Castle Rocktronics

000 – PWR

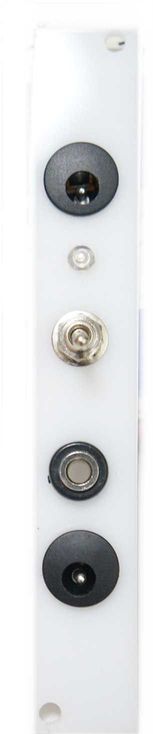
Power distribution & system overview

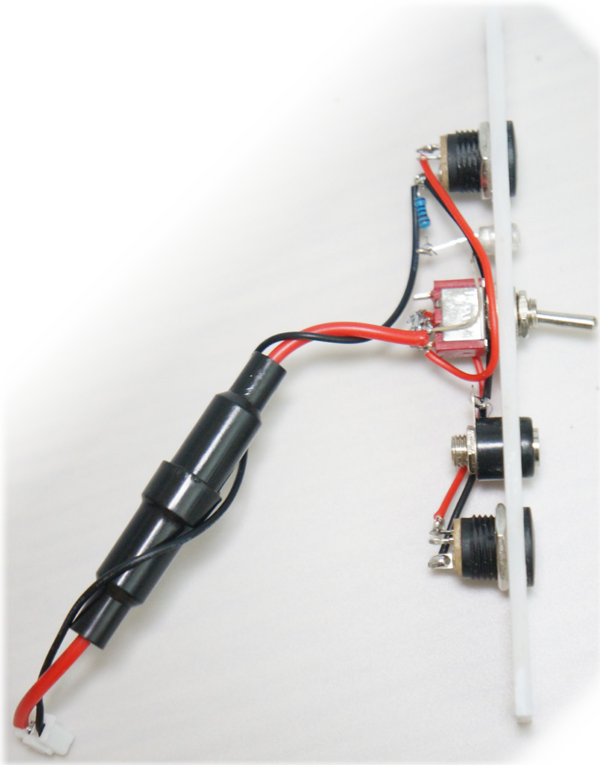
Comments, suggestions, questions and corrections are welcomed & encouraged:
contact@castlerocktronics.com

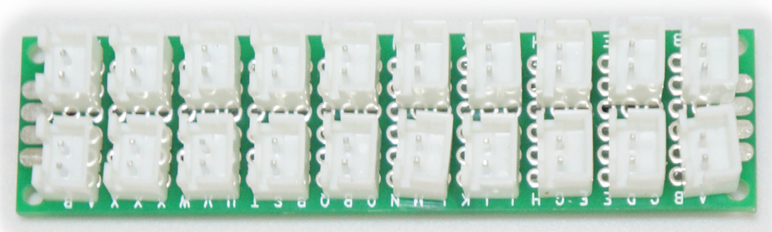
Table of Contents

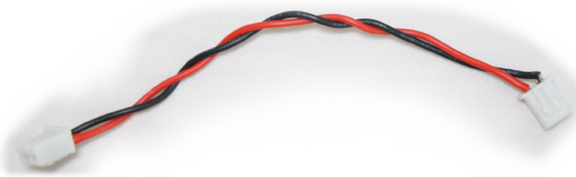
0. Build documents

0.1 Photos

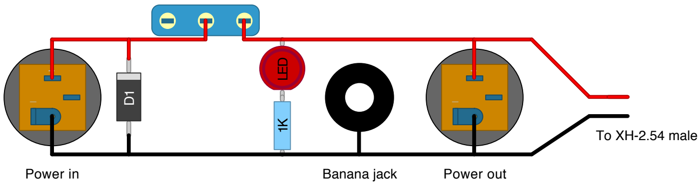




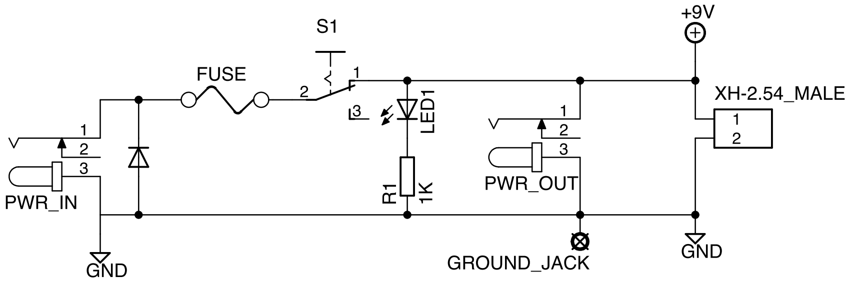


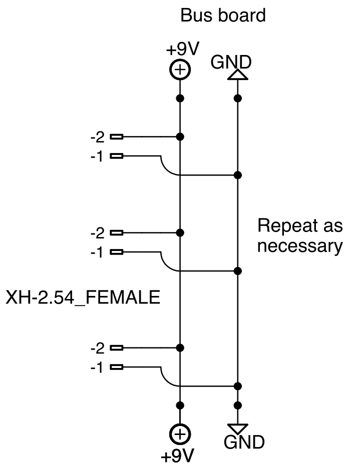


0.2 Layout



0.3 Schematic





0.4 Bill of Materials

Resistors

1K	*x1*

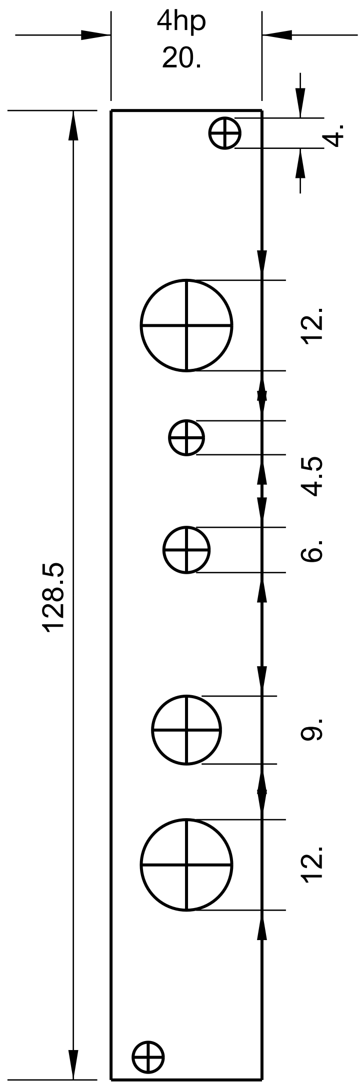
Diodes

LED	*x1*
1N4004	*x1*

Connectors

Banana Jack	
Black	*x1*
XH–2.54	*x?*
DC barrel jack	*x2*

0.5 Drill template



1. System overview

What this modular system is:

- Cheap
- Simple
- Low-power consumption

What this system is NOT:

- 1V/Octave
- Midi-compatible
- Entirely CV controllable
- Painstakingly engineered

1.1 Goals

There were two major goals for the project. The first was to get a modular synth without having to pay an arm an a leg.

I was making stompboxes and noisemakers but a modular was always the goal. I dreamed of a big DIY eurorack or serge system that filled a whole room ever since I started building. More than that though, I am impatient. I decided to go down the simple route so I could build more, quicker and learn along the way. I stumbled across the world of “Lunetta”/CMOS modulars – simple set ups that use and abuse logic chips to great effect, though at the cost of a few compromises. As a result, a lot of the inputs and outputs are DIGITAL. Not in the sense that they have gone through a microprocessor, but that they spit out squared waveforms that are either high or low, or they expect the same as input. Other modules will output analog signals that lie between 0-5V or they might also accept such signals into to modulate something like filter cut-off.

The other thing I wanted was for it to be easily transportable. Massive 4/5U panels lost out in favour of the eurorack 3U size, also because rails for the euro system were easy enough to come across.

The second goal was to learn about electrical engineering as best as I could. This was where I really hit a wall though. I found that the resources I wanted didn’t seem to exist. There was either too little detail, or the presumption that I was a seasoned engineer already. Eventually, I figured that the only way to have these materials was to make them myself. The result are these articles (but more on them later).

1.2 Powering modules

One of the main things I wanted to basically ignore was building a fancy power supply, I basically wanted to use a 9V DC wall wart I had kicking around. Though I stuck to my decision, there was something of an issue: headroom. If I was powering my chips at 9V then when I came to mix them together at an op-amp I’d meet a ton of clipping and active filters would sound nasty. The solution to this is that all circuits, except op-amps, get regulated to 5V. This has the advantage of giving op-amps more headroom over the incoming signal and also that 0-5V is a pretty standard range for CV in other systems: I could hook it up to a eurorack system later!

Similarly, in the theme of keeping it simple, I wanted to use a really easy system of connecting modules to the incoming power. This was easy enough to solve and I settled on “JST” style 2.54mm connectors *(see appendix)*.

JST is not actually the official name of this type of connector, just the colloquial name they are known by. JST actually make a wide range of connectors. The ones I chose to use come under XH series, which means they fit nicely into perfboard because they have the same spacing as DIP chips and sockets. They also plug in fairly snugly while also unplugging easily too, which is perfect for when you inevitably rearrange modules. Make sure to double, triple & quadruple check you keep the orientation for the headers the same! I make the wire carrying +9V red and put it to the left of the little plastic bit that sticks out at the front. This was the wrong choice. If I did it the other way I could use the mnemonic “red on the right” to stop me doing it the wrong way. I wasn’t smart enough at the time.

1.3 Colour coding

Similar to serge-style systems, I wanted to colour code my ins and outs to make live patching easier. Unlike the serge system however, I chose a slightly different coding system and tried to pick different colours for each different type of input, though I use just one colour for all outputs.

Red: (Output) The result the module is ultimately designed to spit out. Often sound, but also CVs, envelopes and triggers.

Black: (Input) Could be audio, CV or digital. Has a direct relation to the output. *Examples: filter input, mixer input, data input.*

Yellow: (CV) Envelopes, LFOs and other ANALOG control signals. Anything where the actual voltage matters. *Examples: Cut-off frequency, gain.*

Blue: (Clock) Digital. Not necessarily a fixed frequency, any input that causes the circuit to “step forward” by one on a rising or falling edge. *Examples: Counters, step-sequencers.*

Green: (Trigger) Digital. One-shot events that trigger on a rising or falling edge. *Examples: percussion, resets*

White: (Gate) Digital inputs that act according to whether they are high or low, rather than triggering on an edge. *Examples: Analog switches, inhibits, address inputs, envelopes.*

1.4 Construction

(Photos for everything mentioned in the following section are shown in the appendix.)

I opted for acrylic panels, just because they were cheaper than aluminium and easier to cut down to size. I got two 128.5x425x2mm white acrylic blanks – equivalent to 3U tall and 84HP wide. I use a right-angled ruler and a box-cutter to snap off what I need then file down the edges a little for some wiggle room.

When my perfboard has to be stacked I use nylon 12mm M2 spacers and M2 bolts, washers, nuts etc. Making a bunch of inter-board connections is done with male/female pin headers, the same as those used for eurorack power connections, but single row instead of double. I get large rows that I cut to size. Single inter-board connections for things like sharing a ground or a simple input are done with point-to-point wiring.

The drill templates are made in CAD software after measuring and double checking. I print out paper templates to make sure everything fits by cutting the holes with a craft knife and see if everything can fit.

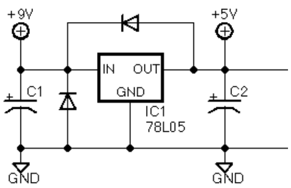
The holes for pots are drilled 2mm larger than necessary, to allow for some error. Washers keep the pots attached with decent reliability. The M3 bolts for attaching the panel to the rails have 4mm holes, also allowing room for error but you have to be a bit more careful. Washers help here too.

Banana jack and LED holder mounting holes are not made larger because these two aren’t board mounted, but also don’t have washers or a very wide lip. Where slide switches are used, I prefer to go with panel mounted ones and cut out what I can with a normal bit first, then finish the job with a little 2mm grinder bit.

Generally I prefer mounting my pots on my perfboard and then using the pots to attach the board to the panel. I hate point to point wiring so I try to keep it to a minimum. When there are no pots as part of the circuit though, I turn to some tiny little M2 right-angle brackets that are for making scale models. The type I bought have two M2 holes on one part, and a sort of elongated 2mm wide rectangle on the other. I chose these for the little extra wiggle room for error. I attach these to the panel with a spacer, which is attached by a hole on the panel itself. Then the boards are attached by an M2 bolt and nut combo to the L-bracket.

1.5 Common circuit elements

Some time should be taken out to explain these frequently used and crucial little building blocks so that I don’t have to dedicate space to them in each separate article.

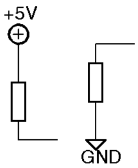


Voltage regulation

For all of the CMOS chip based circuits, I wanted to drop things down to 5v.

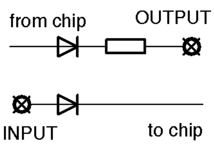
The 78L05 can produce a nice, fairly consistent 5V output so long as it is fed with something in the range of 7–20V. I liked this because it meant if I felt more hungry for headroom that I could use something other than the 9V wall wart I have and kick it up to 12V or even 18V. It also helps us filter a bit of noise out of out power rails. The other additions to the circuit are two capacitors, that store some spare charge to help smooth over fluctuations, and a diode which protects the regulator from getting fried if the output voltage goes higher than the input. It wont protect it from getting fried if you plug it in the wrong way though! Also make sure the large cap is on the input and the smaller on the out.

There is also another diode facing the "wrong way" across the power-rails. This is a polarity protection diode. It serves as a fail-safe for if you accidentally put the connecter in the wrong way or something goes wrong with the power coming it. It's put there so that if ground goes more positive than the power rail, then it'll never go high enough to damage anything, because the diode will keep the voltage pinned fairly low



Pull-up & pull-down resistors

Pull-up and pull-down resistors play a really useful role when working with CMOS chips. Sometimes we want to attach a banana jack to some pin so we can control it, but it might not always be in use. In these cases, we usually want to hold it high or low but still leave it controllable. This is where pull-up resistors come in to play. They keep our pin stuck at the voltage we want to keep it on or off, but when a high or low signal comes in through the jack, then it "overpowers" the resistor and brings the input to the voltage we want. I use 10kΩ pretty much across the board. Too large a resistor slows down the reaction to any incoming signals because of capacitive coupling with the input.

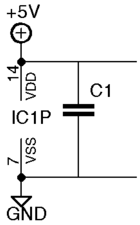


Diode protected inputs & outputs.

This resistor and diode pair on the outputs from CMOS chips help prevent any shorts that might occur and blow up the chip when we connect two inputs together. This can happen when we stack banana cables. If we tie two inputs together without the diode or resistor, then the high signal will try to force as much current as it can through wherever the low signal is. This surge can damage stuff and be a general nuisance. Sometimes the diode is left out, in the case of outputs that we may actually want to use to sink current. This is usually LFOs or sequncers and the like. The resistor will still help stop things shorting out.

An interesting result of this diode protection scheme is that when we do stack inputs, we create an OR gate.

CMOS chips can also blow if their pins are brought above or below the rails they are powered from. We can use diodes to stop negative voltages flowing in at all, and also to "clamp" the voltage to stop it going much further above the power rail (though that's not shown here). This is useful for interfacing other synths to this one as we don't have to worry about blowing things up, but it also means we don't have to worry if we use op-amps to build other oscillators or CV sources that go above 5V.



Bypass capacitors

These little guys might look completely pointless on the schematic, but you will notice they are always as close as possible to the either the ground or power pin as possible and then connect directly to the opposite rail. Bypass caps act a little like tiny batteries. Their purpose is to filter out noise on the power line by providing a little back-up charge.

1.6 The articles

I think this is maybe the real core goal of this project for me. Resources on electronics for synth building that were aimed at beginners seemed few and far between. Similarly, I know that not everyone cares about analysis even if I do. To this end, all the articles are split into a normally maths-free "explanation" section and a more in-depth "analysis" section. I also figure that to teach others, you have to truly understand something yourself. Forcing myself to write about things forces me to make sure that I actually know what is going on inside a circuit.

Basically, I wanted these articles to be the resources I wished were out there, but don't seem to be!

2. The Power Supply

Now we've glossed over what this project is about and got a feel for it, its time to build the only module you will definitely use 100% of the time: the power supply.

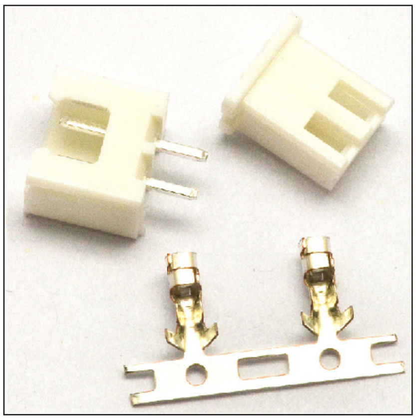
As I said before, I didn't actually want to put effort into designing or building a "real" power supply when I could just use a DC wall-wart. As a result, the power supply is not really a power supply at all, it's just a bus board with a few extra features:

- A fuse to stop anything going on fire (use one equal to your adaptors output – mine is 0.5A).
- A switch to turn it on and off.
- A protection diode to stop things getting fried by a DC jack of the wrong polarity being plugged in – I use centre negative, because that's what I had kicking about.
- An LED to let you know its on (I used a colour changing one :P)
- A daisy-chain output so that you can power external stompboxes or another one of these modulars from one power supply (assuming you have enough juice!). I also use it for powering my breadboard so that the ground is already shared between it and the modular.
- A ground connection for sharing grounds between other banana jacked modular systems and also for using an oscilloscope to poke around.

The bus board is just a bunch of the aforementioned XH-2.54 connectors on a piece of perfboard with their pins tied together, making sure to keep everything the same orientation. Modules are then plugged into this.

One last note: always quadruple check your power connections! Make sure wires from the bus board are the right polarity at both ends so that they don't cook any chips when you put them in the wrong way!

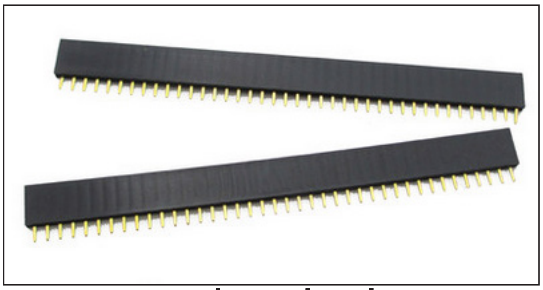
3. Appendix



XH-2.54 connectors

M2 spacers

Male pin header



Female pin header

Grinder attachment

LED holders

M2 L Brackets