Headtrip3

FX Type: **Multi-Effect**
Build Level: Advanced
Based On: mbp original

Last Updated: November 22, 2024 2:01 PM

The ***Headtrip3*** has the same multi-fx patches as the Headtrip2, but the audio portion of the circuit has been redesigned and it has a new PCB layout.

Overview

The **Headtrip** was originally released in 2018 and was one of the first forays into the world of FV-1 DSP effects for madbeanpedals. It featured three mbp fx patches, all of which were a bit on the weird side. For the **Headtrip2**, the design was expanded to a full 8 patches. The Headtrip3 still features the same patches but I decided to simplify the audio through portion of the design while reducing part count. I also removed the "Tails" feature from the previous version since its operation confused a number of builders.

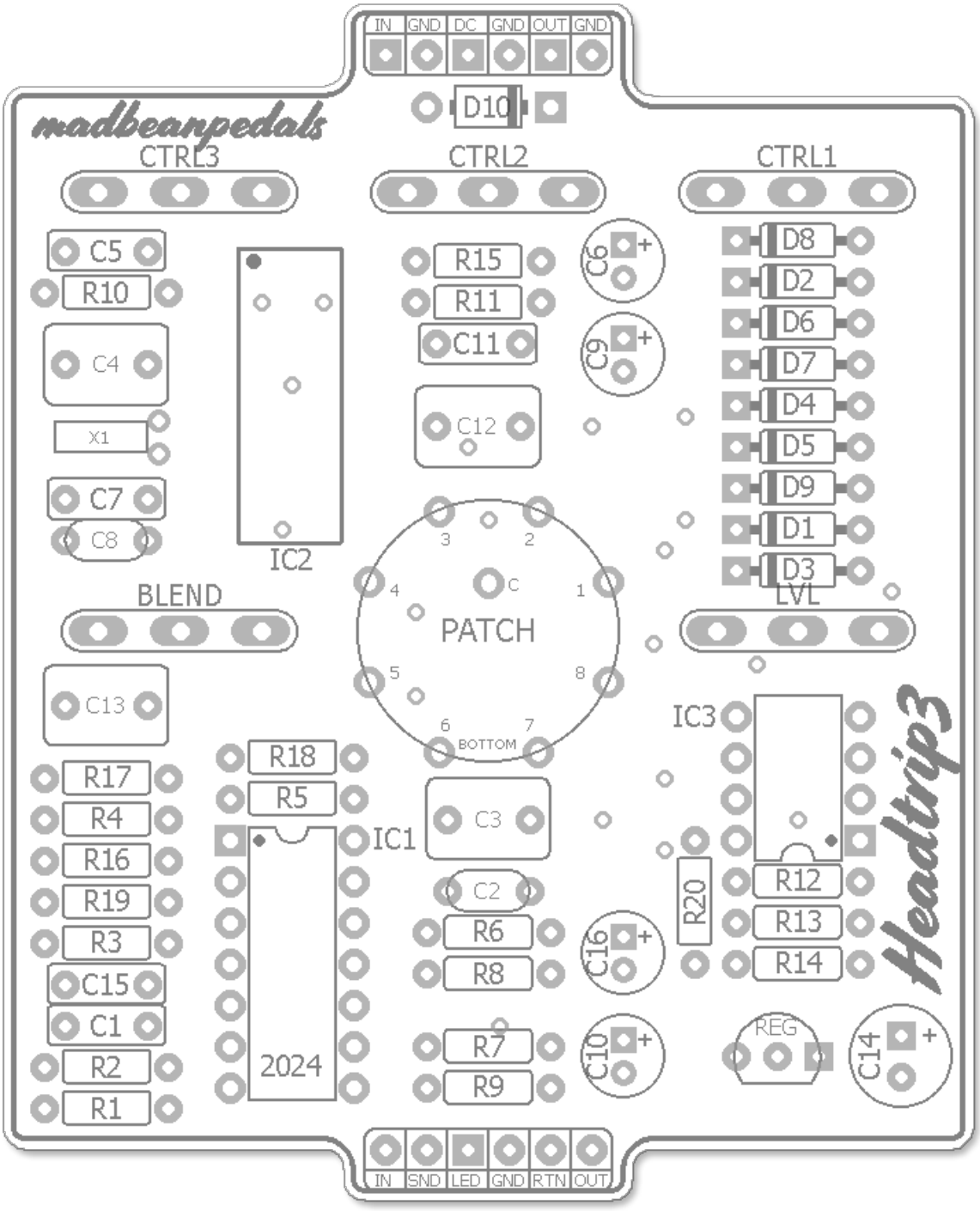
The main idea behind the Headtrip series is to offer FX that have practicality (or at least musicality) with the ability to dial in some strange or unusual sounds. IOW, this is not the project for "This is my phaser sound. Now, here is my chorus sound. Oh, look: it's my boring-a$$ sh*tty delay!" Rather I wanted to come up with ideas that hopefully push the envelope just a bit (the limitation being that all these patches were assembled in the wonderful SPINCAD program rather than hand-coded). While there are limitations with SPINDCAD, I feel like I've come up with a good set of patches that are fun but also applicable to a variety of musical settings.

While the Headtrip3 does include my original patches on the included EEPROM, you should be able to use any number of existing patches done by other designers so long as you have the tools to program your own chip.

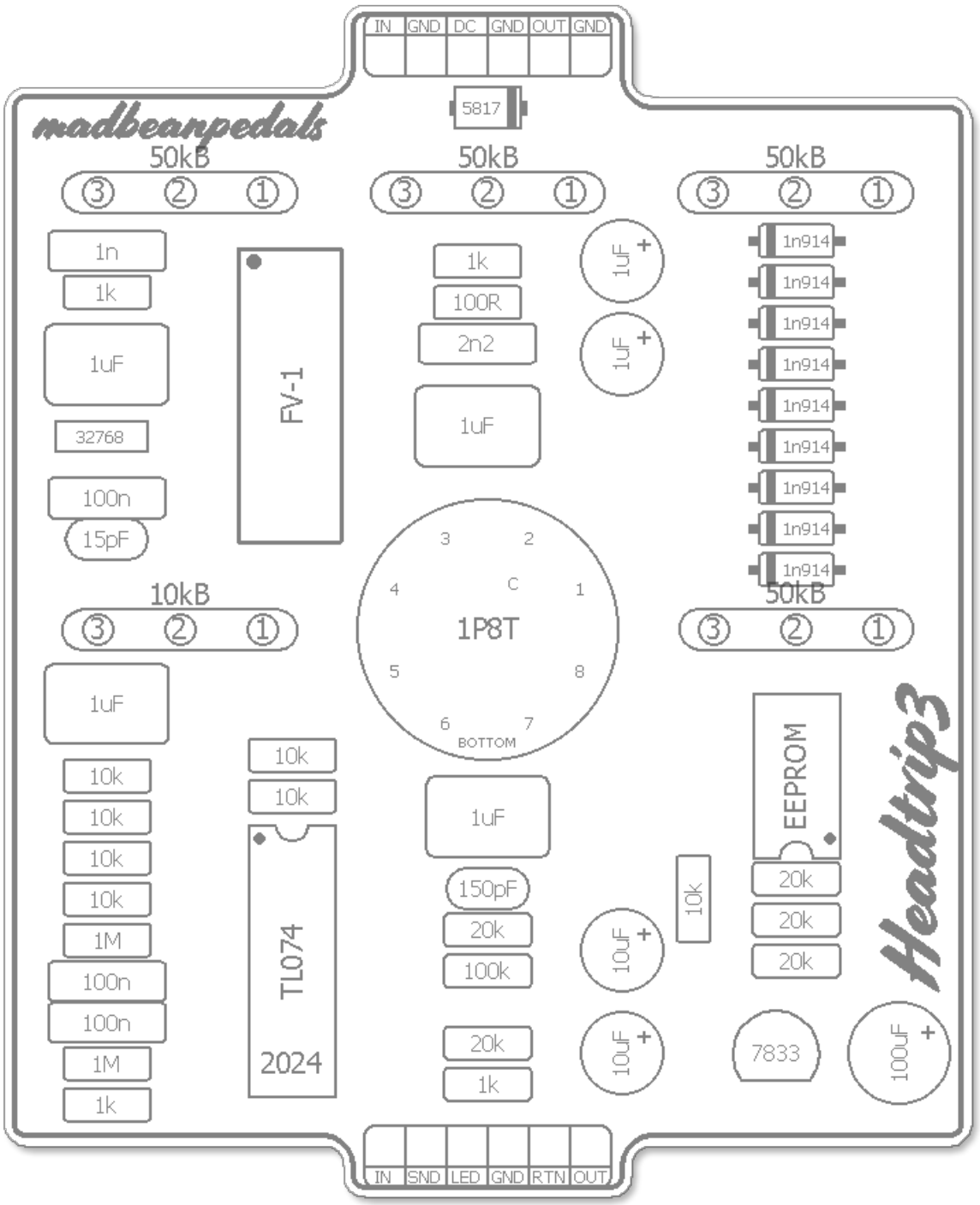
Controls

CTRL1-3: These pots control the various parameters of each internal patch. They will do different things depending on which patch is selected. See the list of patches and controls later in the doc.
BLEND: CCW - clean signal, 1/2 up - equal amounts of clean and wet signal, CW - wet signal only.
VOL: Output control for both clean and wet signals.
PATCH: This rotary switch selects which FX program is active. There are 8 to choose from.

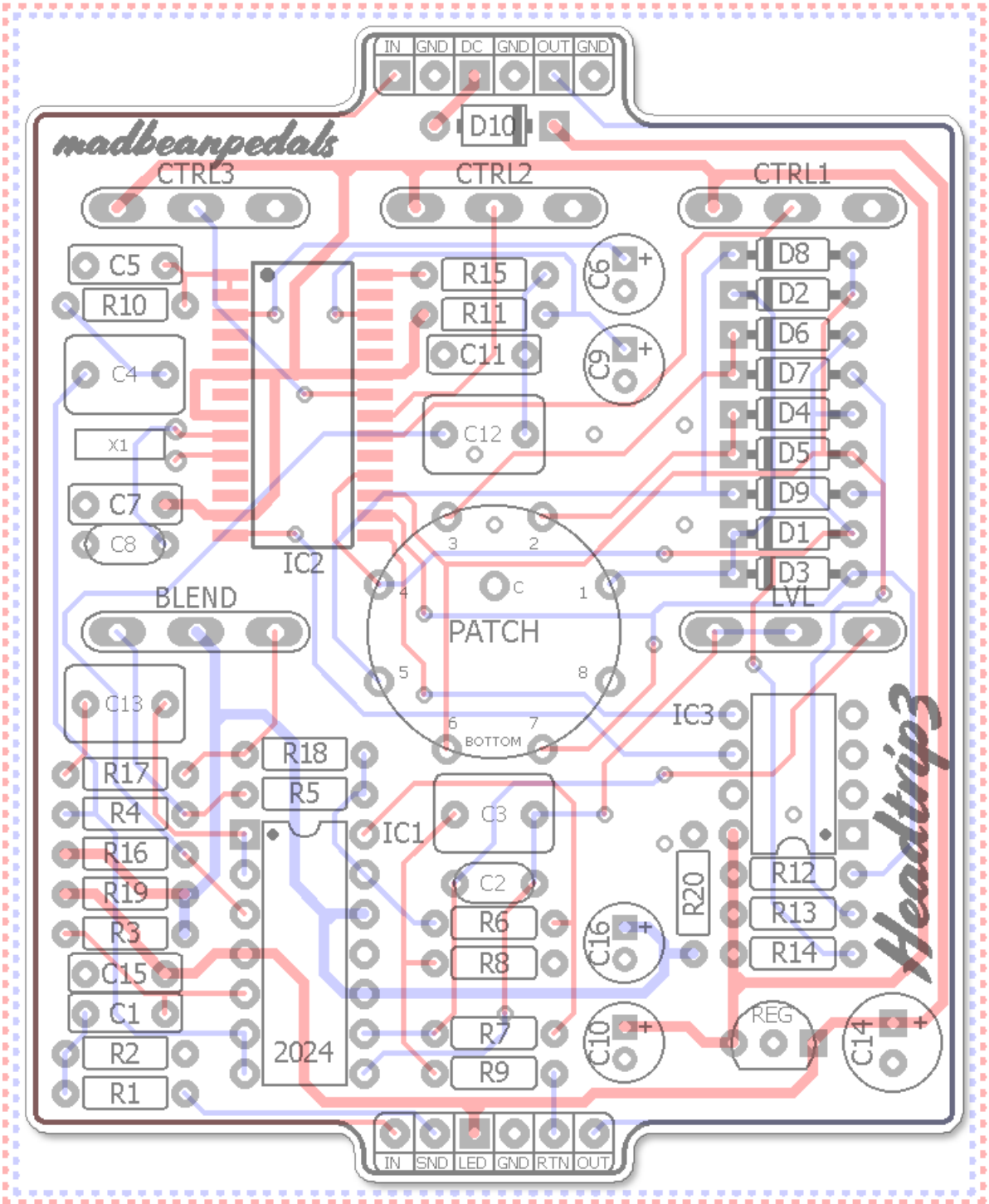
Parts Layout



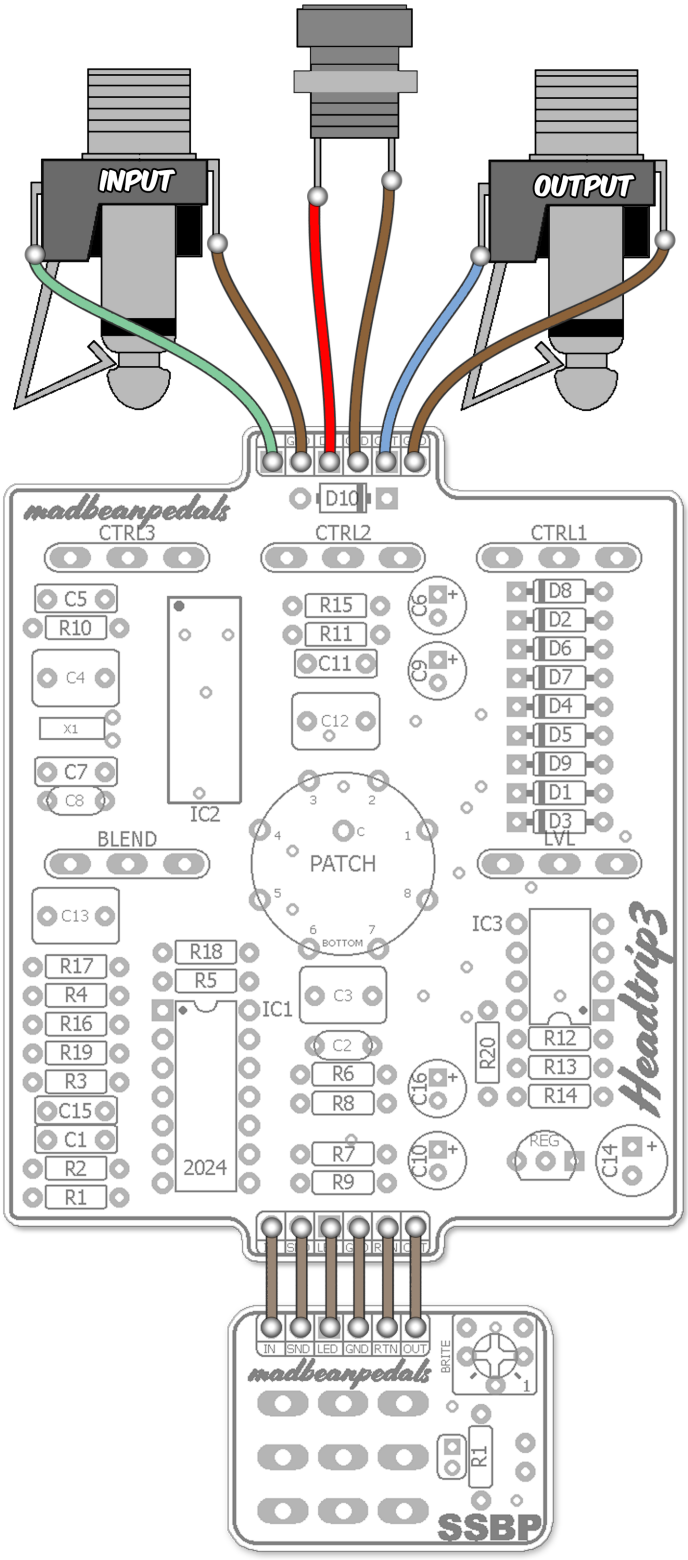
Component Values



Trace Layout



Wiring



The Headtrip3 is compatible with both the **SSBP** and **SSABP** bypass boards available at madbeanpedals. The wiring is the same for both.

B.O.M.

Resistors		Caps		Diodes	
R1	1k	C1	100n	D1-D9	1n914
R2	1M	C2	150pF	D10	1n5817
R3	1M	C3	1uF	IC	
R4	10k	C4	1uF	IC1	TL074
R5	10k	C5	1n	IC2	FV-1
R6	20k	C6	1uF	IC3	EEPROM
R7	20k	C7	100n	Regulator	
R8	100k	C8	15pF	REG	LM7833
R9	1k	C9	1uF	Crystal	
R10	1k	C10	10uF	X1	32768
R11	100R	C11	2n2	Switch	
R12	20k	C12	1uF	PATCH	1P8T
R13	20k	C13	1uF	Pots	
R14	20k	C14	100uF	BLEND	10kB
R15	1k	C15	100n	CTRL1	50kB
R16	10k	C16	10uF	CTRL2	50kB
R17	10k			CTRL3	50kB
R18	10k			LVL	50kB
R19	10k				
R20	10k				

15pF:
https://www.mouser.com/ProductDetail/810-FG28C0G1H150JNT6
http://smallbear-electronics.mybigcommerce.com/capacitor-silver-mica-500v-10-pf-150-pf/

FV-1:
http://smallbear-electronics.mybigcommerce.com/ic-spin-semi-fv-1/

32768Hz crystal:
https://www.mouser.com/ProductDetail/815-AB26T-32.768KHZ
http://smallbear-electronics.mybigcommerce.com/32-768-khz-crystal/

L78L33:
https://www.mouser.com/ProductDetail/511-L78L33ACZ
http://smallbear-electronics.mybigcommerce.com/ic-l78l33acz/

TLE2074 (TL074 upgrade, if you want to spend the $$):
https://www.mouser.com/ProductDetail/595-TLE2074CN

1P8T:
http://smallbear-electronics.mybigcommerce.com/rotary-switch-miniature-1p8t/
https://stompboxparts.com/switches/1p8t-mini-rotary-switch/

Shopping List

Value	QTY	Type	Rating
100R	1	Carbon / Metal Film	1/4W
1k	4	Carbon / Metal Film	1/4W
10k	7	Carbon / Metal Film	1/4W
20k	5	Carbon / Metal Film	1/4W
100k	1	Carbon / Metal Film	1/4W
1M	2	Carbon / Metal Film	1/4W
15pF	1	Ceramic / MLCC	25v min
150pF	1	Ceramic / MLCC	25v min
1n	1	Film	25v min
2n2	1	Film	25v min
100n	3	Film	25v min
1uF	4	Film	25v min
1uF	2	Electrolytic	25v min
10uF	2	Electrolytic	25v min
100uF	1	Electrolytic	25v min
1n914	9		
1n5817	1		
TL074	1		
FV-1	1		
EEPROM	1	included with purchase	
LM7833	1		
Crystal	1	32768 Hz	
Rotary	1	1P8T	
10kB	1	PCB Right Angle	16mm
50kB	4	PCB Right Angle	16mm

Additional Hardware

(1) 125B enclosure
(2) 1/4” mono jacks
(1) Slim 2.1mm DC jack
(1) Standard 3PDT footswitch
(1) 5mm LED

Build Notes

Patch	Type	Name	Description	Ctrl1	Ctrl2	Ctrl3
1	Pitch	ChoirMouse	Octaver + Choir	Oct Down	Oct Up	Choir
2	Filter	Fourgan	Diphthong + Octaver	Envelope	Formant	Octaver
3	Modulation	Sinkhole	Whammy + Tremolo	Space	Rate	Depth
4	Delay	WhiteHole	Delay + Flanger	Time	Regen	Flange
5	Delay	Dreamtime	Delay + MultiFX	Time	Regen	MultiFX
6	Reverb	Spacebomb	Reverb + Flange	Reverb	Low Cut	FlangeBomb
7	Reverb	These Halls	Reverb + Choir	Reverb	Low Cut	Choir
8	Reverb	These Ruins	Reverb + PitchShift	Reverb	4th Down	5th Up

ChoirMouse: This is my take on the EQD Organizer™. The design of the Organizer and Headtrip2 are different so I've combined certain elements into one control. Ctrl1 is the octave down volume. Ctrl2 is the octave up volume. Ctrl3 does several things at once. As you turn it up, it sends the upper octave into a feedback loop with reverb and a slight delay. This creates some "shimmer" or a choir type effect. This control can be used when Ctrl1 and 2 are off.

Fourgan: Probably my favorite of the new patches. This is a diphthong type envelope filter which exploits low pass filtering to imitate the sound of a human voice. Ctrl1 sets the sensitivity of the envelope and is keyed off your picking dynamics. Ctrl2 pans between two diphthongs. CCW is "ae". CW is "u". In-between is a mix of both. Ctrl3 is where the real fun starts. It dials in upper and lower octaves so as it is turned up the whole effect takes on an organ-like quality. It's reminiscent of some of the EHX B9™ sounds, IMO.

Sinkhole: My take on the very familiar Whammy™ detune setting. It's a faux chorus sound created by pitch-shifting the incoming signal up and down slightly. It does not modulate on its own so it's not a true chorus. Ctrl1 sets the total amount of pitch shift up and down from min to max. Ctrl2 and Ctrl3 are Rate and Depth for a tremolo effect after the pitch shifting.

Whitehole: This patch has a very straight forward delay (850ms) with the ability to dial in some flanger sounds on the repeats. It's not a full-fledged flanger but it definitely gets you in the territory. Crtl1 sets the delay time (up to 850ms). Ctrl2 is the number of delay repeats. Ctrl3 fades in the flanger sound into the delay repeats as it is turned up.

Dreamtime: This is the same patch from the mbp Dreatime and is the only holdover from previous mbp FV-1 projects. This is another delay with feedback but the repeats are more filtered than the Whitehole. Ctrl1 is delay (up to 975ms) and Ctrl2 is feedback. Ctrl3 is where the magic is. The control pans between a multitude of fx. CCW is no effect. At a quarter turn it introduces a slow modulation. At half up, fast modulation with less depth. At three quarters, the modulation turns into a ring modulator. And in the last 25% that ring modulation turns into tremolo. Neat!

Build Notes

SpaceBomb: This patch is a reverb with an neat little “flange bomb” effect. Ctrl1 sets the reverb room size. Ctrl2 is a high pass filter. So, as you turn it up it reduces the low end of the reverb output (helpful when you want to keep things from getting too muddy). Ctrl3 is the “flange bomb”. This is not a true flanger, rather it uses an envelope that drives a small delay head before the reverb. The harder you pick, the more the delay shifts creating, well, a flange bomb. That’s the only way I can describe it!

These Halls: Of course, there was going to be a shimmer reverb. Everyone wants one even though lots of people think it’s overused. Shimmer in FV-1 is okay. Not the best sounding effect due to the lack of fidelity in octave up sounds. So, These Halls makes a good compromise in filtering the upper octave choir into more a of pad rather than distinct octaves. It actually sounds pretty good this way, IMO. Ctrl1 and 2 are the same as the Spacebomb (reverb amount and low cut) and Ctrl3 dials in the shimmer/choir effect.

These Ruins: This one is probably the least versatile of the 8 patches, but it’s a sound I like and can inspire some new ideas. It’s a reverb at its core, but with a pitch shifter. Unlike the shimmer effect, this one dials in a 4th below and 5th above (all the pitch effects are polyphonic, BTW). Ctrl1 sets the reverb amount. Ctrl2 sets the 4th down volume and Ctrl3 sets the 5th up volume.

- Some of these patches will take on a new dimension when you turn the BLEND signal all the way up to reveal the wet only signal, so keep that in mind when you are exploring all the sounds available to you.

Circuit Voltages

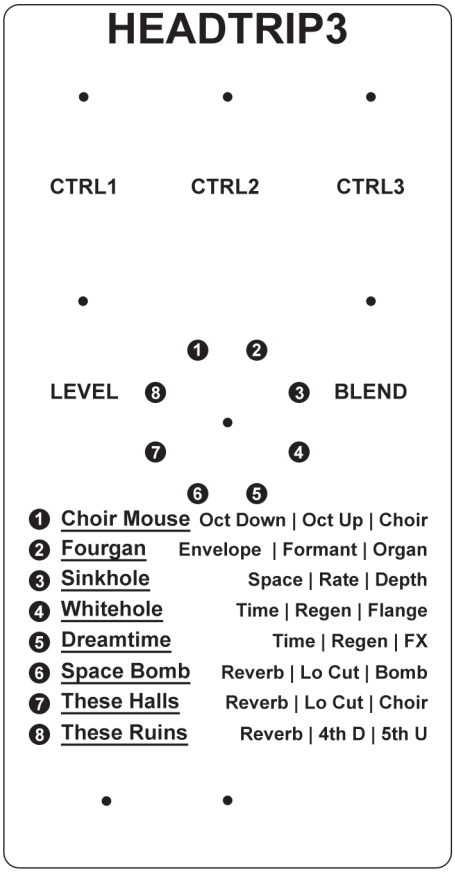
IC1	TL074
1	4.58
2	4.58
3	4.57
4	9.17
5	4.15
6	4.59
7	4.59
8	4.59
9	4.58
10	4.58
11	0
12	4.58
13	4.58
14	4.58

IC3	24LC32a
1	0
2	0
3	0
4	0
5	3.25
6	3.25
7	0
8	3.25

IC2	FV-1
1	1.62
2	1.62
3	1.62
4	0.6
5	3.25
6	3.25
7	0
8	3.25
9	1.51
10	1.34
11	0
12	0
13	3.25
14	3.25
15	3.25
16	3.25
17	3.25
18	0
19	0
20	1.64
21	2.34
22	2.23
23	3.25
24	0
25	0
26	3.22
27	1.66
28	1.66

- 9.5vDC One Spot
- Current Draw ~ 61mA
- Voltages will change on pins 16,17 and 18 depending on the rotary switch setting.

Simple Artwork



- Because the Headtrip3 has so much going on, I've made a simple label available to you that has descriptions of the patch controls. You can print directly from this page (do not use any page scaling). Trim your print just inside the outer border. smallbear has clear decal stock for ink and laser printers.

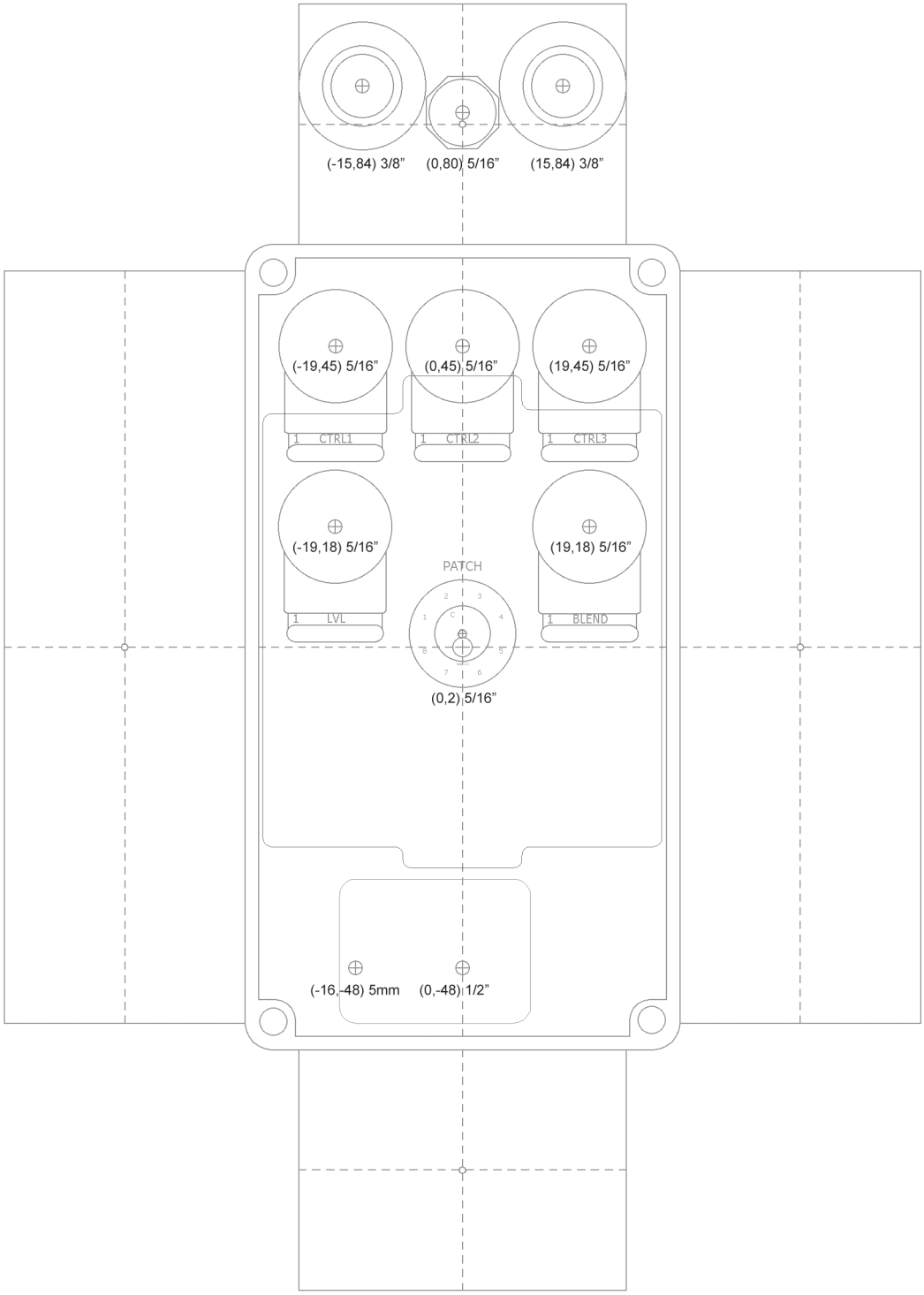
- I've included this artwork in the .PSD file located in the Headtrip3.zip. So, if you have a compatible gfx program you can use this artwork as a starting point to add your own graphics, choose your own fonts, etc.

- At least a few of the 1p8t rotaries I've bought in the past have full continuous turn (you can turn from setting 8 back to 1) but all the ones I've bought more recently stop at 8 so you have to turn it CCW back to 1.

Drill Template

Coordinates are denoted in (X,Y), drill size format starting from the center (0,0) location of the enclosure.

Tayda drill template:

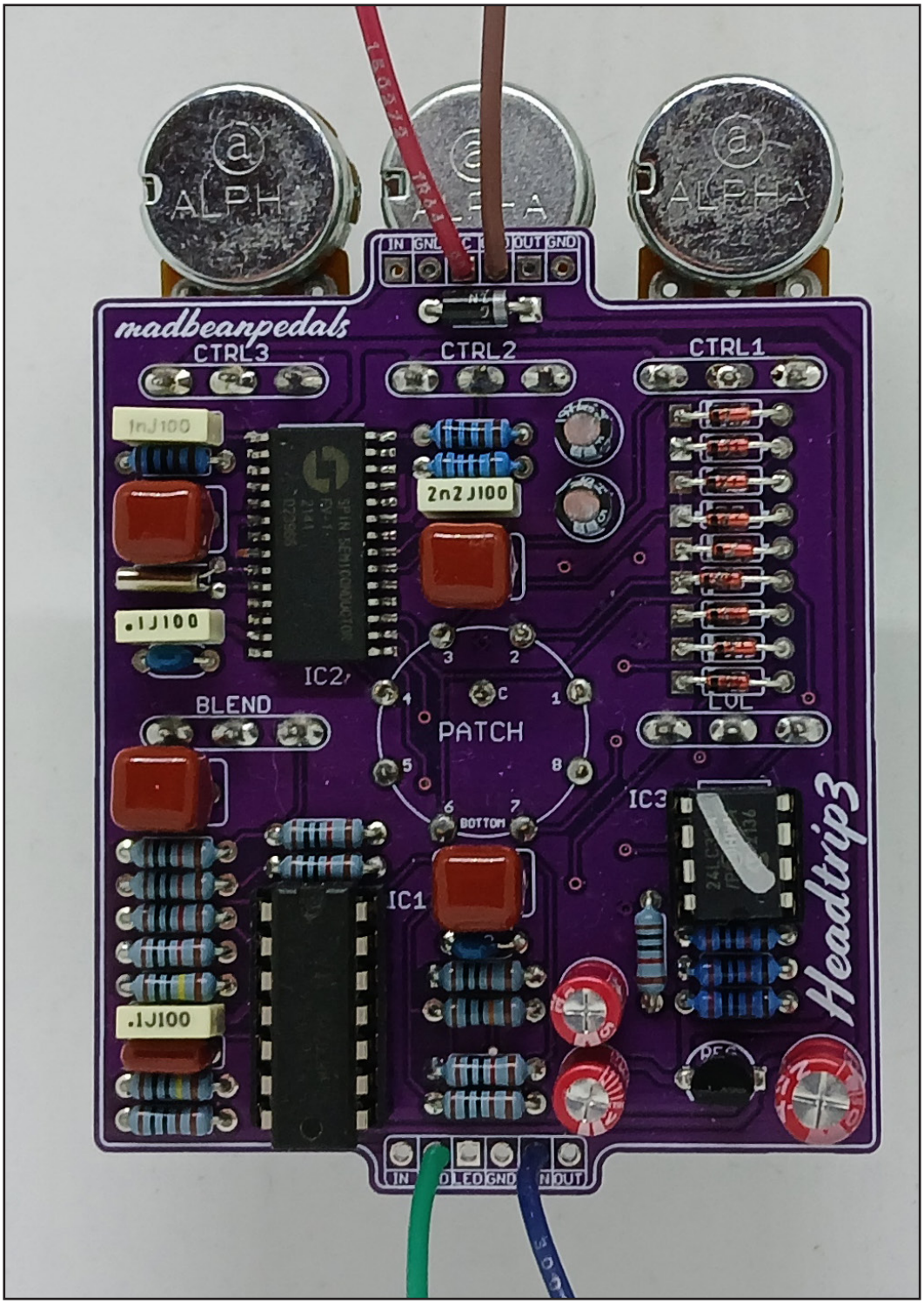


Hardware

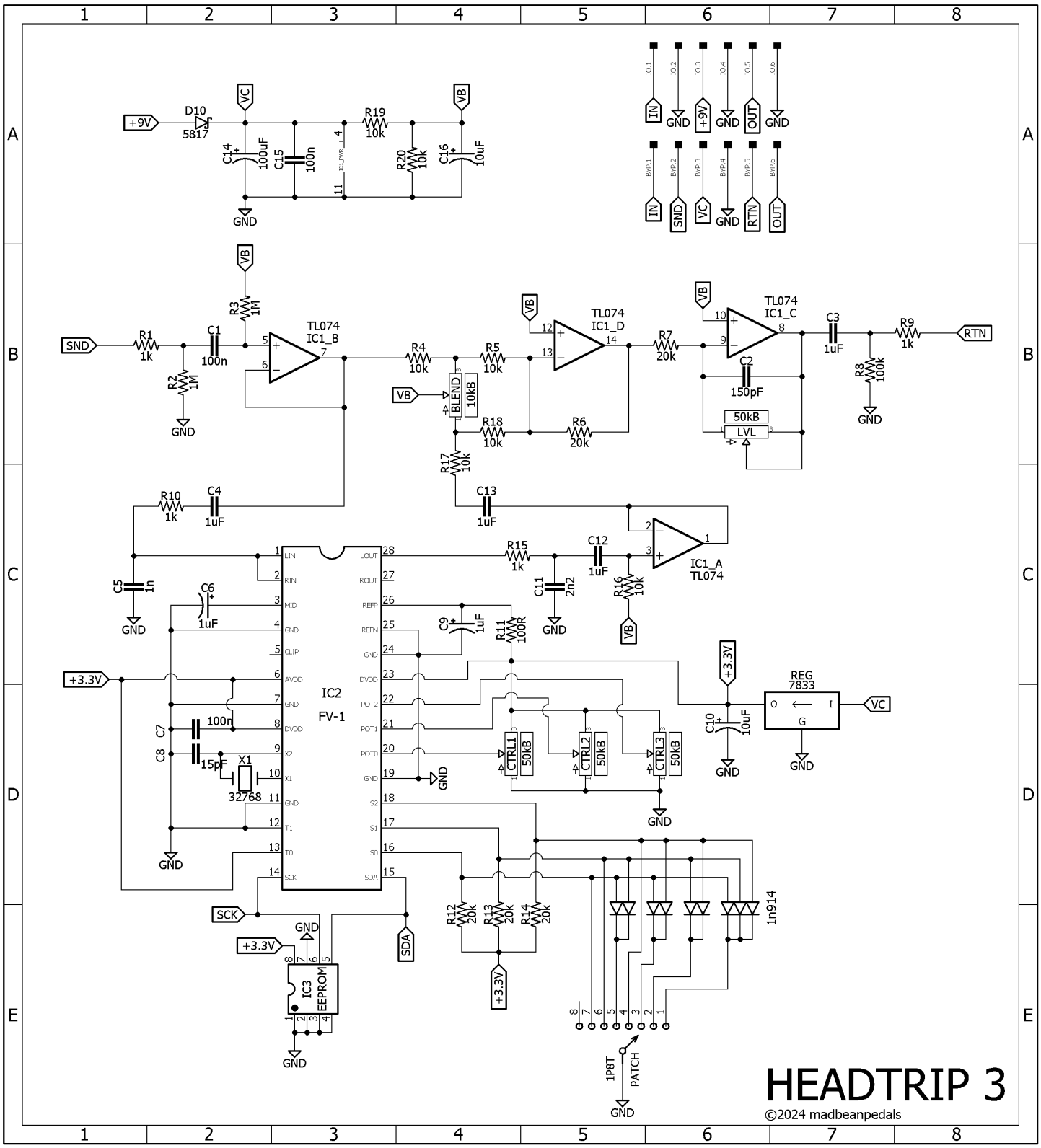
125B enclosure
Mono jacks
Slim 2.1mm DC jack
Standard 3PDT footswitch
5mm LED

NOTE: Different 1/4" and DC jack styles may require different sized drill holes.

Build Pic


madbeanpedals
CTRL3
CTRL2
CTRL1
1nJ100
2n2J100
.1J100
IC2
BLEND
PATCH
BOTTOM
IC1
IC3
.1J100
Headtrip3

Schematic


HEADTRIP 3
©2024 madbeanpedals
IC2
FV-1
TL074
IC1_A
TL074
IC1_B
TL074
IC1_C
TL074
IC1_D
REG
7833
IC3
EEPROM
1P8T
PATCH
BLEND
10kB
50kB
LVL
CTRL1
CTRL2
CTRL3
+9V
VC
VB
SND
RTN
SCK
SDA
GND
+3.3V
IN
OUT
SND
RTN
BYP
1n914

Bonus

Of course, you love the Headtrip2 patches I've included with this project but maybe you want to explore some other effects without having to build/buy a whole 'nother PCB, right? Well, you can do this with the Headtrip3! You can actually program your own EEPROMS with the patches of your choosing, burn them and install them right into the Headtrip3. Just replace the included pre-programmed chip with your own and you are ready to go.

Unfortunately, showing you how to do all the steps required is far beyond the scope of this document. But I can at least tell you what you need to get started. Perhaps I will be able to make a comprehensive tutorial on the whole process at some point.

Things you need

A Pickit2 programmer
These are no longer made but there are clones out there (usually from China on eBay). I cannot attest to their reliability or functionality so realize there may be a risk here should you purchase one. It may be possible to do the programming with a Pickit3 but I've never used one, so I cannot offer advice here.

Pickit2 v.2.61 software
This can still be downloaded from the Microchip website: https://www.microchip.com/DevelopmentTools/ProductDetails/pg164120
You may need the version that installs the .NET Framework A with it depending on your operating system and/or system configuration.

EEPROMs
You'll need one or more 24LC32A EEPROMs to program banks of patches.
https://www.mouser.com/ProductDetail/579-24LC32A-I-P

Free patches you can get right now without coding anything!
https://mstratman.github.io/fv1-programs/
https://github.com/HolyCityAudio/SpinCAD-Designer/tree/master/patches
http://www.spinsemi.com/programs.php

Optional

SPIN Assembler
http://www.spinsemi.com/Products/software/spn1001-dev/SpinSetup_1_1_31.exe
You can use this to program your own .spn files (these are the individual effects patches) and that's fine if you want to explore that. But, you can also use the Spin software to create the HEX files needed to program the EEPROMS via the Pickit2 programmer and software. Luckily, there are lots of .spn files available for free already!

SpinCAD development software
https://github.com/HolyCityAudio/SpinCAD-Designer
An alternative to coding your own patches is to use SpinCAD. This software is GUI based rather than code based and it allows you to construct algorithms using a node architecture using the JAVA platform (you need to install Java Runtime to use SpinCAD). What this means is you can make your effects by simply connecting blocks of programming together without needing to understand the actual coding required to make it all work. This is what I used to develop all my FV-1 patches so far. If you download the entire file from GITHUB you'll also find some SpinCAD files you can use as a guide to get started!

Support Forums
http://www.spinsemi.com/forum/index.php
https://www.madbeanpedals.com/forum/index.php?board=81.0
https://www.diystompboxes.com/smfforum/index.php?board=19.0

Bonus

For the inquisitive, here are some basics on how FV-1 development occurs.

Idea

You have an idea for an effect. This effect must be limited to the three control pots available with the FV-1 chip, but luckily there are many things you can do with three pots. For example, let's say you want to have a delay with time, regen and tone controls.

Algorithm

This idea must be translated into an algorithm that includes all the blocks necessary to execute the program. For our example, these would include input and output for the FV-1, a delay block, a feedback block and either a low pass, high pass or combination of the two into a tone control.

Code

There are two ways you can translate your algorithm into a working set of instructions that the FV-1 can execute. One is to use the assembler program created by SpinSemi specifically for the FV-1. This obviously involves coding and if you have experience with that it will probably be an easy transition to make. If you don't want to spend time learning to code, you can use the SpinCAD program created by DigitalLarry for the DIY community. This is a GUI-based program that lets you skip all the coding parts and simply connect pre-coded blocks together in many ways. For our delay example, you'll have a "delay" block that you can connect a pot to control the delay time. For feedback, you can use a "loop" block connected to a mixer whose volume is controlled by a pot to set the delay feedback level. There are several ways to utilize filters in SpinCAD controlled by a single pot, as well.

Assemble

Once your program is coded with either of the two methods above you need to get it into a form that the FV-1 can read. With the SpinASM you can save .spn files for programs you have created. These are collected into a single bank of 8 patches (you can use less than 8, of course). For SpinCAD, you can collect your different patch files in much the same way, which is then saved to a "bank" file. These files are then exported to .hex files which is what we use to program the EEPROMs - the device that stores the programs that the FV-1 reads and executes in real time.

Program

Whether you use the SpinASM or SpinCAD, once you have exported your bank to a .hex file the process is the same from there onward. This file is loaded into the Pickit2 software, then the Pickit2 hardware is connected to the EEPROM and the software burns the .hex file to the chip. The chip is installed in the circuit board and off you go!