



HEXINVERTER.NET

presents

jupiter storm

cosmic noise oscillator



Assembly Manual v1.0

Introduction

This manual exists to aid DIYers in creating their own Jupiter Storm module from a bare PCB from hexinverter.net

Please note that this is an advanced project and a lot of mechanical and electronics ability is taken for granted. If not purchasing a panel from me, you will have to craft your own panel and of course decide how to mount the PCB to the panel when you are done. Please consult the Muffwiggler synthDIY forum for information on how to do this sort of thing -- I will NOT be explaining these things in this manual (or any that I provide) and if you email me asking, I will tell you to go to Muffwigglers anyway as there are countless topics about these very things already in existence there :)

If you are wondering what Jupiter Storm is and what features it has, I suggest reading the User Manual over at the hexinverter.net project site ([click](#)), as these things are detailed in that manual (not this one).

Technical Notes

Jupiter Storm consumes approximately 30mA at +/-12V. It will run on 15V systems without any changes as well.

The Noise Core in Jupiter Storm is based around three square wave CMOS logic VCOs. There are no microcontrollers used in the design.

Because the CD4046 ICs that generate the square waves swing all the way from 0->15V and back at audio frequencies, there is a risk of coupling noise into the power supply due to the strain this high voltage oscillating imposes on the system's supply, ESPECIALLY if your system's supply is based on switching regulators and not linear (ie: TipTop Zeus) or is already heavily taxed (ie: reaching maximum current capacity with the modules you have installed already) or inadequately filtered (not enough filter capacitance). If this is an issue in your system, I suggest moving around Jupiter Storm to a different level/cabinet and trying to improve your grounding configuration, or, increase your power supply capacity if all else fails. In my system with a home built linear supply, I notice no noise, but, some people have reported noise on their power rails from the module, so, it's up to you to track it down and fix it if so!

Notes About Parts/Substitutions

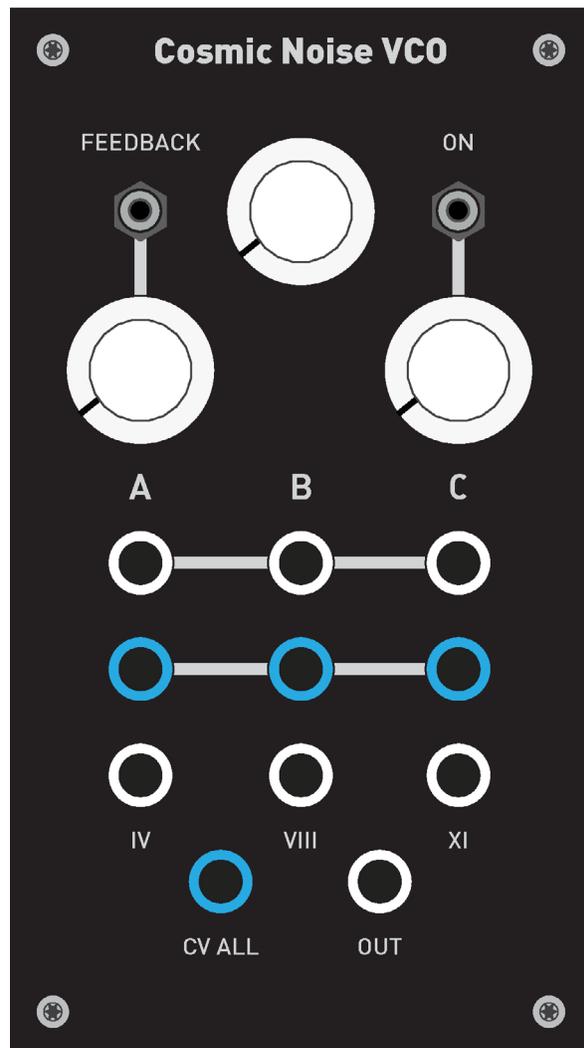
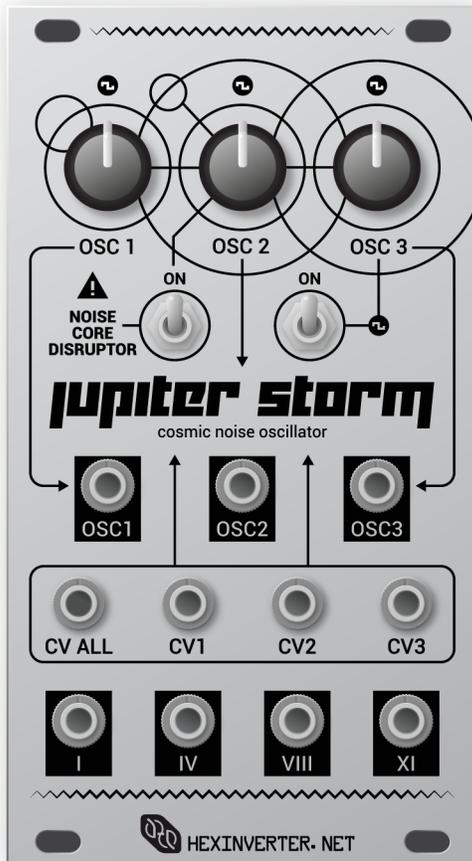
The way the Noise Core Disruptor (or “feedback”/”FBK”) mode works in this module is by creatively abusing the circuitry in the Noise Core. While developing this module, I experimented with different brands of CMOS chips and discovered that the resultant characteristic of the sound varies quite a bit from different IC manufacturers!

In other words, you MUST use Texas Instruments brand ICs where indicated on the Bill of Materials or your module will sound pretty different. Feel free to experiment with different brands if you like. Some might not even work as expected at all! Let me know if you discover something cool :)

You might choose to use 2k tempco resistors in for RT1/2/3 in your build. I really do not think it is necessary -- actually, I think it is a waste of money and that your tempcos are better used elsewhere -- but, some might like to use them, so, I figured I would mention it here!

Same goes for the 2N3904 exponential converters -- you can match these in pairs if you like, but, I really think it is a waste of your time due to the nature of the circuit being NOISE!

Panel Renders



Here are some panel layouts that you may find useful for inspiration. Both of these panels (14HP eurorack on the left and 4U high dotNET modular format on the right) are/will be available in limited quantity from the hexinverter.net shop.

PLEASE NOTE: you will need sub-mini toggle switches if you use the 14HP eurorack panel. The part number for these switches is in the Google Docs BOM which is linked to from the hexinverter.net DIY project site. These are slightly smaller than the standard mini toggles you might have laying about. The alternative is that you can drill the 5.5mm holes out a bit to 6.5mm to use your standard mini toggles instead.

Wiring Guide

Note: Don't get too caught up with the output names. They are pretty much arbitrary, so name them whatever you want!

