PCB for PLL Code loader using RP2040 Zero

Mark Hughes GM4ISM

Based on firmware from G4EML and with input from G4KMH and GM6BIG

Colin G4EML has produced a suite of code on Github for the RP2040 to load various PLL evaluation boards

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

I 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 time Digital transmission modes that the code can generate. A header for the GPS engine is included.

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 a thermal design which is not necessarily optimum.

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 suitable 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

By fitting a schottky diode D1 between the PLL rail 3.3V supplies and the RP2040 3.3V rail the board can be powered by the PLL but 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 jumpered (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 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

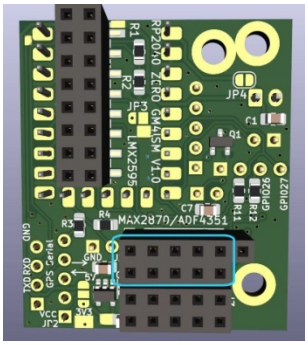
The board can accommodate a SOIC-8 EEPROM (SPI) eg a 25LC640A

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

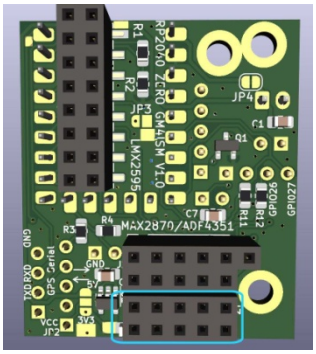
Most users won't need the EEPROM but it is an option. This *may* be required for Lehane, G8KMH's forthcoming firmware for the LMX2575 which this board can support.

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 (#4)

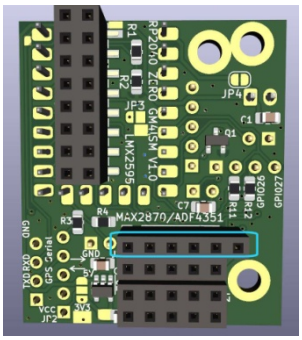
Note that the SV1AFN board does not have an onboard 3.3V regulator

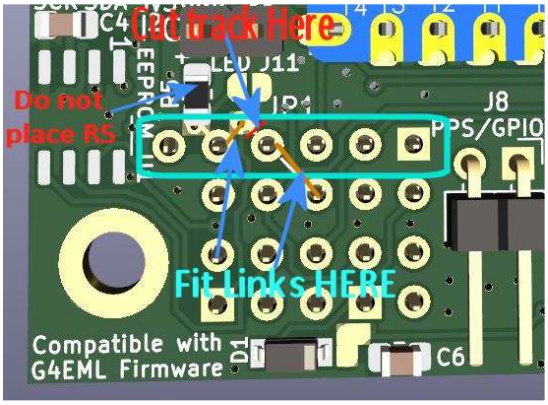
The 3.3V regulator on this board can be used to supply the PLL board (about 100mA) For this purpose an Ultra low noise regulator is needed

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

Whilst it may be able 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 (#5). If you want to use the board with this style of PLL board you need to make minor modifications which would stop it being used for some of the other style PLLS

There is 1 track to cut and 2 links to be made (a 6 pin SIL socket is required) to be able to use this configuration. 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.





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.

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

Use of 1PPS with 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 at least 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

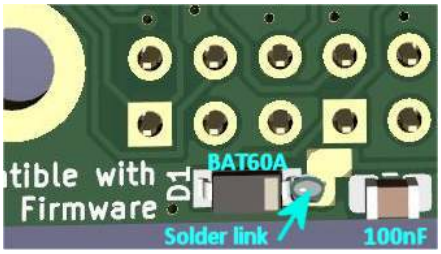
Getting going

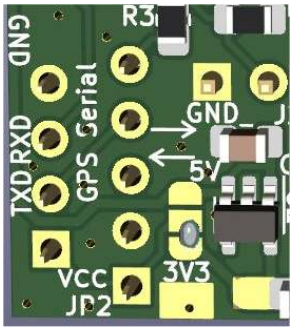
For ‘Standard’ black ADF4351 and LMX2870 boards with NE06-M GPS

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 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





Fit at least one PLL board header for the PLL of choice (noting that the SIL version of the ADF4351 board requires minor changes to the board.)

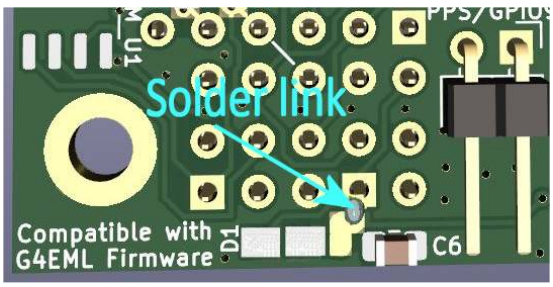
Fit GPS, control and Key headers as needed

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

Using the board with the SV1AFN PLL

The 3.3V LDO regulator, preferably an ultra low noise variant in SOT23-5 can be used for this setup

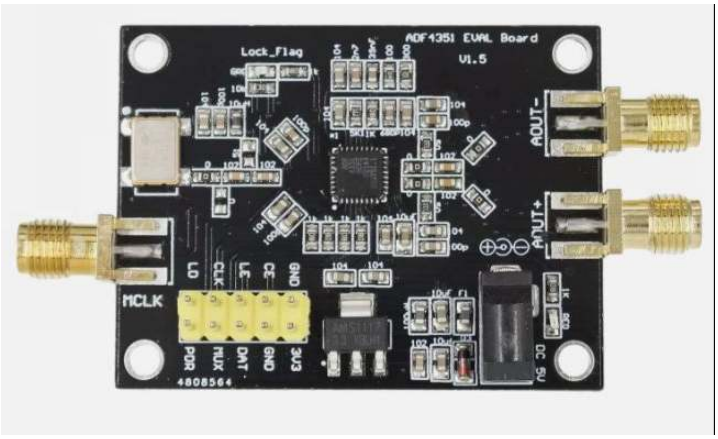
Eg TPS7A2033 or LDLN025M33R**.** Use decoupling caps in and out (0805 C3 and C5) as dictated by the datasheet, usually at least 1µF 0805 Bridge the regulator output to the Header J3 pin7 using the expose pads. Do not fit the diode D1.



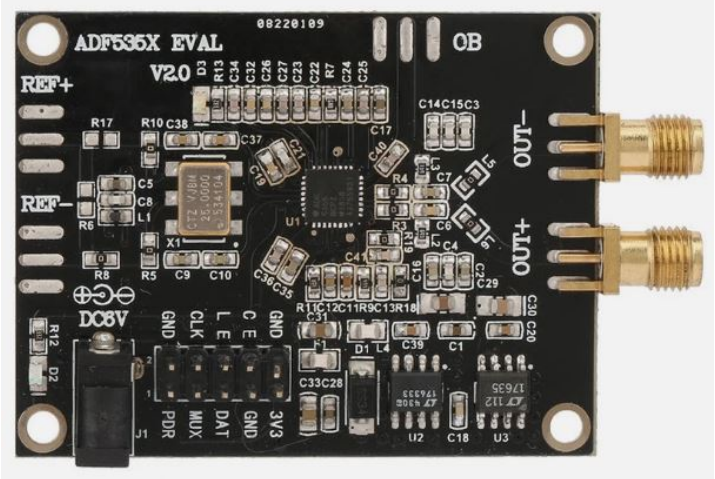
Examples of supported PLL boards

It is Essential to check pinouts of your PLL boards match those presented on this PCB

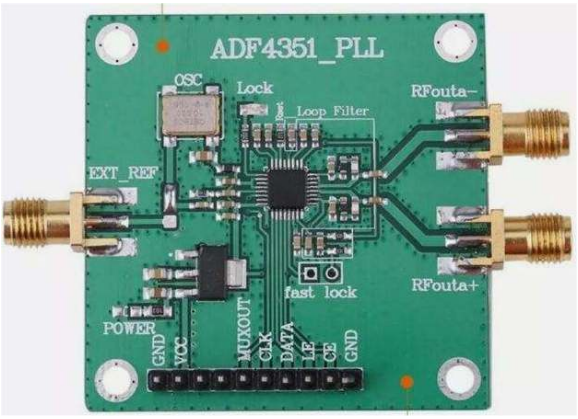
#1



#2



#3 (No Firmware support at this time)



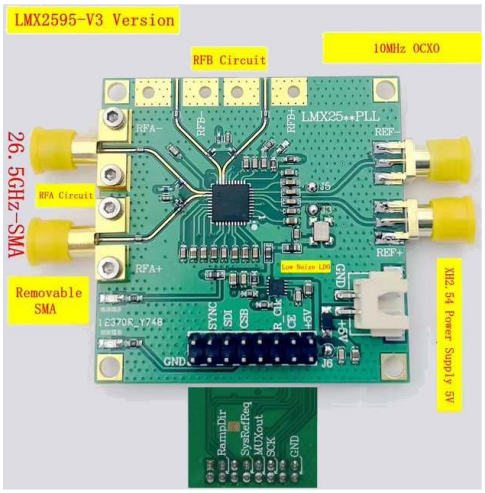
#4

SV1AFNs board connections

#5

Details here

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



#6