Vita

What is it?

Conway's Game of Life, invented by John Horton Conway in 1970, is probably the most recognized form of cellular automata. Simple rules are followed to turn cells on (alive) or off (dead), resulting in complex, often surprising patterns. This is the inspiration and basis for Vita.

At its most basic (and default settings), Vita is an 8x8 grid of outputs that are activated at a constant +5 when the associated cell in the simulation is alive. The output and display represent the current generation in Conway's Life. You create an initial pattern, iterate through the generations based on the rules of Life, then the output jacks and display change to reflect the current generation, and you make music or noise or pretty graphs or whatever is inside you crying to get out. (Btw, you should probably have that looked at by a medical professional).

Vita can also be flipped onto its head and become an 8x8 grid of inputs that contribute (or not) to a single output based on whether their associated cell is alive (or dead). Think of a 64X1 switch. In a grid. That is controlled by Conway's Life. Useful? Not sure. Interesting? I hope so!

Editing the current state

Clicking the cell state display activates an inactive cell and deactivates an active cell. The RAND button and/or trigger randomizes the current state of the cells. The CLEAR button and/or trigger clears the current state (i.e., sets all cells to inactive).

Setting the initial state

The STORE button remembers the current state as the initial state. This is important for both getting back to this state via the RESET button and/or trigger jack to restart the simulation as well as recalling this state from a saved patch. Only the stored state, not the current state, is remembered and recalled from a saved patch

Iterating through generations

The NEXT trigger and button iterate through generations, reflecting the generational state in both the output jacks and the display. Some patterns repeat indefinitely. Some patterns die quickly. Experiment. Or look up classic patterns. Or both! The edges wrap around, so things like the classic glider pattern will move around forever.

Resetting to the initial state

The RESET trigger and button reset to the initial state. Recall that this is the last state remembered via the STORE button.

End State triggers

Extinct – A short +5 pulse is produced when iterating through generations results in a generation with no cells alive.

Rpt – A short +5 pulse is produced when iterating through generations results in a repeating pattern. The number of past patterns remembered is controlled by the slider. The default is 1, which means only a static pattern will be detected. There is an arbitrary max of 20, but the number should only be set as high as necessary as more remembered patterns means more processing each generation transition.

Grid view

The GRID button activates a view for more easily matching cells on the grid to their associated jacks. Labels are added to the jacks, and grid lines and labels are added to the cell display.

Seq or Switch toggle

This puts the module into sequencer (seq) or switch mode. In sequencer mode, the grid is a set of outputs where each jack represents the state of its associated cell in the current generation. In switch mode, they are inputs that contribute to the output if they are alive in the current generation. Note that you cannot switch modes with any cables plugged into the grid. This could harm the sensitive circuity and is thus not allowed.

Sequencer output modes

Alive – The output jacks are +5 when the associated cell is alive and 0 when not.

Dead - The output jacks are 0 when the associated cell is alive and +5 when not. This is the inverse of Alive.

Birth – Newly birthed cells, cells that went from not alive in the previous generation* to alive in the current generation, are represented by +5 in the output. All other cells are 0.

Death - Newly deceased cells, cells that went from alive in the previous generation* to not alive in the current generation, are represented by +5 in the output. All other cells are 0.

* The previous generation is the generation immediately prior to the current generation when iterating. When resetting, the previous generation and the current generation (initial state) are the same for purposes of detecting births and deaths. Thus there are no births and deaths detected in the initial generation.

Switch output

The jack in the switch section will have a value representing the live cells based on the following two toggles.

Average – When this is on, the inputs are averaged (added and then divided by the number of inputs). When it is off, they are just added. When off, the output can get very high very fast depending on the number and amplitude of potential inputs. Forewarned is forearmed.

Declick – When this is toggled, the output is faded in and/or out very quickly when necessary to avoid clicks. This is often useful for sounds. Maybe not for LFOs. It's up to you.

Notes

Due to UI refresh rates, the cell state display can appear to be slow or frozen when running through generations very quickly (as in past normal LFO rates and into the audio range). I could make the UI refresh rate faster at the cost of CPU usage or add a settable refresh rate, but I'm not sure if this is a real use case. The outputs are not dependent on the visual display and should keep going.

Further reading

For a discussion of Conway's Game of Life, see https://en.wikipedia.org/wiki/Conway%27s_Game_of_Life.

For more discussion and a gazillion patterns, see https://conwaylife.com/wiki.

Final thoughts

As with all my modules, I sincerely hope you find Vita useful, inspiring, or, at least, fun. If you have any suggestions for improvements, please make your ideas known through my Cherry Audio forum. Also, if you build anything you think is cool with Vita or any of my modules, let me know! I'd love to see how people use them.

Thank you!

borkman