

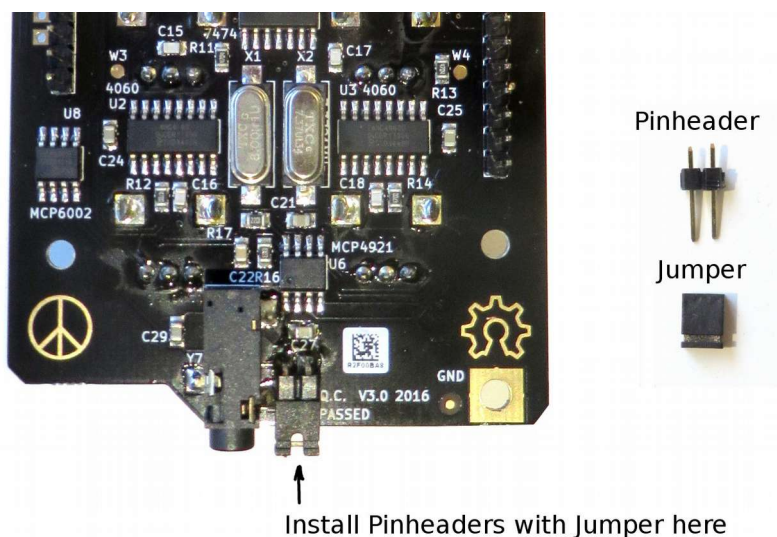
## How to set up the Open.Theremin V3 for Control Voltage

The Open.Theremin is a stand alone instrument that generates sounds in different waveforms when played. Eight waveforms can be selected through the TIMBRE knob on the board. These waveforms can be reprogrammed using different software available online. To even further expand the possibilities the Open.Theremin can be set in "Control Voltage Mode" to control any sound synthesizer with an analog control voltage input (CV).

When in control voltage mode the signal of the pitch antenna is translated into an analog signal that will be output through the audio-jack on the board. The audio sound will not be available anymore. However you can always reinstall the original software to go back to normal audio-mode. The volume signal is not available on control voltage for now.

So here is how to set the Open.Theremin to control voltage mode:

1. You need to solder a pair of pinheaders to the free pads (P5 + P6) next to the audio-jack and install a jumper. You can find these jumpers in many electronic stores. Alternatively you can solder in a switch or just a short piece of cable to make the short between the two pads.



2. Now you need to re-upload the software with the modification for the control voltage mode. With the newest Open.Theremin V3 software loaded in the Arduino IDE go to the tab named "build.h" and change the following line from a 0 to a 1

```
#define CV_ENABLED 1
```

Now re-upload the software (for details see the Open.Theremin V3 instruction manual).

Your theremin will now generate analog values in the range of 0-5 V on the left channel of the audio jack. Maybe you want to recalibrate the theremin by pressing the FUNCTION button for 3 seconds. The PITCH and REGISTER button can still be used to change the zero beat and sensitivity respectively.

For more information, documentation and source code of the open design go to:  
[www.gaudi.ch/OpenTheremin](http://www.gaudi.ch/OpenTheremin)

Open.Theremin Instruction by Urs Gaudenz, licensed under a Creative Commons Attribution 4.0 International License.  
25. Oct. 2016



## Additional advanced instructions:

If you want to adapt the output signal to your specific needs you can do this by modifying the following lines of code (in the “ihandlers.cpp” tab):

```
#if CV_ENABLED                // Generator for CV output

vPointerIncrement = min(vPointerIncrement, 4095); // vPointerIncrement is the signal of the pitch antenna
mcpDacSend(vPointerIncrement); //Send result to Digital to Analogue Converter (audio out)

#else //Play sound
```