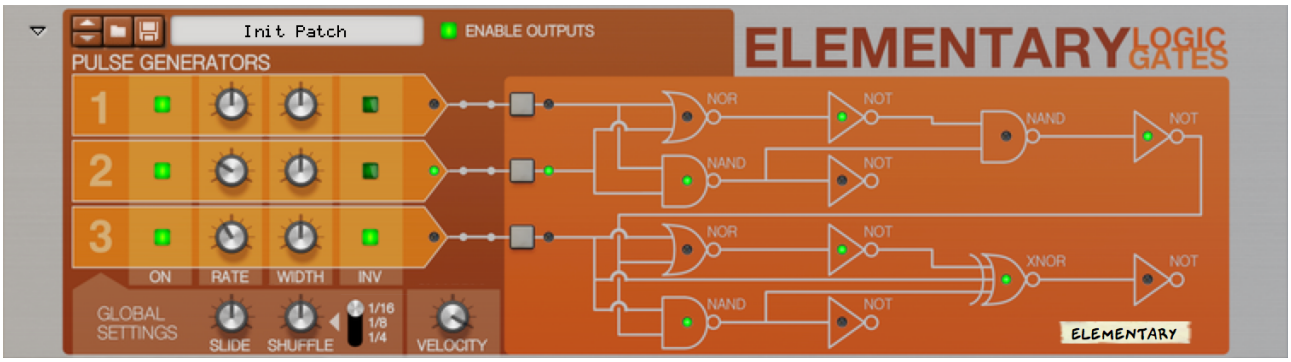
ELEMENTARY LOGIC GATES

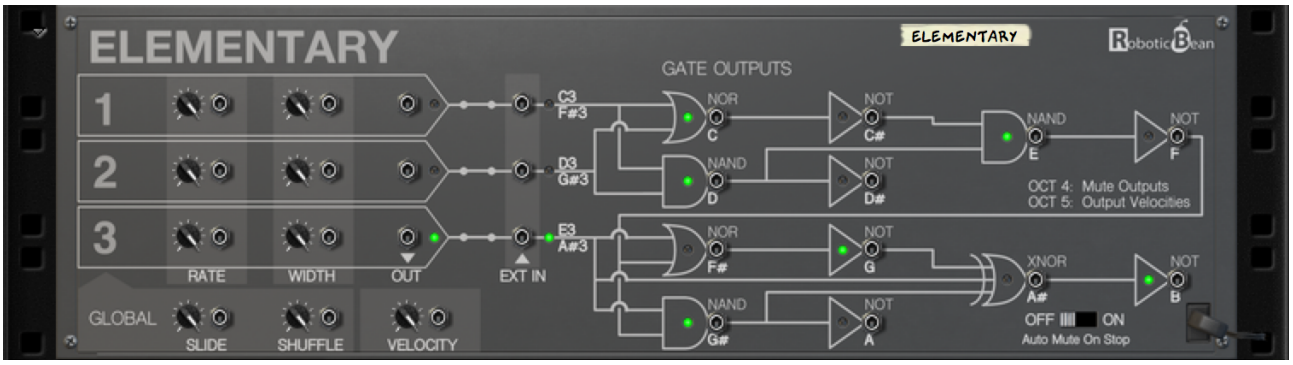
User Guide

Version 1.0.1

Contents

Introduction



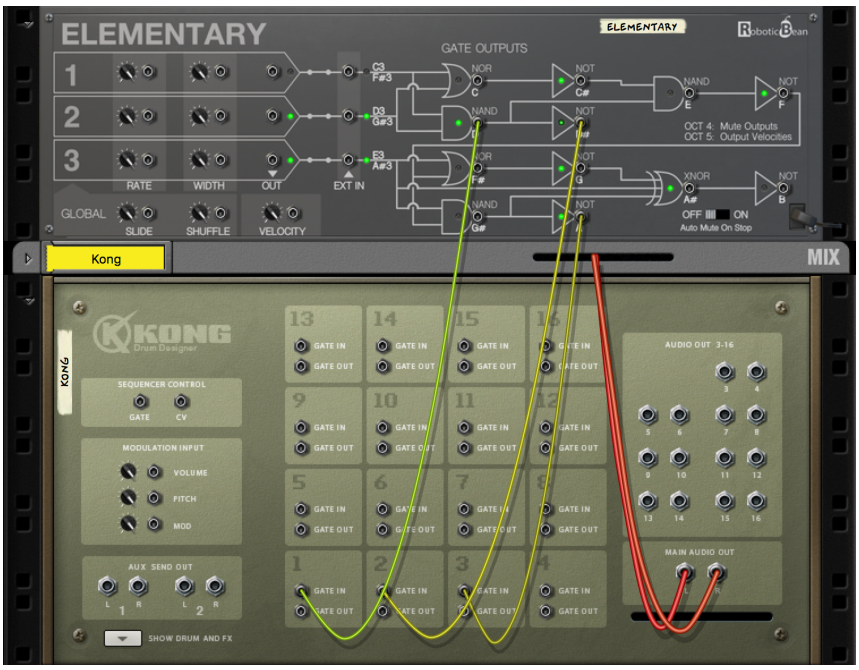


Elementary is a pattern generator based on boolean logic gates. It takes three input signals and combines them into 12 different patterns, each one with its own dedicated CV output. Elementary comes with three tempo synced pulse generators (clocks) that provide default input signals to the logic gates, so you can start generating beats out of the box.

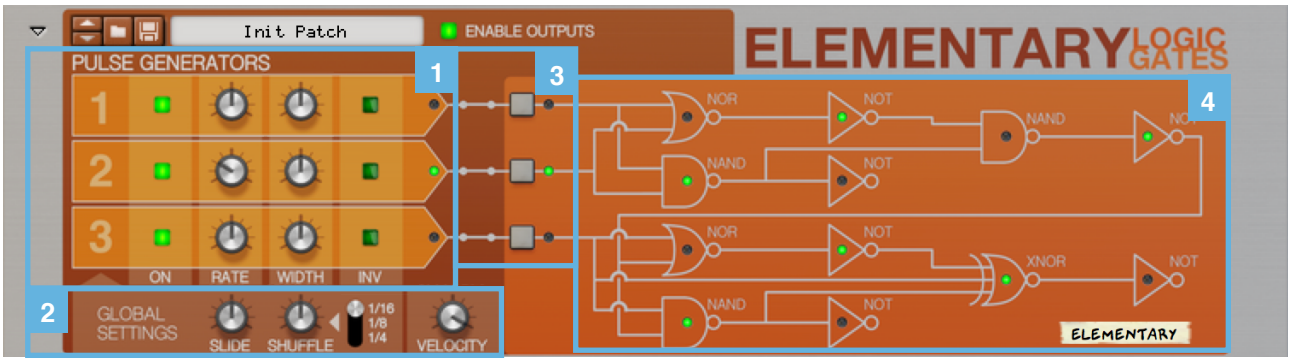
Elementary really starts to shine when you start using the external CV inputs, because you can start combining patterns from other devices such as Euclid Rhythm Generator.

Getting Started

To get started with Elementary, we recommend that you create a Kong loaded with your favorite drum sounds, and then simply experiment by connecting some of Elementary's gate outputs to different pads on the Kong. We've also included some Combinator patches you can use as a starting point.



The Front Panel



At the top of the front panel there's the patch browser, and the global Enable Outputs button that turns all outputs of Elementary on and off. The rest of the front panel can be divided into four major sections:

1. **Pulse Generators**
2. **Global Settings**
3. **Inputs**
4. **Logic Gates**

Pulse Generators

Elementary has three tempo synced rectangular pulse generators, or clocks. They will generate clock pulses as long as they are enabled and Reason's main transport is running. The output of each pulse generator is prerouted to one of Elementary's inputs. Each pulse generator has its own **On** button, **Rate** and **Width** knobs, an **Invert** button, and an **Activity LED**.

On

This button decides whether the pulse generator is active or not. This will affect both the Input section and each pulse generator's separate CV output on the back panel.

Rate

This determines the rate (speed) of the pulse generator, in terms of note divisions of Reason's main transport. The available divisions are 2/1 (two bars), 1/1 (one bar), 1/2, 1/4, 1/8, 1/8T, 1/16, 1/16T and 1/32.

Width

This knob determines the pulse width for each generator, in percent. At 0%, the output from the pulse generator will be low almost constantly, with an extremely short peak at the beginning of each division. At 100%, the output will be high all the time except for a short dip at the end of each division. At 50%, the pulse generator will produce a square wave.

Invert

The invert button swaps the high and low states of the pulse generator, so the division will begin in the low state, and end in the high state. This is useful for creating complementary clock pulses, or back beats. The output of the pulse generator will still be low when the transport is stopped.

Activity LED

The activity LED shows the current state of the pulse generator. When the LED is lit green, the pulse is high. If you have cable connected to the pulse generator's dedicated CV output, the LED will show a faint green glow when the output signal is Low.

Global Settings

The are four global controls on Elementary: **Slide**, **Shuffle Amount**, **Shuffle Resolution** and **Velocity**. They will affect all pulse generators equally.

Slide

Offsets the sync reference for all the pulse generators, in MIDI ticks (1/3840 notes). The range is +/- 120 ticks, which means that you can offset the sync reference 1/32 note, either forwards or backwards. At negative values, the groove will feel slightly rushed, and at positive values the groove will sound a little more relaxed. At 0 (knob at noon), the beat will be exactly on time.

Shuffle Amount

Shuffle causes some of the pulses to play later or earlier in relation to the main beat. The shuffle amount knob decides how much the affected pulses will be moved in time. The control ranges from 25% to 75%. At 50%, no shuffle is applied and all the pulses will be played at equal intervals. At 25%, the shuffled pulses will be played 1/32 early, and at 75% they will be played 1/32 late.

Shuffle Resolution

The shuffle resolution switch decides which pulses are affected by the shuffle control. If the shuffle resolution is higher than the rate of a pulse generator, then shuffle will have no effect on that generator. Possible values are 1/4, 1/8 and 1/16.

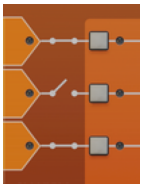
Velocity

The velocity knob will scale the velocities (signal levels) on all of Elementary's 15 gate outputs: The three pulse generator outputs and the 12 logic gate outputs. At 0%, all outputs will be quiet and at 100% they will all play at maximum velocity.

Inputs

Elementary has three inputs that go into the grid of logic gates. Each input has two indicators and a trigger button: The **External Signal Indicator**, The **Trigger Button** and the **Activity LED**.

External Signal Indicator



The “dip switch” to the left of each input tells you whether the input is receiving signals from the corresponding pulse generator or from its external CV input. The dip switch will open if you connect a cable to the external CV input, breaking the connection from the pulse generator. Note that the pulse generator can still be used via its dedicated signal output.

Trigger Button

By clicking and holding the trigger button you can force the input to go High, regardless of what signal is being used. The trigger buttons can be automated in Reason’s sequencer.

Activity LED

This LED will light up green whenever the input signal is High, including when the trigger button is held down. If you force the inputs to mute via MIDI (see further down), the LED will light up red instead.

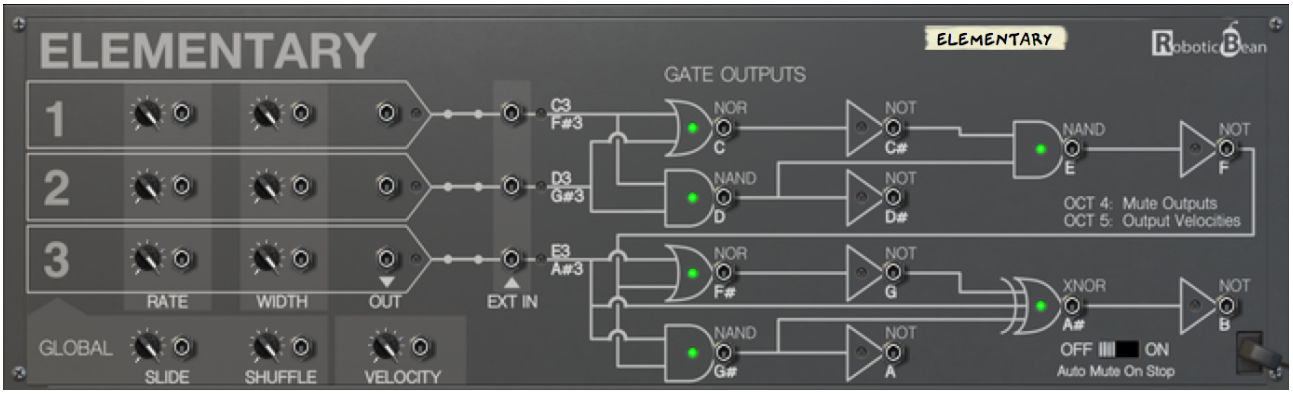
Logic Gates

Each of Elementary’s logic gate is marked with both the name of each gate type, as well as the corresponding symbol. The lines in the diagram show how the gates are interconnected.

Each gate has an activity LED that shows the current state of the gate. Each LED has six possible states:

Off	There’s no signal on the gate output, and there’s no cable connected to it either.
Faint green	There's a CV cable connected to the gate output, but there’s no signal on it right now.
Strong green	The signal on the gate output is High, at full velocity (but scaled by the velocity knob).
Red	The gate output is currently being muted via MIDI.
Faint yellow	The gate output’s velocity is being overridden via MIDI (and scaled by the Velocity knob), but there’s no signal at the moment.
Strong yellow	The signal on the gate output is High, but the velocity is being overridden via MIDI (and scaled by the Velocity knob).

The Rear Panel



The rear panel has almost the exact same layout as the front panel. In particular, notice that all the LEDs and logic gate diagrams stay exactly in the same place when you flip the rack around. We've done this to make it easier to find and connect patterns that you like.

For a detailed explanation of the LED states, see the previous section.

Modulation CV Inputs

Each knob on the front panel has a corresponding CV input with trim knob on the rear panel. The received CV signal will be attenuated by the trim knob, and the value will then be added to the front panel setting.

Pulse Generator Outputs

Each pulse generator has a separate CV output, so you can tap the signal before it goes into the logic gates.

External CV Inputs

Each trigger button on the front panel has a corresponding external CV input on the back panel. When you connect a cable to this input, the connection between the input and the corresponding pulse generator is broken, as illustrated by the dip switch in the circuit diagram.

Elementary will interpret the external CV signal as High if the value is greater than 0, otherwise it is interpreted as Low.

Logic Gate Outputs

As for the logic gates, each one has its own separate CV output, and they can of course all be used in parallel. The value on the output gate is determined by the global Velocity setting (via the knob and CV input), and in some cases also by incoming MIDI notes.

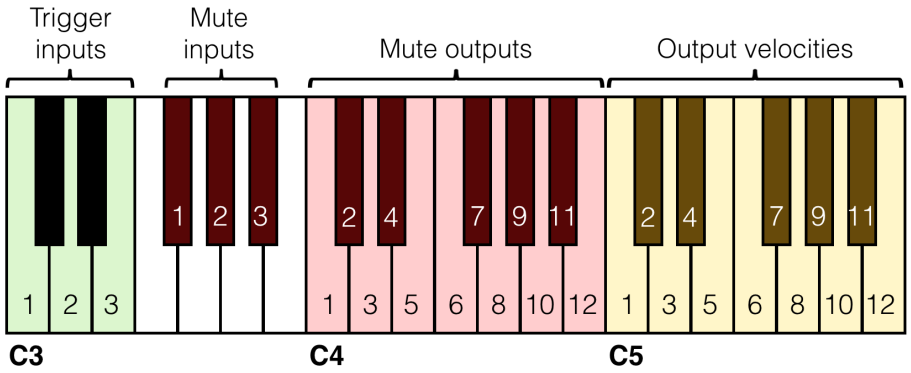
Auto Mute on Stop

When enabled, this switch will ensure that all CV outputs on Elementary output 0 while the transport is stopped. This is useful when you're processing free running signals such as external LFOs.

MIDI Note Overrides

You can also override the inputs and outputs of Elementary using MIDI notes. To do this, you can either right-click on Elementary and choose "Create track for Elementary", or you can put Elementary in a Combinator and check the "Receive Notes" box in the Combinator programmer.

Elementary will respond to incoming MIDI note messages in the following manner:

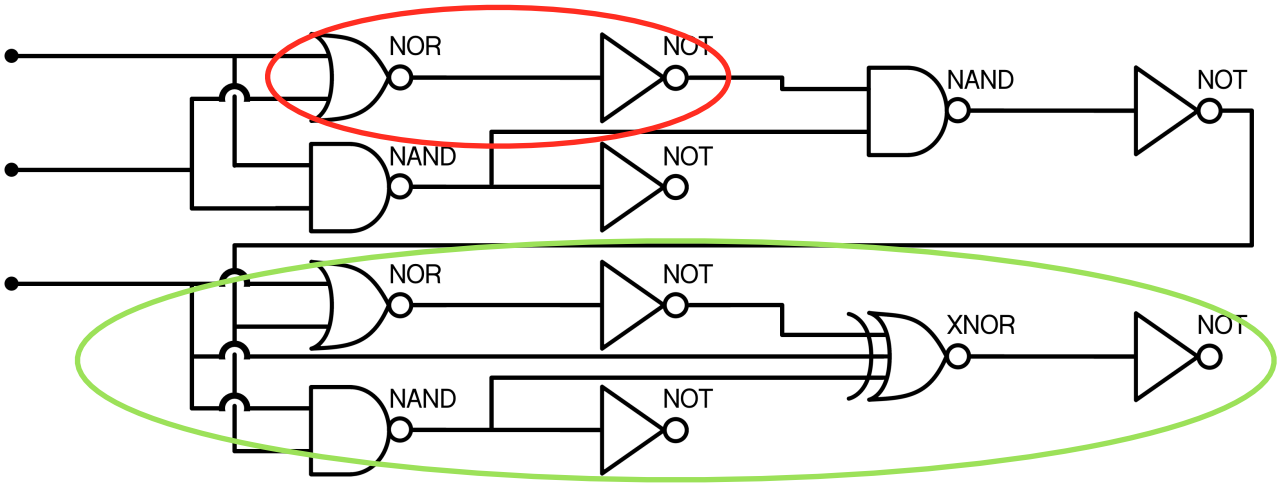


Range	Action	While pressed
C3, D3, E3	Trigger inputs	The input state is forced to High.
F#3, G#3, A#3	Mute inputs	The input state is forced to Low.
C4 - B4	Mute outputs	The gate output state is forced to Low.
C5 - B5	Output velocities	The velocity of the gate output is set to the velocity of the MIDI note. The velocity is only applied to pulses that are triggered while the key is pressed.

The Logic Gates Explained

The gate configuration of Elementary may look a bit intimidating at first, but once you understand the structure it's not that complicated. First we can note that the gates are organized in six pairs, each consisting of one of NAND, NOR and XNOR paired with a NOT gate.

The NOT gate, as we will see below, inverts the value it receives, so it will always create a perfect complement to its paired gate.

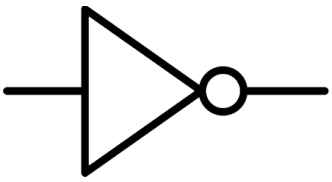


Second, you can also see the logic diagram as consisting of to halves, the upper and lower. The upper half is only affected by inputs 1 and 2. The lower half is affected by the output of the upper half combined with input 3.

Descriptions and Truth Tables

Here follow some details about the logic gates used in Elementary.

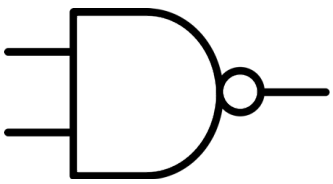
NOT



NOT takes a single input and inverts the value, so an input value of 0 will result in 1 and vice versa. This is used to create the complement to another gate.

In	Out
0	**1**
1	**0**

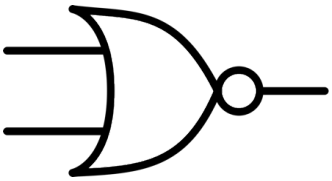
NAND



The NAND gate (Not And) will return 1 as long as not all inputs are 1.

In A	In B	Out
0	0	**1**
0	1	**1**
1	0	**1**
1	1	**0**

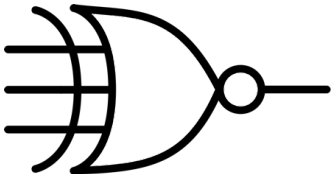
NOR



The NOR gate (Not Or) will produce a 1 as long as all inputs are 0.

In A	In B	Out
0	0	**1**
0	1	**0**
1	0	**0**
1	1	**0**

XNOR



XNOR means Exclusive Not Or. In our case, since we're using three inputs, it's also called the "even function" since it will return 1 only if there's an even number of 1's on the inputs.

In A	In B	In C	Out
0	0	0	**1**
0	0	1	**0**
0	1	0	**0**
0	1	1	**1**
1	0	0	**0**
1	0	1	**1**
1	1	0	**1**
1	1	1	**0**

MIDI CC Implementation

MIDI CC#	Property
15	Clock 1 Rate
16	Clock 1 Gate Width
17	Clock 2 Rate
18	Clock 2 Gate Width
19	Clock 3 Rate
20	Clock 3 Gate Width
21	Clock 1 Enable
22	Clock 2 Enable
23	Clock 3 Enable
24	Clock 1 Invert
25	Clock 2 Invert
26	Clock 3 Invert
27	Trigger 1
28	Trigger 2
29	Trigger 3
30	Shuffle Amount
31	Shuffle Resolution
33	Slide
34	Enable Outputs
35	Velocity

Remote Map Template

```
// Remote Map template for Utilities  Robotic Bean: Elementary
// Version 1.0.1
Scope Robotic Bean     com.roboticbean.Elementary

//    Control Surface Item  Key   Remotable Item   Scale Mode
//Map _control_         Enable Outputs
//Map _control_         Slide
//Map _control_         Shuffle Amount
//Map _control_         Shuffle Resolution
//Map _control_         Velocity

//Map _control_         Clock 1 Enable
//Map _control_         Clock 1 Rate
//Map _control_         Clock 1 Gate Width
//Map _control_         Clock 1 Invert

//Map _control_         Clock 2 Enable
//Map _control_         Clock 2 Rate
//Map _control_         Clock 2 Gate Width
//Map _control_         Clock 2 Invert

//Map _control_         Clock 3 Enable
//Map _control_         Clock 3 Rate
//Map _control_         Clock 3 Gate Width
//Map _control_         Clock 3 Invert

//Map _control_         Trigger 1
//Map _control_         Trigger 2
//Map _control_         Trigger 3

//Map _output_          Gate LED 01
//Map _output_          Gate LED 02
//Map _output_          Gate LED 03
//Map _output_          Gate LED 04
//Map _output_          Gate LED 05
//Map _output_          Gate LED 06
//Map _output_          Gate LED 07
//Map _output_          Gate LED 08
//Map _output_          Gate LED 09
//Map _output_          Gate LED 10
//Map _output_          Gate LED 11
//Map _output_          Gate LED 12

//Map _output_          Clock LED 1
//Map _output_          Clock LED 2
//Map _output_          Clock LED 3

//Map _output_          Input LED 1
//Map _output_          Input LED 2
//Map _output_          Input LED 3
```