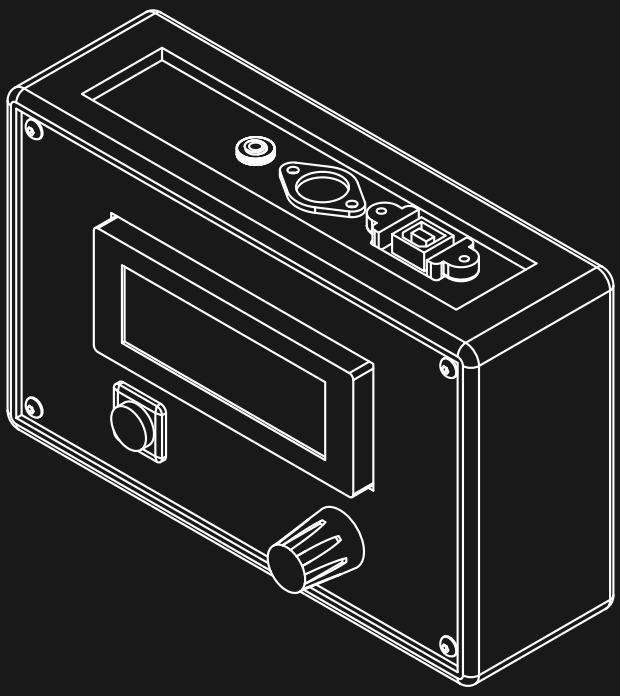
cVert

A truly random MIDI controller



What?

cVert (see-vurt) is a different type of MIDI controller.

Rather than imitating a traditional instrument, cVert generates random notes at random times, based on the radioactive decay of unstable elements.

cVert contains a Geiger-Müller tube, which detects radioactivity. It uses this as an input to generate MIDI notes. Since the decay of unstable isotopes is inherently unpredictable, the notes which are generated are truly and completely random.

cVert can limit the notes it generates to suit a specifc scale or range, according to a series of user selected parameters.

cVert also features a 32 step sequencer. The steps of the sequencer are filled with randomly selected notes, one of which will change each time a a radioactive decay is detected.

What for?

cVert can be used for a huge range of weird things. Here are some examples for inspiration.

WARNING: DO NOT MESS WITH RADIOISOTOPES UNLESS YOU KNOW WHAT YOU'RE DOING. Stick to just background radiation if you're not completely sure.

Music:

Interesting results can be produced by using cVert in sequencer mode to control a drum machine, or to play segments of a chopped up sample, like an amen break or a vocal clip.

In single note mode, cVert can generate ambient sounds. Using it to play a reverbed piano provides relaxing, contemplative background music.

cVert can also be used to add an unpredictable element to a live performance. Utilising the MIDI out feature enables cVert to be easily integrated into an existing live setup. Please consider the experimental nature of cVert when doing this, as the MIDI out is not isolated and may be unreliable.

Generative art:

When combined with the right software, cVert can be used to produce random visual art.

Using the "MidiBus" library and Processing, cVert can provide a random input for a generative artwork. In fact, all of the background art in this zine was generated using this technique! cVert can also be used with platforms such as openFrameworks or Cinder, or to control web-based applications via the Web MIDI API.

If the range of possible pitches and note numbers (0-127) is not big enough for your application, a recommended technique is to use a counter which constantly increments up to your desired maximum value. cVert can then be used to stop the counter when a note is played, resulting in a random number in a useful range.

Weird stuff:

If you really want to dive into the nitty gritty of what cVert can be used for, Pure Data is a graphical programming environment which lets you process MIDI data on the fly. It can do lots of fun things, including live audio and visuals.

How is it made?

cVert is an open-source project, so all of the design files are available free of charge so that you can make your own.

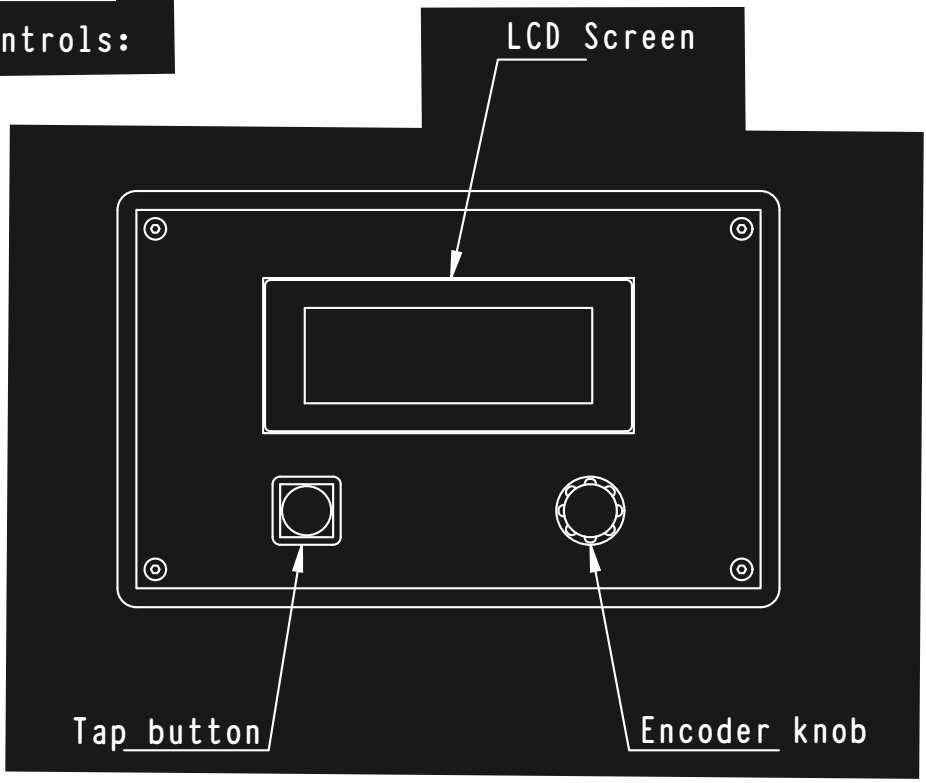
The functional part of cVert is an Arduino Pro Micro, a cheap Geiger counter module and a few external pars such as a 555 timer and some resistors. While the components are fairly simple, having a basic knowledge of electronics (or help from someone who does) is recommended if you want to build your own.

A custom PCB is available for cVert to make assembly simple, however it can also be built on perf board. This way cVert can suit the needs of individuals who want to add features or change components, whilst being permanent enough for a regularly used device.

The case for cVert is designed to be 3D printed and laser cut. While these technologies are becoming more accessible via hackerspaces etc, they are not yet available to all. The internals of cVert can therefore be fitted into a standard ABS project box, available from a range of online stores (aluminium not recommended as it may attenuate the radiation).

How do I work this?

Controls:

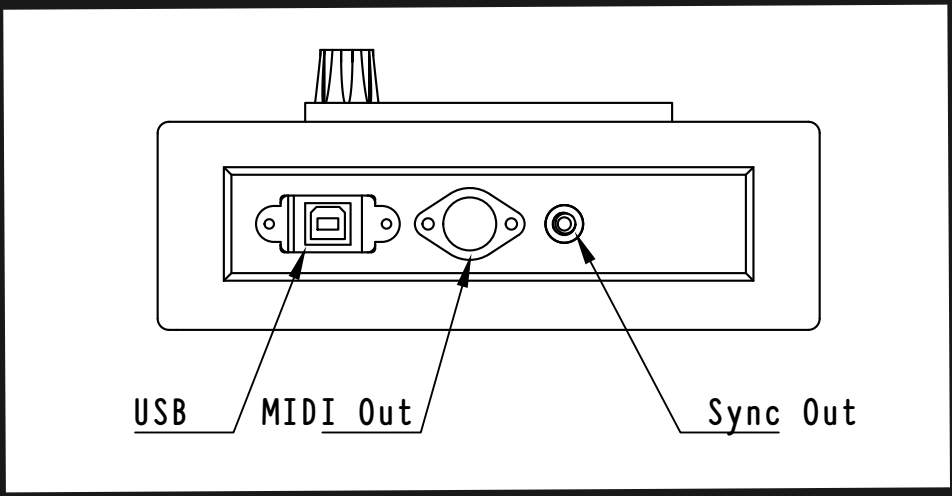


The front panel features a 20 x 4 character LCD display, an encoder knob and a tap button.

Turn the encoder knob to navigate the menu or to change a parameter, and click it in to select. Clicking and holding the encoder knob will clear all steps from the step sequencer.

The tap button is used to simulate a decay event, i.e. to force cVert to generate a random note right now.

Rear Panel:



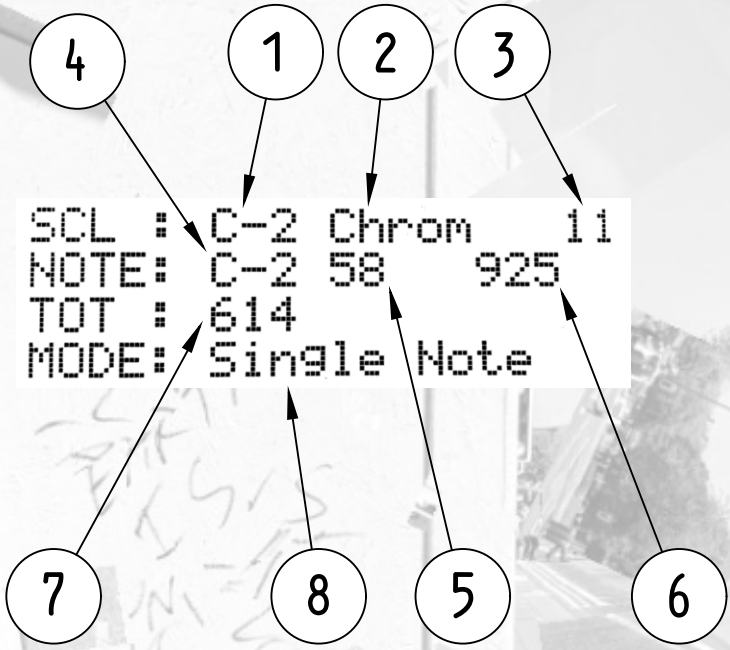
The rear panel features connections for USB, MIDI and for syncing external devices.

The USB plug is used to provide power to cVert and to communicate with a computer when used as a USB MIDI controller.

The 5 pin DIN socket is used to provide MIDI out to external hardware synthesisers etc. cVert still needs to be powered via USB (e.g. a phone charger) when using hardware synthesisers.

The 3.5mm jack provides a 5V clock signal (0.5ms high pulse, tip positive) for syncing cVert with external hardware.

Home Screen:



1. Root note of the current scale
2. Current scale type
3. Current octave range
4. The last note which was played
5. The velocity of the last note
6. The length (ms) of the last note
7. The total number of decays (notes played)
8. The current operating mode (single note or sequence)

Menu Options:

Click the encoder knob in when at the home screen to enter the menu.

HOME - Returns to the home screen.

Mode - Switches between single note and sequencer modes.

Scale Type - Change the scale which notes are selected from. "None" means the same note is always played. Use "Chromatic" to play all notes within the selected range.

Root Note - Selects the root note of the scale, from C-2 to G8.

Range - The number of octaves of the selected scale to select notes from. 0-11.

Min/Max Velocity: Change the upper and lower limits of the range of velocities to play notes at. Set these to be equal for no velocity variation.

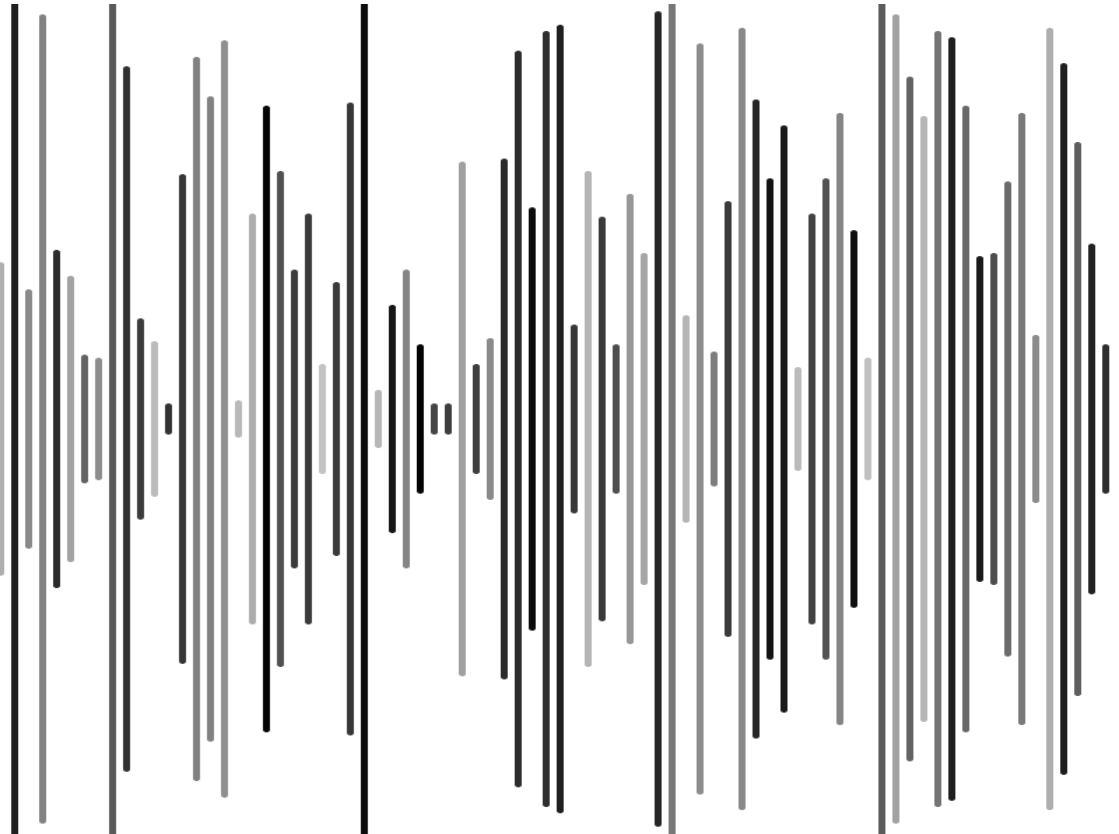
Note length: The length (in ms) to play each note for in single note mode.

Sequencer steps – The number of steps to use when in sequencer mode. 4, 8, 16 or 32.

Sequencer BPM – Sets the tempo in beats per minute of the sequencer. Only approximate due to microcontroller limitations, use MIDI notes if true synchronisation is needed.

Sequencer divider – number of times per beat a note will be played.

Brightness – Adjusts the brightness of the case LEDs, if present.



More Info

The source files for cVert are available online, and include:

- 3D case files
- Laser cut panel files
- Bill of materials including supplier links
- Electronic schematic
- Arduino code
- Mechanical drawings
- This zine!

These can be obtained from:
github.com/walkerdanny/cVert

As cVert is open source, community contributions are welcome! If you feel like you have something to add, be it a code improvement, extra feature, correcting a mistake or something else, feel free to make a pull request!