PCB for PLL Code loader using RP2040 Zero

Mark Hughes GM4ISM

Based on firmware from G4EML and with design ideas from G4KMH and GM6BIG

This issue covers V1.0 and V1.1 boards

Colin G4EML has produced a suite of code on Github for the RP2040 to load various PLL evaluation boards it can be found here

Lehane G8KMH is in the process of extending this to include ADF2575 evaluation boards found on eBay etc.

I have designed a PCB to work with this firmware, to allow the RP2040 Zero to be plugged directly into the PLL evaluation board. My board has positions for several header sockets, supporting several PLL variants.

Serial data from a GPS engine such as the NEO6-M can be decoded to synchronise digital transmission modes that the code can generate. A header for NEO6-M GPS engines can be fitted.

The PCB

In its most simple form, the PCB only has to marshal the SPI lines from the RP2040 Zero to the PLL header. The board and GPS if used can be powered from the PLL 3.3V line subject to not exceeding the max current capacity of that output. That is not a known quantity but it is expected to be within the capabilities of most PLL board regulators. The other alternative is to power the PLL from the RP2040 3.3V line, however this is not recommended.

Firstly the RP2040 Zero on board regulator is only rated at 250mA continuous output and has to run the processor, leaving a max of about 200mA available, The PLL board may take over 150mA and a GPS antenna another 50mA. This puts the regulator on its limit and assumes an ideal thermal design which is not necessarily the case.

Secondly, this regulator is not a low noise device and the 3.3V line may have additional noise due to the RP2040. It is not really suited for the PLL 3.3V supply.

As supplied, the RP Zero 3.3V is isolated from most other parts of the cct although it can be linked with a solder bridge to the GPS interface. It can also be linked to the PLL 3.3V line as below.

By fitting a schottky diode D1 between the PLL 3.3V rail and the RP2040 3.3V rail, the board and RP2040 can be powered by the PLL 3.3V in some cases. (Not all target PLLs present 3.3V) The diode means the PLL can't be powered from the RP Zero. It allows for the RP2040 to be powered by its USB port without having the PLL 3.3V and RP2040 Zero 3.3Vs 'clash'

There are several points when optional decoupling caps (0805 size) can be fitted should they be needed.

Minimum suggested Component load

Diode D1 Capacitors C2 and C6

GPS

A serial port (Software serial, RX GPIO0 and TX GPIO1) is provided for connection to the GPS engine. This is expected to be a 3.3V device, however the PCB can be configured to supply either 5V or 3.3V to the GPS with JP2

If you wish to use a 5V GPS engine with a 3.3V RP2040, the RX data and PPS lines needs to be level shifted. R1, R2 *Serial data), R3 and R4 (PPS) need to be implemented on the board. Suggested values are 3K3. The PPS can be routed to GPIO9 via R3 / R4 but is not presently used in the firmware.

R1 and 3 are bridged out with solder (0 Ohms) and R2 and 4 are omitted for 3.3V logic levels. Operation with a 5V GPS has not been tested.

GPS supply

For most users, the J2 (rear of board) will be jumpered to 3.3V and that will be jumpered to the main RP2040 Zero 3.3V line (front of PCB to the right of D1 position)

If your GPS engine expects more that 50-60 mA it may be preferable to use a separate 3.3V regulator from the 5V line. The board can accommodate a SOT23-5 fixed regulator and mandatory decoupling caps (at least 1uF). In most cases (eg NEO6-M) this will not be needed.

Pin-outs have been arranged so 2 versions of the popular NEO6-M GPS boards can be directly plugged in on top. Check the connections are suitable for your GPS, they aren't all the same!

Note on this board the serial RX and TX pin designations are for the board ie TX outputs data, RX is data in (from TX on your GPS engine)

EEPROM Option

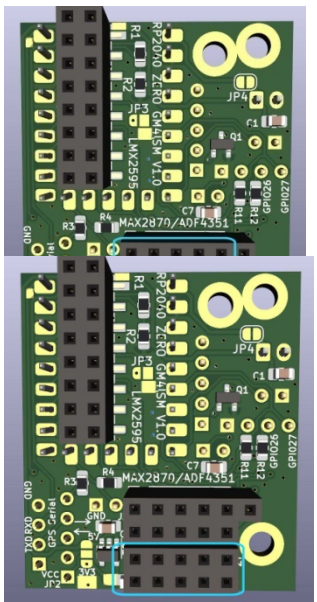
The board can accommodate a SOIC-8 EEPROM (SPI) eg a 25LC640A

The potential need for this is because the emulated EEPROM on the RP2040 is only 4K and limits the number of preset channels to 10 (16 are addressable in the standard hardware)

Emulated EEPROM also suffers from a much reduced number of write cycles compared to a dedicated EEPROM so if you are constantly changing the memories, there is a potential to wear out the RP2040 flash memory. G4EML's firmware does not presently support external EEPROM and as such I can't test the functionality of the EEPROM circuit.

Most users won't need the EEPROM but it is an option. This *may* be required for Lehane, G8KMH's forthcoming firmware for the LMX2575 which this board can support.

PLL Board Headers



Socket used for Max2870 (#1) and Black ADF4351 PLLS (#2)

Also compatible with some ADF5355 PLL boards (#3) although not presently supported by the firmware. No need to place LD components (R5, R6, Q1 etc) Use JP1 to ground Pin 9 of the PLL header.

Socket used for SV1AFN ADF4351 PLL (#5)

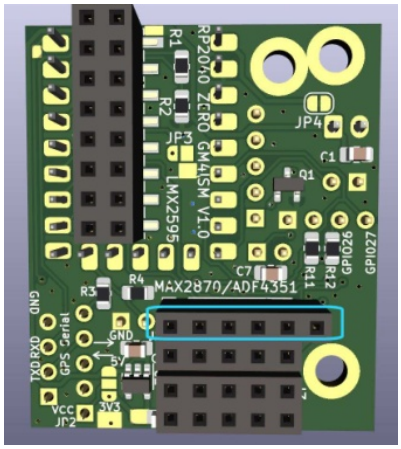
Note that the SV1AFN board does not have an onboard 3.3V regulator.

The 3.3V regulator on this board can be used to supply the PLL board (about 100mA). For this purpose an Ultra low noise regulator is preferred.

Not tried yet but the TPS7A2033 or LDLN025M33R look suitable. These are not as good as the expensive AD regulators (that don't fit) but are pretty reasonable with about 7μV RMS noise.

Whilst it may be possible to put all the headers on the PCB at the same time generally only one is required. All are independent EXCEPT for the ADF4351 SIL Green board #4 and #7 . If you want to use the board with these PLL boards you need to make minor modifications which would stop it being used for some of the other style PLLS

There are tracks to cut and links to be made to be able to use these PLLs. The links and track remain accessible with headers fitted so it is possible to fit appropriate headers for both the SIL header PLL board and the other but the wiring changes make them mutually exclusive without changing the links etc. There was insufficient room on the PCB to fit handbag links for the options.



V1.0

A surface mount 2x8 header (J5) is required for the LMX2575 (#6). This fits directly below the RP2040 Zero. At the time of writing, Lehane G8KMH is testing his revised firmware to support these PLLs before publishing it. The mute pin (13) on the LMX2575 connector can be connected to GPIO2 with Jumper J3. This may be required for Lehan's firmware. I do not expect to test this combination as I don't have the PLL eval board

Other headers

As per Colin G4EML's firmware:-

Header J9 is presented for memory selection, labelled Control.

Header J1 is presented for CW keying input

Headers J4 and J7 are for the GPS receiver. 2 commonly available types are supported, one employing 5 pins with 1PPS the other has a 4 pin header. The pin layout is not the same on all GPS modules.. **Check to see if the headers are directly compatible with your GPS engine**

1PPS (if available) is presented at J8. If a 5V GPS engine is used, R3 and R4 will reduce the PPS voltage for 3.3V interfaces including the RP2040 Zero where PPS is fed to GPIO9

To use 1PPS from a 3.3V GPS engines means not fitting R4 and placing a link (solder bridge) in place of R3 (it has close spaced enlarged pads to enable this).

Header J10 presents I^2C for possible expansion. Pullup resistors minimum 10K should be fitted somewhere on each Data and Clock line, which are on GPIO26 and GPIO27 of the RP2040 respectively. Some I^2C peripherals have pull-ups fitted.

Header J11 presents 3.3V and an open collector transistor Q1 through a Resistor R6 for an optional Lock indicator LED for those boards that present LD (lock detect) on the header (Black ADF4351)

Driving the LED directly from LD or the MUX pin may be an issue though many eval boards do it. As far as I can see from datasheets, these pins are supposed to be limited to 500uA! Q1 should have a high h_{FE} meaning it needs only uAs into the base to conduct hard enough to fully drive a LED, R5 should be about 33K and h_{FE} 300 or more, so about 20mA Led current is attainable (limit this with R6 as required)

A SOT23 BC847C is a good choice for Q1, h_{FE} is over 500. I have also used an MBT2222 without problems.

Getting going

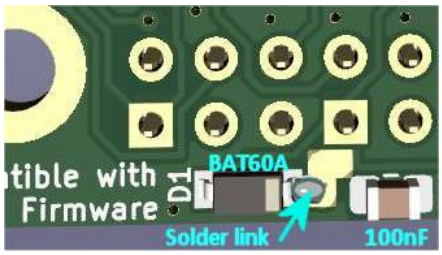
It is recommended that the boards are dedicated to one particular target PLL board. Some target PLLs require a couple of tracks to be cut and links to be made. This is best done before fitting other components such as the header sockets. Any optional components like the lock detect LED driver cct should be fitted before headers too. It makes life easier!

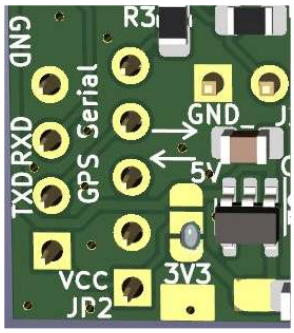
For 'Standard' black ADF4351 and LMX2870 boards with NE06-M GPS

Either Fit D1 (carefully noting the polarity) or place a solder link across it's pads (which have been extended to make this easy). Fitting the Diode is highly recommended for most circumstances.

I would suggest also fitting 100nF capacitors C2 and C6 , decoupling on the 3.3V rail.

For a 3V GPS, link across the pads of R1 (serial data to RP2040). Solder a link to the pad next to D1 cathode and solder the link on JP2 for 3.3V (1-2) to provide it power. Should PPS from the GPS engine be needed in future firmware releases, 0805 size R3 (10K) and R4(12K) are provisioned on the back of the PCB to allow for 5V GPS engines. for 3.3V GPS, R3 can be linked out with a solder bridge. At this time nothing needs to be fitted.





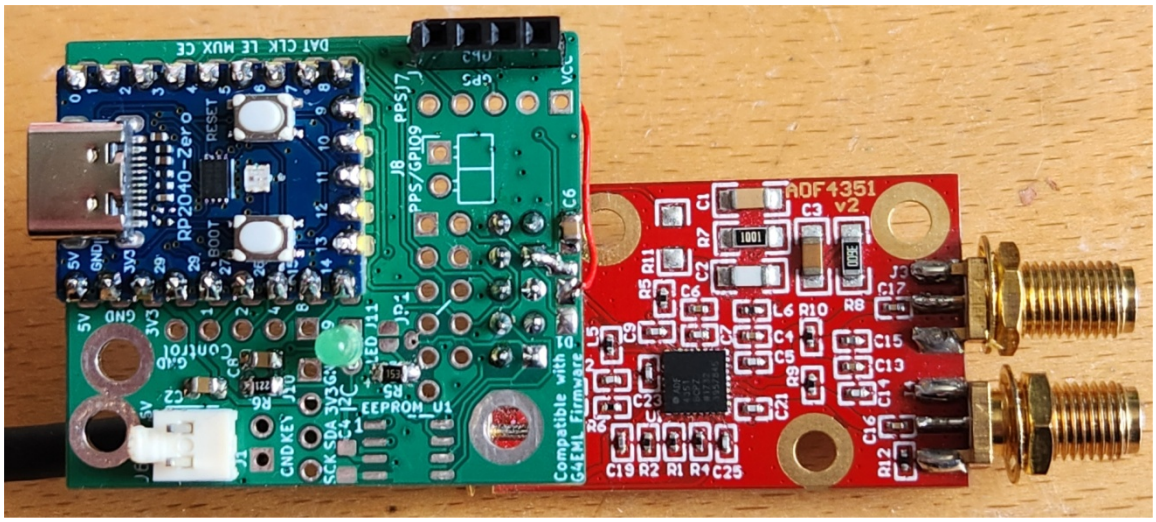
Fit a PLL board header for the PLL of choice (noting that the SIL version of the ADF4351 board #4 and board #7 require minor changes to the board.)

Fit GPS, control and Key headers as needed.

Fit and programme your RP2040 Zero. I recommend that if D1 is not fitted, that the PLL is not attached while using the RP2040 USB connection. The RP2040 can also be programmed in advance. When first connected the RP2040 presents as an external drive. Copy the uf2 file to it and that's it! It reboots and can be found as a serial device (9600 Baud 8N1) It talks nicely to Putty. Current firmware from G4EML's Github repository is RP2040Synth_V1-05.uf2

Using the board with the SV1AFN PLL

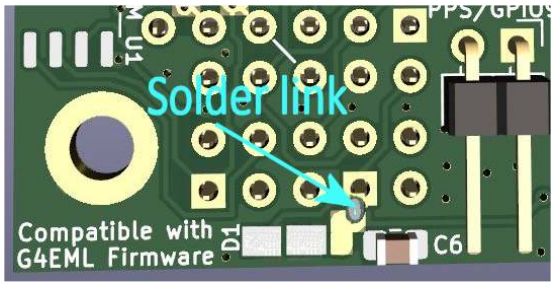
<u>Important</u> :- When I finally got hold of an SV1AFN PLL I discovered an error in the layout of my V1.x RP2040 zero boards. The interface socket is rotated 180 degrees from the intended orientation.



The hat works fine but the PCB does not sit over the PLL board as intended, it is off to the side.

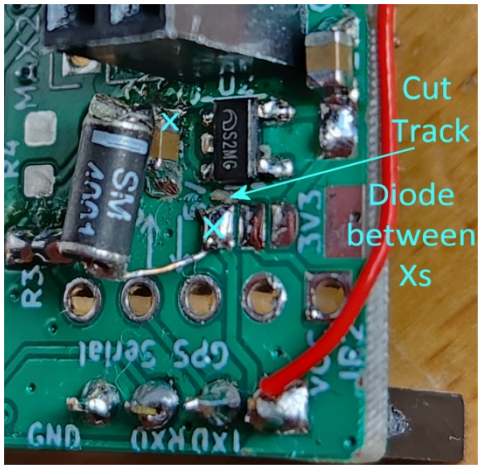
I haven't figured how this error crept in. Since the connector J3 is only used for this type of PLL no other board types are affected.

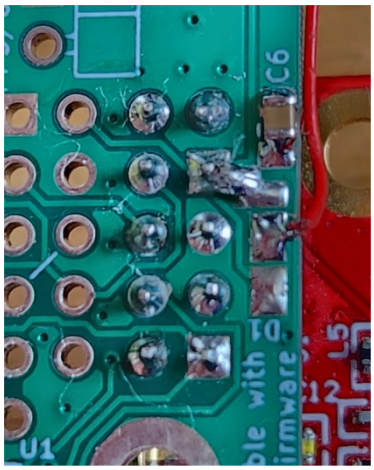
The 3.3V LDO regulator, preferably an ultra low noise variant in SOT23-5, can be used for this setup Eg TPS7A2033 or LDLN025M33R**.** (I used a ME6211C33 from my stock, it is not ultra low noise). Use decoupling caps in and out (0805 C3 and C5) as dictated by the datasheet, usually at least 1µF 0805 Bridge the regulator output to the Header J3 pin7 using the expose pads. Do not fit the diode D1.



If a 3.3V GPS is required while using the SV1AFN PLL, it ***could*** be supplied by the same 3.3V regulator as runs the PLL board. However if the 3.3V is used for other things as well as the SV1AFN PLL, you can potentially load the regulator excessively ***unless*** the 5V input voltage dropper to the 3.3V reg mod is applied. The GPS can alternatively be powered from the 3.3V line out of the RP2040, requiring a small wire jumper, see below. The V1.1 board has tracks and pads that make this easier to implement.

Pics below show the Diode voltage dropper mod and a link (red wire) to power the GPS from the RP2040 zero's 3.3V, not the hat's onboard regulator (done on the V1.1 PCB pads, see V1.1 section). Neither mod is needed if you are just running the SV1AFN board without a 3.3V GPS attached.

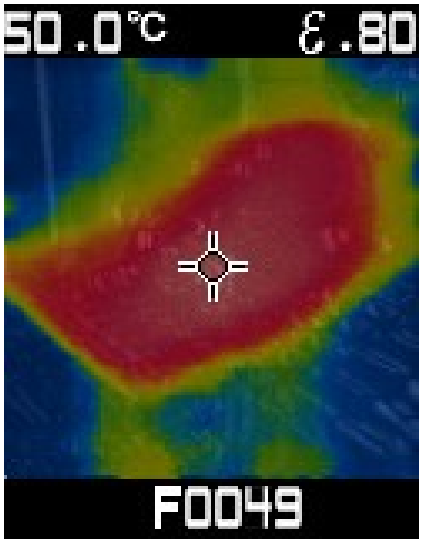




The voltage dropper mod on a V1.0 board cuts the track feeding the 3.3V regulator and non Schottky diode is piggybacked as above. The diode must be rated at at least 250mA so a 1N4148 won't do

ON the V1.1 board either fit D2 or a dropper resistor (suggest 3.9Ω 1/4W 1206 size) to take some of the dissipation from the regulator. The 3.3V regulators are all LDO and work with as little as 3.6V in. Loosing up to 1V in a suitably rated resistor or diode keeps the regulator cooler

FLIR camera picture below shows the 3.3V regulator and dropper diode running at 220mA for over 1 hour.

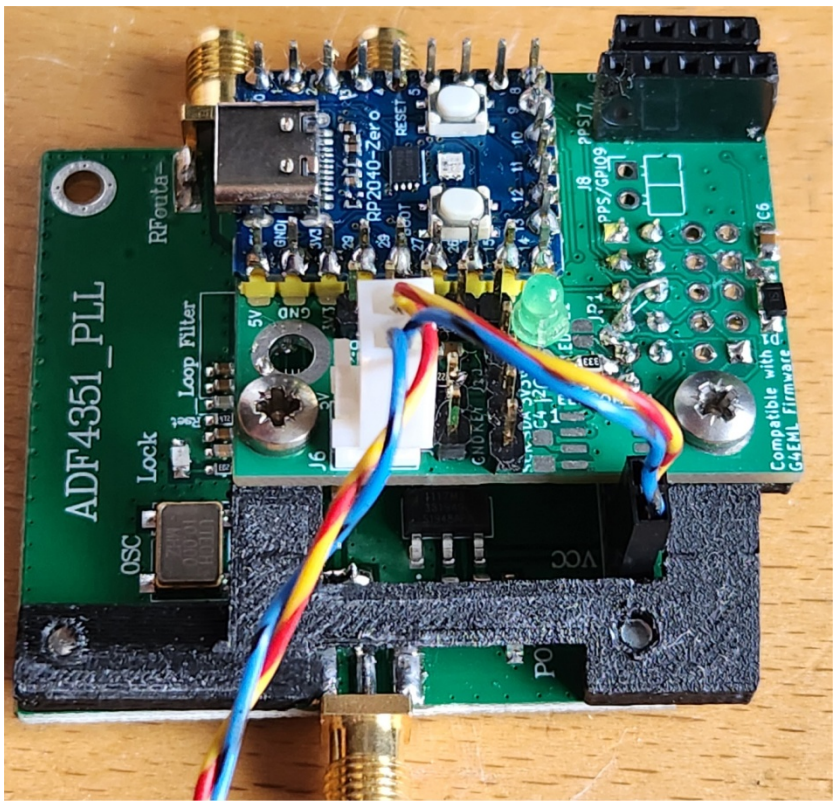


Using the V1.0 board with the ADF4351 PLL #4

<u>Important</u> :- minor changes to the board are required to use this type of PLL board

The mods are reversible with solder links if the board is to be used with different target PLLs later.

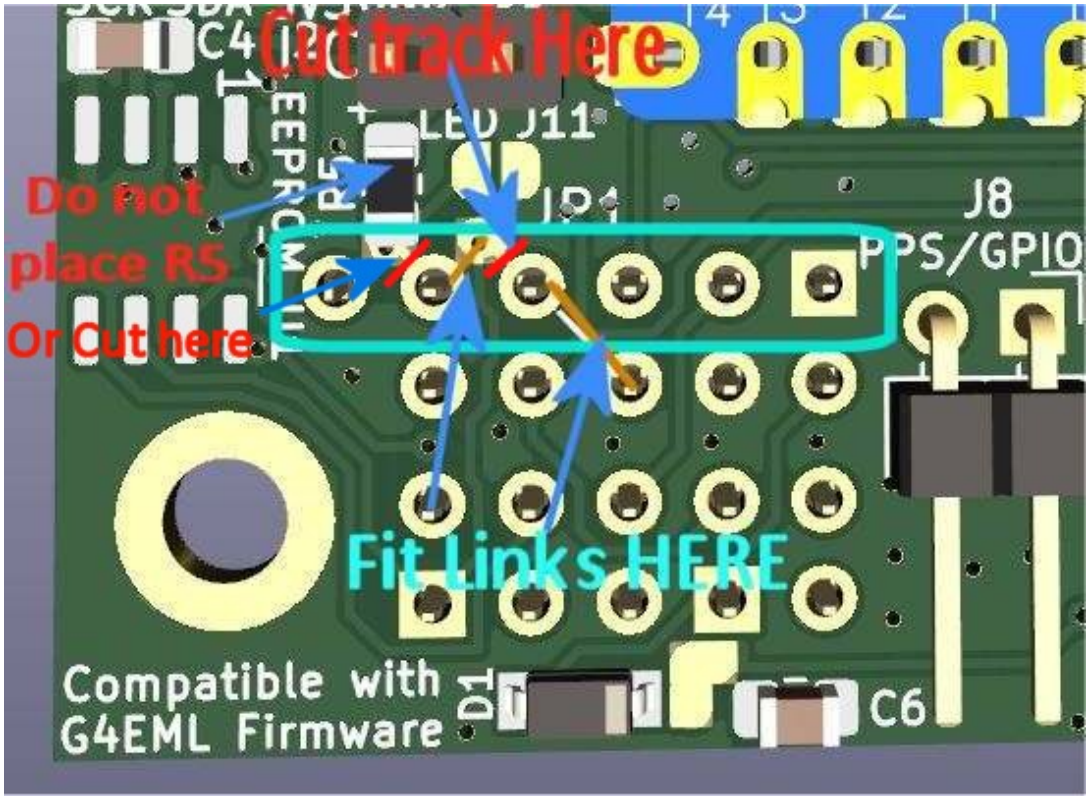
The PCB hat does not align with any mounting holes on this board type. Being a Single In Line header makes this mechanically a little unstable. A 3D printed bracket to support the hat has been designed and can be 3D printed from the stl file on the website. I will consider 3D printing them for users that want them.



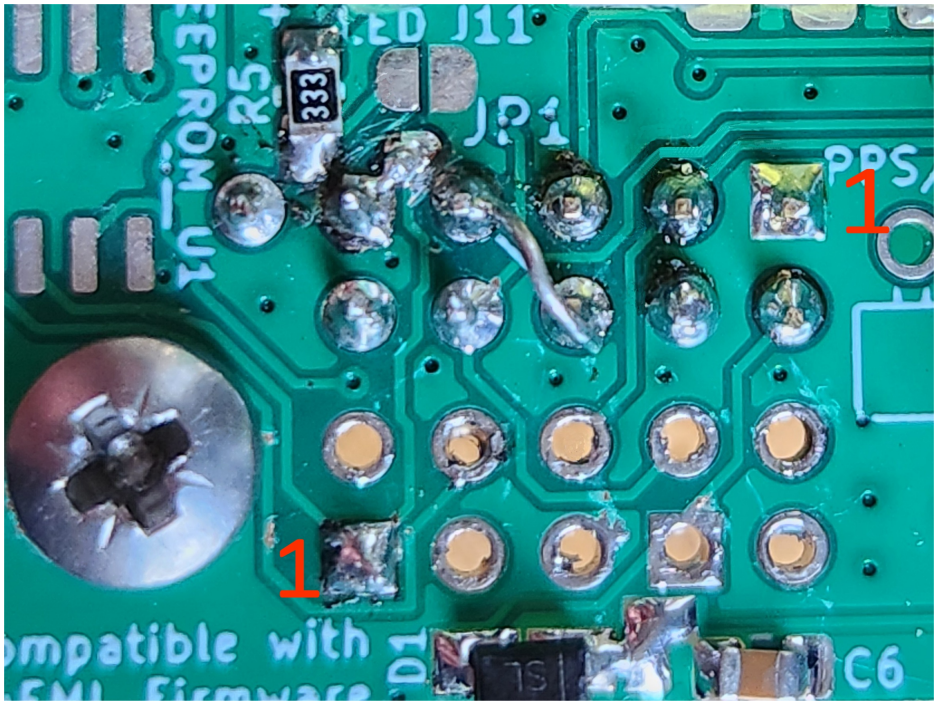
Both the PLL and hat need 5V The PLL can be powered independently or as above, the 5V hat connection can be linked to the appropriate pins on the PLL. The USB 5V on the RP2040 zero connects directly to this 5V line. There is no 3.3V pin on the header. Likewise, the Lock Detect from the ADF4351 is not presented on the header. The Diode D1 is shown fitted (from previous tests) but is not needed for this board.

Note that this board also has a 10MHz Ref oscillator not 25MHz as others. It can be linked to use an external reference. Check the board, as mine had neither Ref input selected (Solder link)

I use mine with an external 10MHz and have taken the power off the internal oscillator to prevent spurii that are possible die to leakage from the powered ref osc which is a not exactly 10MHz



Seen on the actual modified board. Note that the link to J2 pad 7 needs to be cut with care so as not to cut the adjacent track that runs between pins 7 and 9. I strongly suggest cutting tracks before fitting the headers!



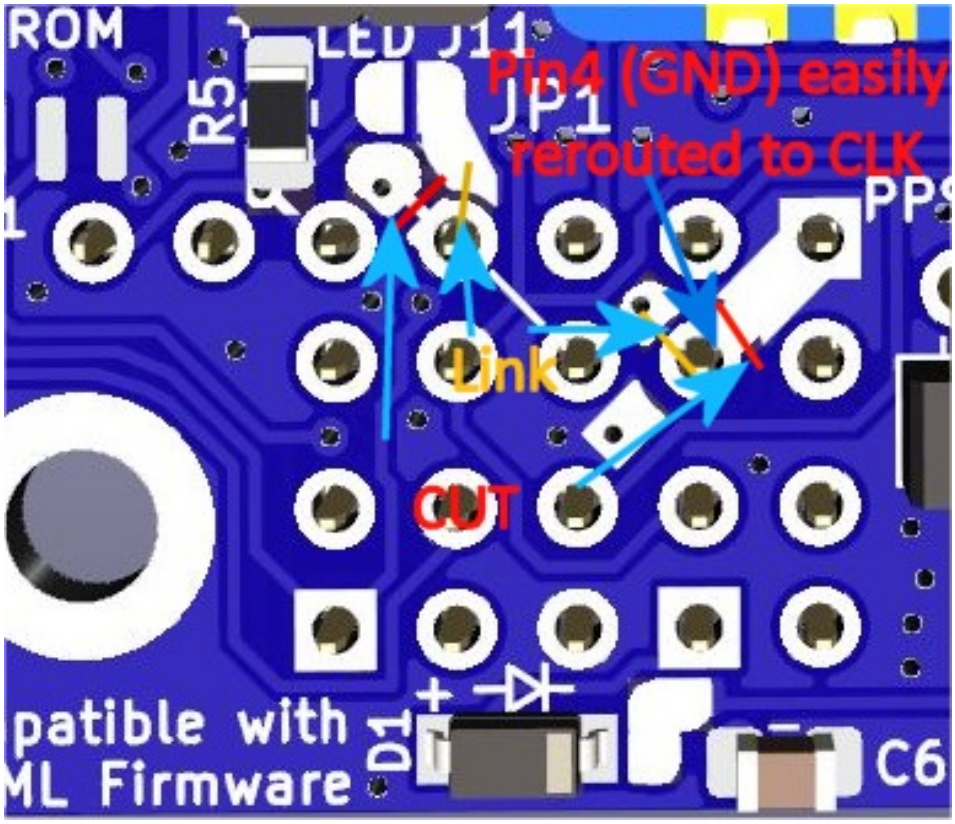
V1.1 with PLL board #4

The Blue V1.1 board has an extra header adjacent to the MUX pin for the SIL ADF4351 board #4 This enables the Lock Detect input to the hat to be allocated to an unused pin on the PLL board. The PLL board can then be modified to link that pin to the PLL Lock detect pin. Mods are as for V1.0

V1.1 differences and capabilities

As well as the extra header pin for type4 PLLs, V1.1 also supports yet another recently discovered ADF4351 board that derives its power from a USB C connector. This is shown as #7 in the section identifying supported PLLS

Pin 4 of the header J2 must be disconnected from Ground and connected to CLK. A cutable track and a convenient adjacent pad on the CLK line are provided for this. Pin 7, normally CLK, needs to be disconnected from that (cutable track) and linked to an exposed adjacent grounplane 'pad'

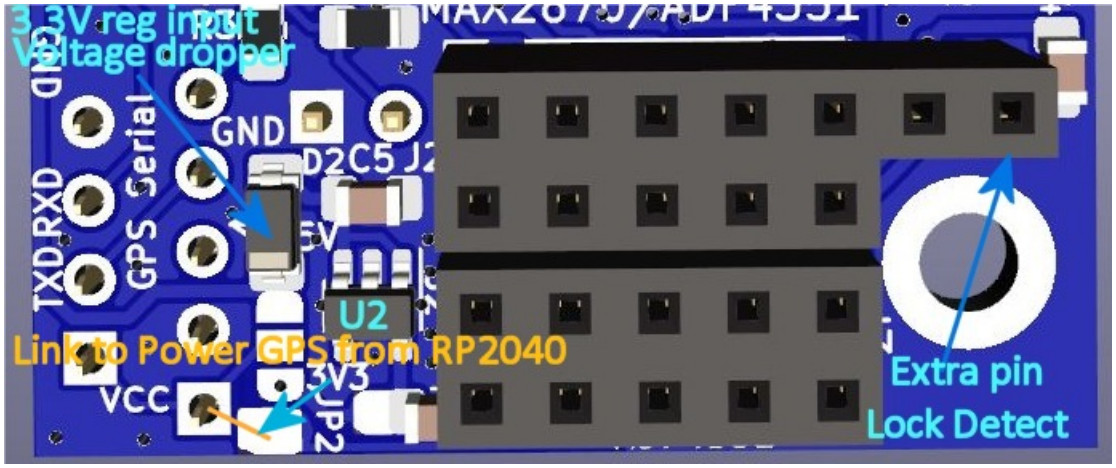


To use this PLL with V1.0 board would require J2 pin 4 plated hole to be drilled out so that header cannot connect to Ground. A wire could then be attached to the header pin in isolation and routed to the CLK line (Pin1 of J3). Pin 7 can be treated as described above, the cutable track is already available and exposed Ground is nearby on JP1

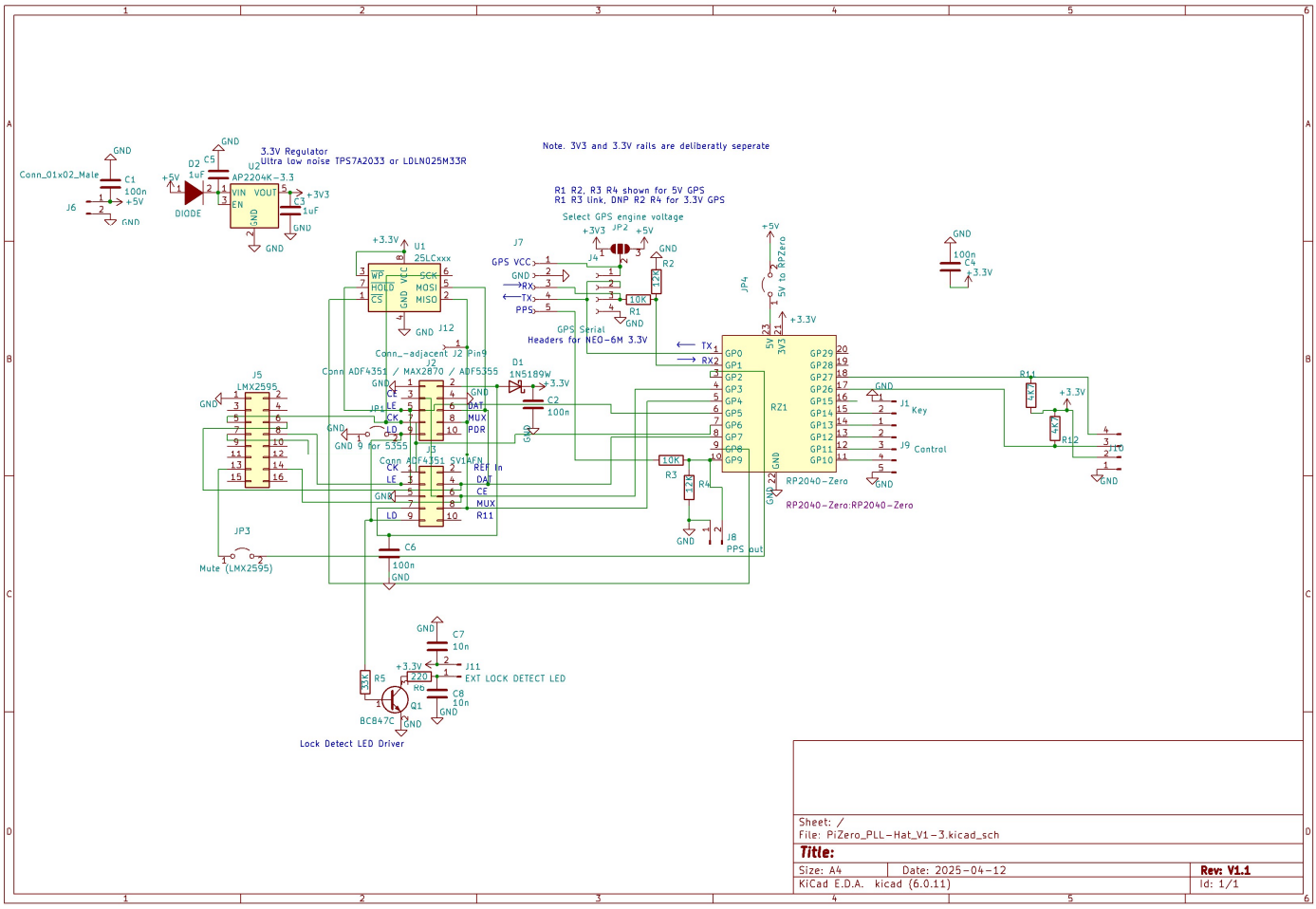
V1.1 also implements pads for a Diode D2 or Dropper resistor) to reduce regulator chip dissipation.

This is only needed if the regulator is heavily loaded. The 100mA or less of the SV1AFN PLL does not require the regulator to have the voltage dropper fitted and in that case, D2 pads are bridged with a solder link

On the rear of the board is a pad adjacent to the GPS headers. This is connected to the RP2040 zero 3.3V rail and fitting a solder jumper allows the RP2040 to power a 3.3V low current GPS like the NEO6-M



V1.1 Schematic

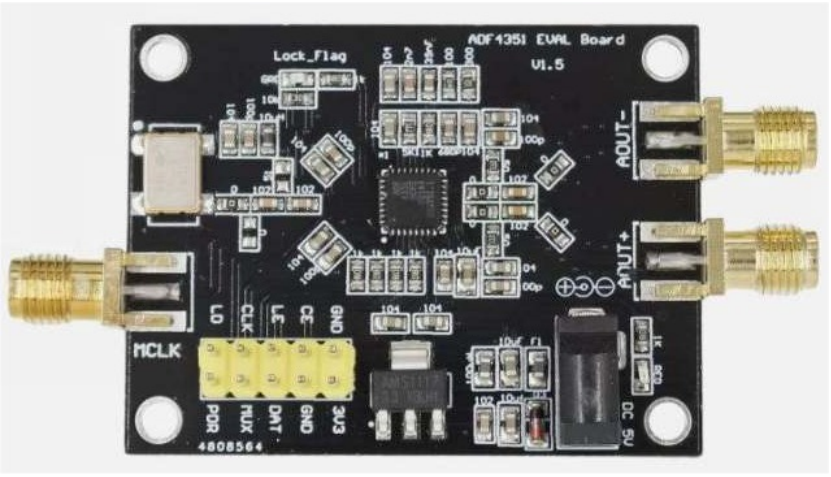


Identifying supported PLL boards

It is essential to check pinouts of your PLL boards match those presented on this PCB Not all of these boards have been tested as I don't have them!

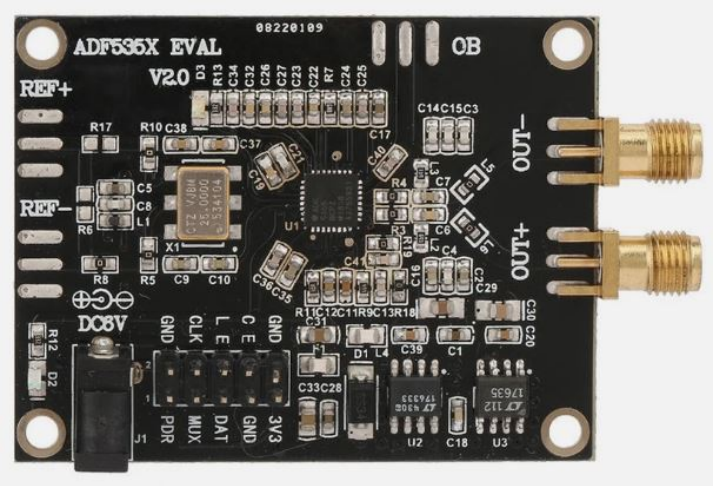
#1

……



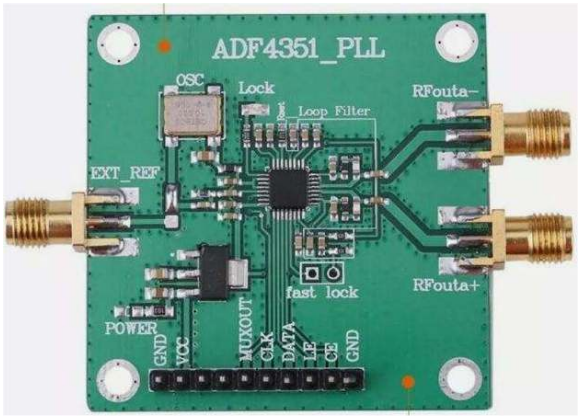
#2

……



#3 (No Firmware support at this time)

……



#4

Tested OK Requires tracks to be cut and links made

……

SV1AFNs board connections

#5

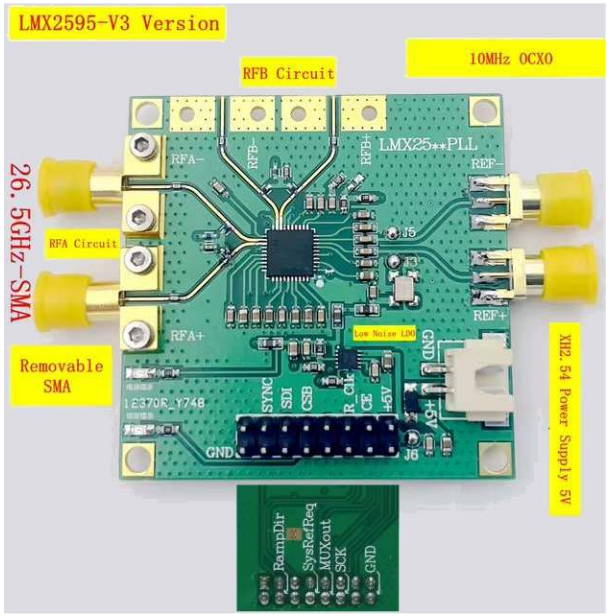
Details here

https://www.sv1afn.com/en/products/adf4351-pll-synthesizer-module.html

Please note. The orientation of the PLL socket on V1.x boards is rotated 180 degrees from what was intended. The RP2040Zero hat works fine but ends up off to one side of the PLL board, not over it.

Only discovered when I finally got my hands on the PLL board.

……..



#6

……

ADF4350 137Mhz-4.4GHz/ADF4351:35Mhz-4.4Ghz

#7

Can be used with V1.0 board but the mods are more involved.