ProgRock2: Triple GPS-disciplined programmable clock

Contents

1. Introduction

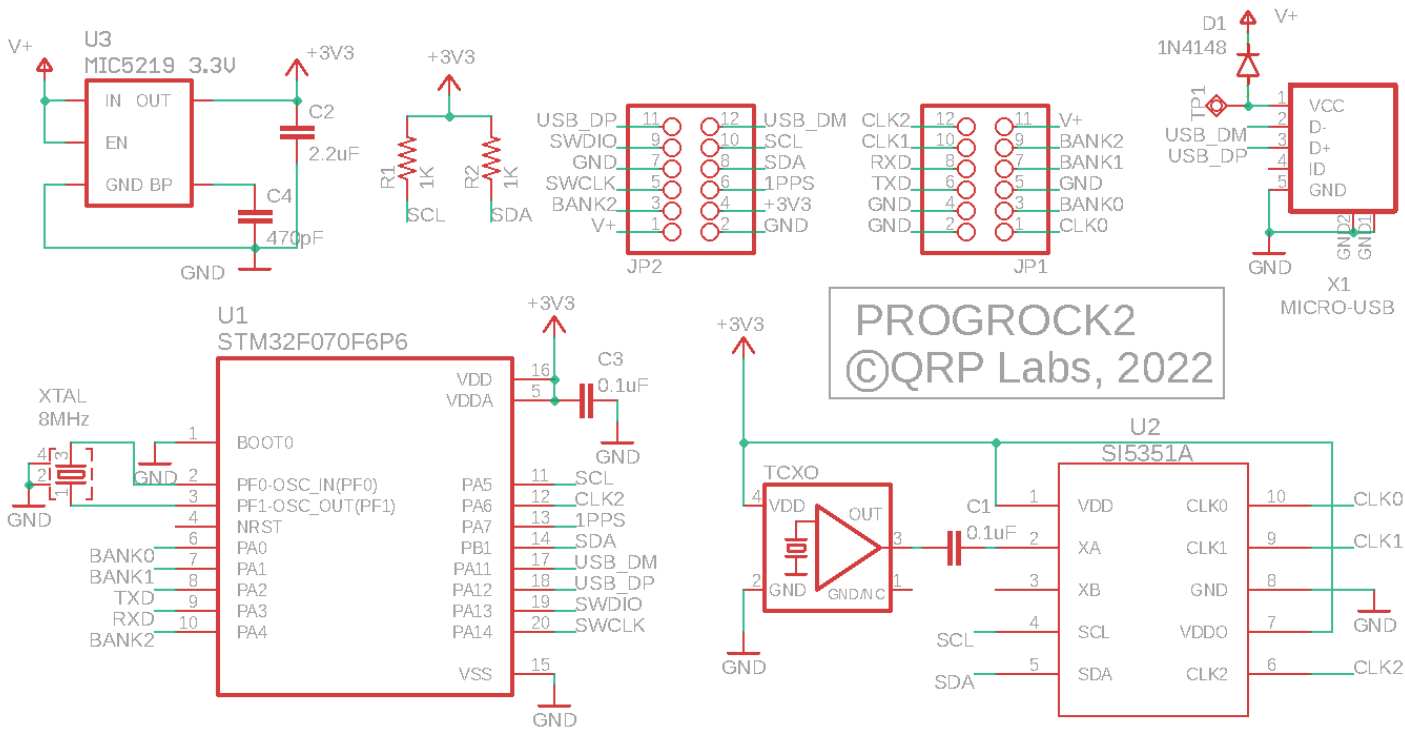
ProgRock2 is the successor the popular ProgRock kit. It features improved performance, smaller size, and is ready-assembled using all-SMD components. Configuration is via a micro-USB port which can also be used to update the firmware.

- Tiny size PCB, a little smaller than an HC6 crystal: 0.725 x 0.675 inches (18.4 x 17.1mm)
- Factory assembled, ready-to-use (no assembly required)
- 3 independent 3.3V p-p squarewave outputs (2 if you use GPS discipline)
- You can feed the outputs through LPF kits to get sinewave outputs
- 8 selectable “banks” of frequencies, chosen by 3 input control signals
- Frequency range approx 2kHz to 200MHz from onboard Si5351A or MS5351M
- Extended frequency range up to approx. 300MHz if you don’t mind violating the Si5351A datasheet specifications
- Quadrature output mode (Clk0 and Clk1 on same frequency but configurable 0, 90, 180 or 270-degree phase offset)
- GPS frequency discipline using 1pps from a GPS receiver
- Power supply voltage 3.5 to 12V DC
- Frequencies and configuration stored in non-volatile memory for next power-up
- QRP Labs Firmware Update (QFU) bootloader

PLEASE READ THE ENTIRE MANUAL USE INSTRUCTIONS VERY CAREFULLY BEFORE APPLYING POWER TO THE BOARD!

2. Design

The ProgRock2 schematic is shown below.



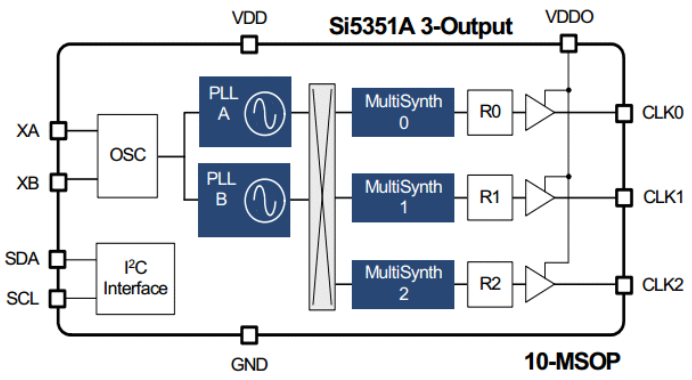
This is a very simple circuit consisting of:

- STM32 Cortex M0 CPU controller
- Si5351A/MS5351M triple clock generator
- 25MHz 0.25ppm TCXO reference (Temperature Controlled Crystal Oscillator)
- MIC5219 3.3V voltage regulator
- Micro-USB connector
- Pads for other connections

All these are SMD components, pre-assembled at the PCB factory, onto the tiny ProgRock PCB.

The Si5351A synthesizer chip is now used in many other QRP Labs products such as the QCX CW transceiver series and QDX digital transceiver. This is a Digital Phase Locked Loop (PLL or DPLL) synthesizer which provides three separate frequency outputs, each having a frequency range spanning 3.5kHz to 200MHz. The frequency stability is governed by the a crystal reference.

Due to unavailability of the Si5351A, the equivalent MS5351M may be used. For test details on the performance of Si5351A vs MS5351M demonstrating the suitability of the MS5351M (in fact, slight superiority in many regards), please see http://qrp-labs.com/synth/ms5351m.html



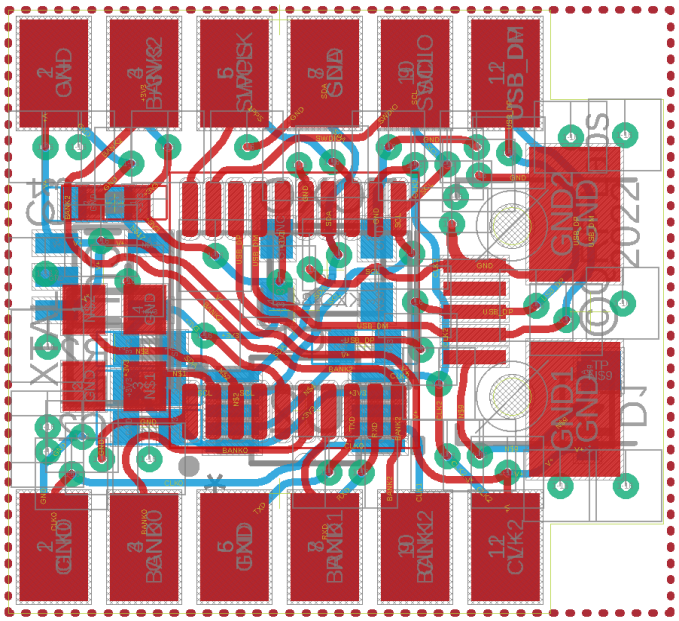
The block diagram (right) is taken from the SiLabs Si5351A datasheet. Briefly, the 27MHz reference oscillator is multiplied up to an internal Voltage Controlled Oscillator in the range 600-900MHz (the PLL), then divided down to produce the final output frequency. The multiplication up and the division down are both fractional and so the frequency resolution is extremely finely controlled. The chip has two PLLs and three output

divider units. The chip must be configured using its serial I2C interface. R1 and R2 are 1K pullup resistors required for the I2C bus.

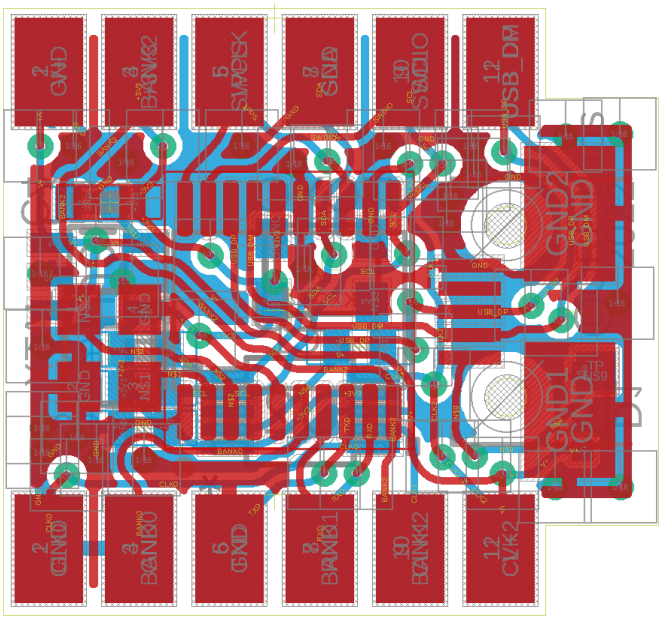
For high frequency stability, a 0.25ppm TCXO is used (the same TCXO as used in QCX-series CW transceivers and QDX digital transceivers).

The Si5351A does not preserve its configuration registers through a power cycle. To provide persistent storage of parameters, the microcontroller needs to retain the configuration parameters in non-volatile storage. STM32-series microcontrollers do not have onboard EEPROM and to save parts count and board area, no additional I2C EEPROM chip has been used in ProgRock2. Instead, the top 1KByte sector of the Flash memory of the STM32 is used as a non-volatile storage area for the configuration parameters.

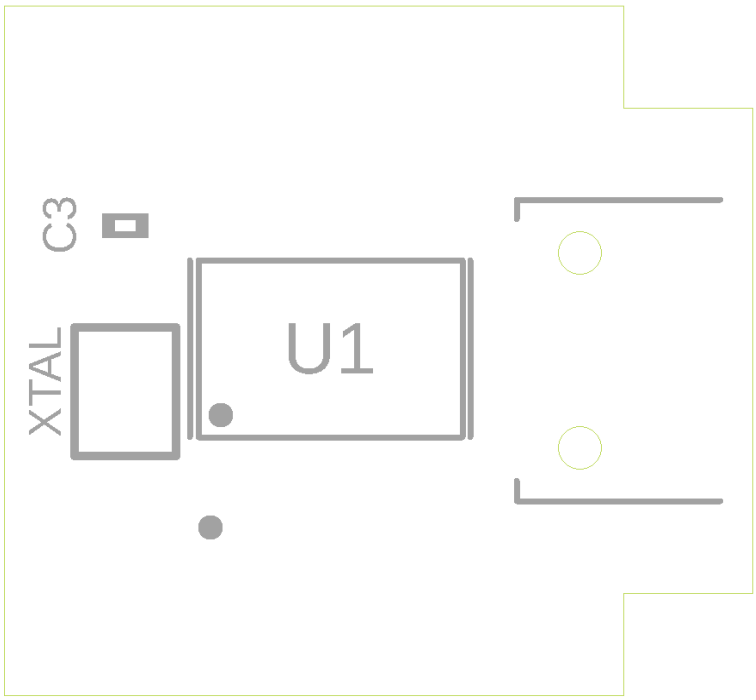
PCB Trace diagram:



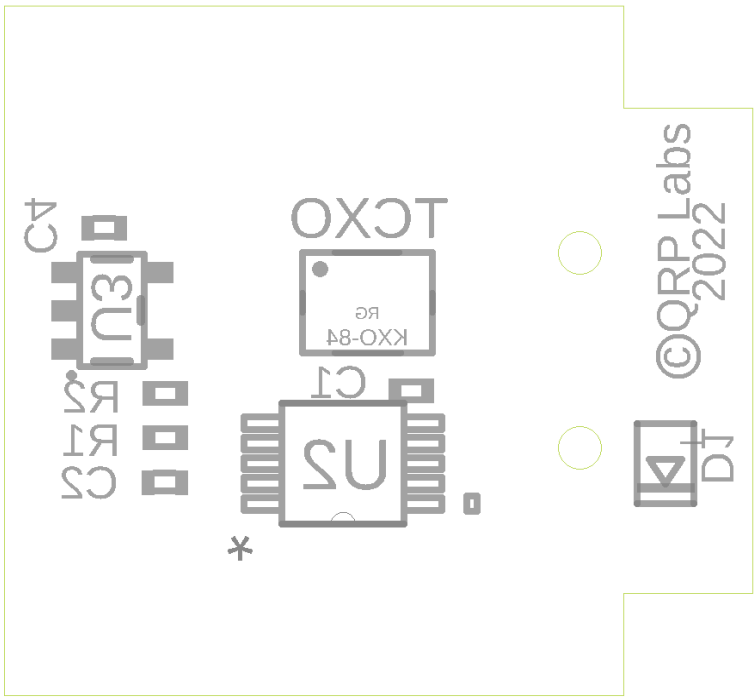
PCB Trace diagram with groundplanes:



PCB component layout, top side:



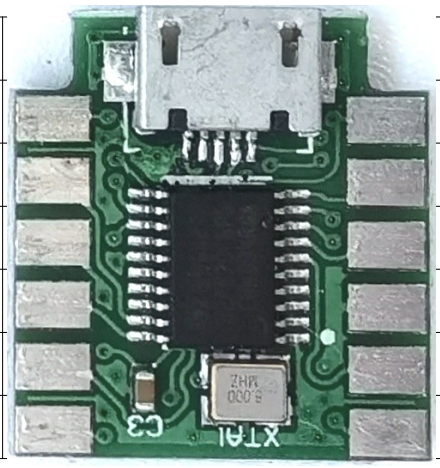
PCB component layout, bottom side:



3 Connections

This table shows the pinout of the ProgRock2 module:

Bottom	Top		Top	Bottom
13. USB -	1. USB +		12. V+	24. CLK 2
14. SCL	2. SWDIO		11.BANK 2	23. CLK 1
15. SDA	3. GND		10.BANK 1	22. RXD
16. 1 PPS	4. SWCLK		9. GND	21. TXD
17. +3V3	5. BANK 2		8. BANK 0	20. GND
18. GND	6. V+		7. CLK 0	19. GND



You will note that there is a row of 6 pads along each long edge of the PCB, on top and bottom sides, making a total of 24 possible connections. Some, such as GND, are on multiple pads. For convenience these are labeled 1 to 24. In the diagram above, the pads on the top side are shown closest to the PCB pads, and the bottom side pads are the outer table column.

The layout was carefully designed to allow as much future flexibility as possible.

The signals are as follows:

Signal	Pins	Description
GND	3, 9, 18, 19, 20	Ground.
V+	6, 12	Positive supply voltage.
+3V3	17	3.3V output from onboard voltage regulator.
USB-, USB+	13, 1	USB port (also connected to micro-USB connector); note, also sometimes called USB_DM, USB_DP respectively.
SWDIO, SWCLK	2, 4	Chip programming pins: Factory use only.
SCL, SDA	14, 15	I2C serial bus (internal use only, currently).
RXD, TXD	22, 21	USART port (unused currently).
BANK 0	8	Bank select input 0 (3.3V max).
BANK 1	10	Bank select input 1 (3.3V max).
BANK 2	5, 11	Bank select input 2 (3.3V max).
CLK 0	7	Si5351A Clock 0 output.
CLK 1	23	Si5351A Clock 1 output.
CLK 2	24	Si5351A Clock 2 output (not available when using GPS discipline).
1 PPS	16	1 PPS input, connect to 1pps output of GPS for GPS discipline. Note that the signal requires 3.3V. If using a 5V GPS output, use a 3.9K series resistor (value not critical; 3.9K was used during testing).

Supply voltage:

ProgRock2 current consumption is approximately 30-35 mA. It does vary a little depending on Si5351A output loading, the configured outputs and their frequency.

The MIC5219 datasheet specifications for voltage and power dissipation need to be observed. According to the MIC5219 datasheet the input voltage range, for a 3.3 V 50 mA output is +3.5 to +12 V.

At a +12 V supply and 30 mA current consumption the power dissipation in the voltage regulator will be 261 mW. According to the MIC5219 datasheet this level of power dissipation is acceptable. Therefore even a +12V supply may be used. However, 261 mW is quite a lot of power dissipation (and hence heating) so you may wish to bear that in mind and use a lower supply voltage if possible.

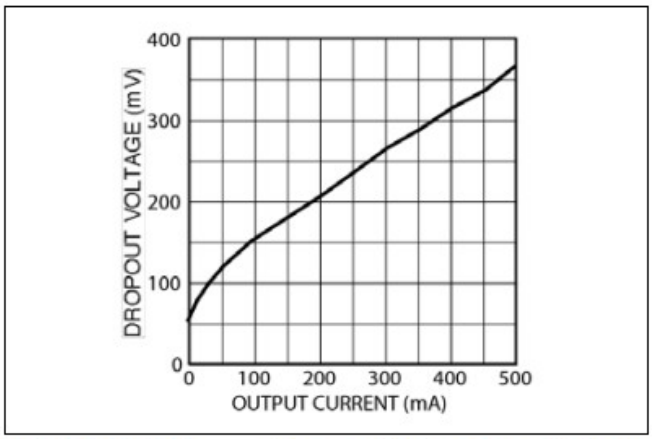


FIGURE 2-11: *Dropout Voltage vs. Output Current.*

The minimum acceptable supply voltage is determined by the dropout voltage graph in the MIC5219 datasheet which for 30mA load is approx 100mV. I'd suggest allowing a little safety margin and supply ProgRock2 with at least 3.5V.

Care should be taken when powering ProgRock2 directly from the USB cable (see below), and using a power supply connected to +V at the same time. The USB cable +5V will power ProgRock2 via an onboard diode, resulting in about 4.4V supply to ProgRock2. If you have connected an additional external supply voltage to ProgRock2, and that is less than 4.4V, then your external supply will fight with the USB voltage, potentially drawing excessive current through the onboard diode.

3.3V output

The regulated 3.3V output from the onboard voltage regulator is provided on one of the pads. If you use this, please be sure to observe all MIC5219 datasheet specifications regarding loading etc.

Bank selection

Bank selection inputs BANK 0, BANK 1 and BANK 2 may be used to select one of 8 banks of three frequencies programmed into the module. They have internally activated pull-up resistors. To activate a BANK input, it should be grounded.

For example, if you want to select bank 5: ground the BANK 0 and BANK 2 inputs (4 + 1 = 5).

If the BANK pins are left unconnected, bank 0 is active by default.

Note that these processor I/O pins should not be connected to a voltage higher than 3.3V.

Clock outputs

CLK 0, CLK 1 and CLK 2 are the three clock outputs from the Si5351A synthesizer. They are unbuffered, direct connections to the Si5351A chip. As such, they should be connected with care, so as not to damage the Si5351A chip.

The outputs are 3.3 V peak-to-peak squarewaves, with a declared output impedance in the Si5351A datasheet of 50-ohms. What this appears to mean in practice is that if you connect a 50-ohm load, the output will be reduced by 50%, to 1.65 Vpp.

For best phase noise performance, as well as least crosstalk between Si5351A outputs, it is recommended to use loads of at least 1 K-ohm.

USB connection

ProgRock2 has a micro-USB connector for connecting to a host PC terminal emulator, for configuration of the ProgRock frequencies.

+5V power may also be supplied by the micro-USB connector. There is a diode feeding the +5V connection of the USB cable to the +V supply voltage of the ProgRock2. So if you connect only a USB cable, ProgRock2 will be supplied from that. If you are supplying an external voltage below 4.4V damage can occur, please see section above regarding power supply.

The USB connections are also available as pads on the edge connectors.

1pps

If a positive-going 1pps signal from a GPS module such as the QRP Labs QLG2 (http://qrp-labs.com/qlg2) is connected here, the ProgRock2 frequencies will be GPS disciplined to a high accuracy. However CLK 2 is used by this process so becomes unavailable for your independent configuration. CLK 2 is configured to 9,999,999.5 Hz for the purposes of the GPS discipline procedures. GPS discipline mode is only cleared after cycling the power (when no GPS 1pps signal is connected).

This input pin requires a signal of not more than 3.3V. If you are using the 5V output of a QLG2, then a 3.9K series resistor is recommended (value not critical, 3.9K was what was used during testing and development). Alternatively the QLG2 can be configured by jumper wires to provide 2.8V output logic, this would be suitable for direct connection to the ProgRock2 input.

Unused connections

SWDIO, SWCLK are In-Circuit-Programming connections used only by the factory during initial download of the bootloader into the ProgRock2 module. They have no further use.

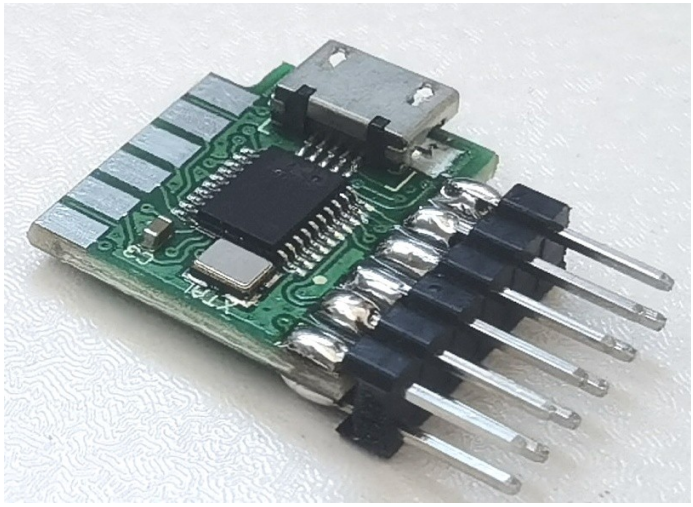
SCL, SDA are the I2C bus (used for processor communication with the Si5351A) and have no further use currently. They are provided on the edge connector for possible future expansion possibilities.

TXD, RXD are a USART transmit/receive signal pair but are unused. The pads are provided for possible future expansion.

4 Installation ideas

There are many ways to install a ProgRock2 module in your project. The most obvious is just to solder wires to the PCB itself. But here are some more ideas!

2x6-pin header strip

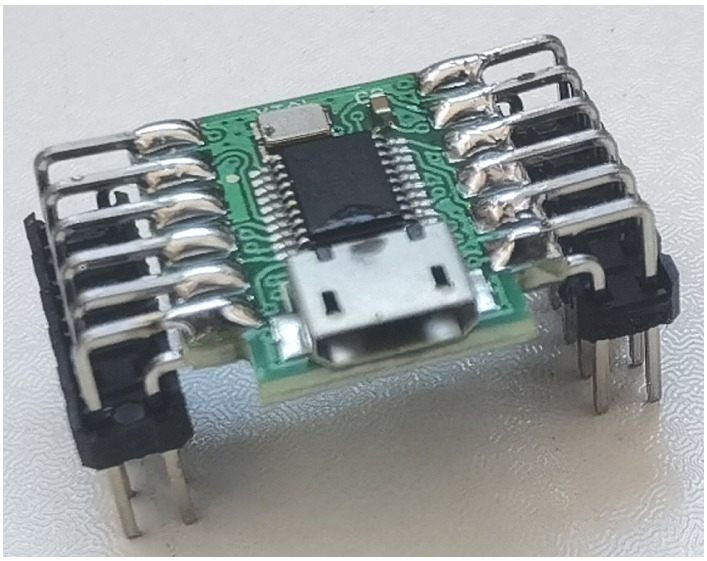


A suitable pair of 2x6-pin header strips is available as an option when purchasing ProgRock2. The PCB slides between the two rows of pins (the short ends), which can then be soldered to the pad surface. This could be done on both long edges of the PCB for the full 24 pins of the connection, if female header connectors and cables were used.

Or – just connect a single 2x6-pin header along one edge, and have the ProgRock2 module stand up vertical on your project board, soldered in or plugging into a 2x6 female header socket.

The pinout was quite carefully designed such that power supply lines, Bank select inputs and Clock outputs are all available on the edge shown.

Pair of 2x6-pin right-angled header strips

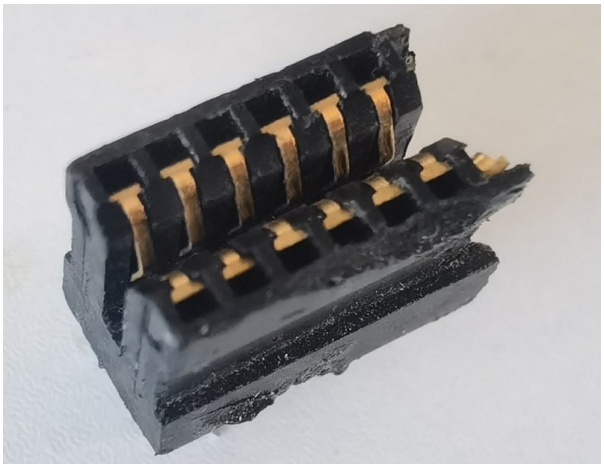


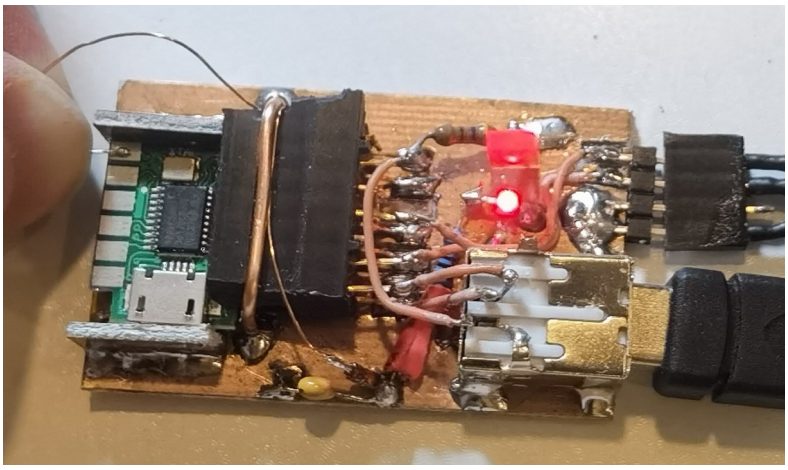
Another way is a pair of right-angled header strips as shown in the photograph. This way your module has 24 pins and can be plugged into either female pin header connectors on your board or soldered in directly, with the module flat in the same plane as your board.

These right-angled headers are available as an option when you purchase your ProgRock2 module.

Old-school edge connectors

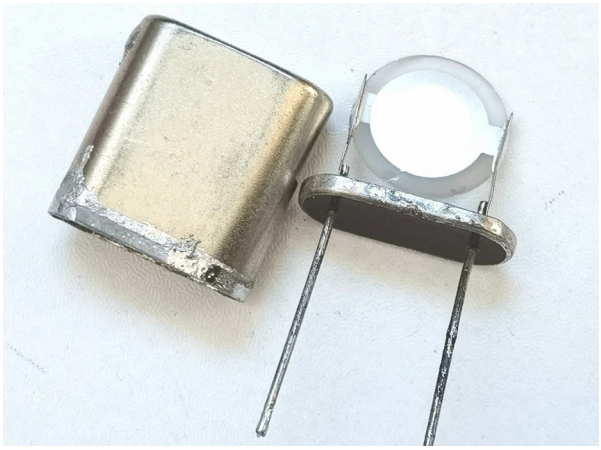
The 0.1-inch pitch pads fit in an old-school PCB edge connector socket. This is what we did at QRP Labs HQ to build the test and bootloader flashing jig.

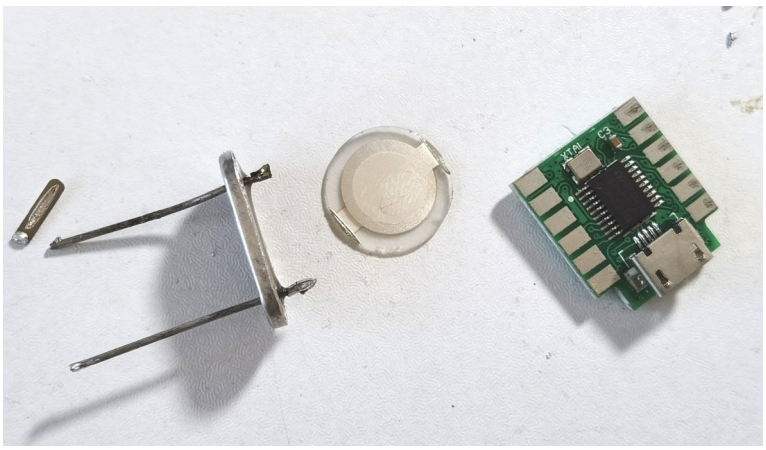


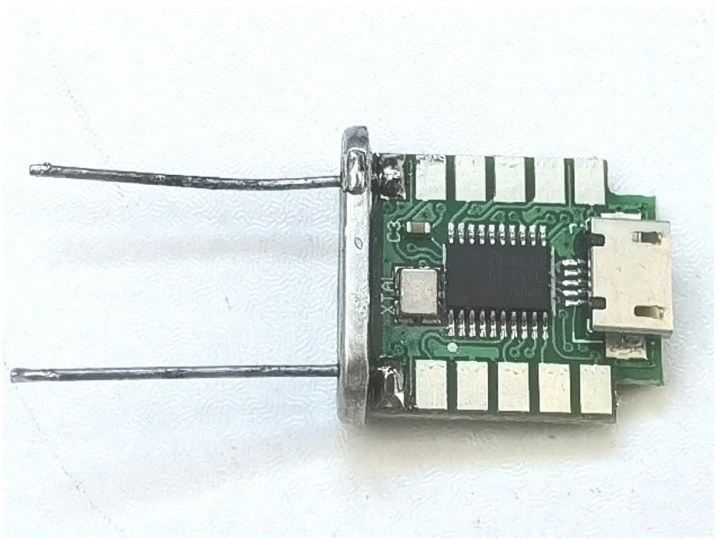


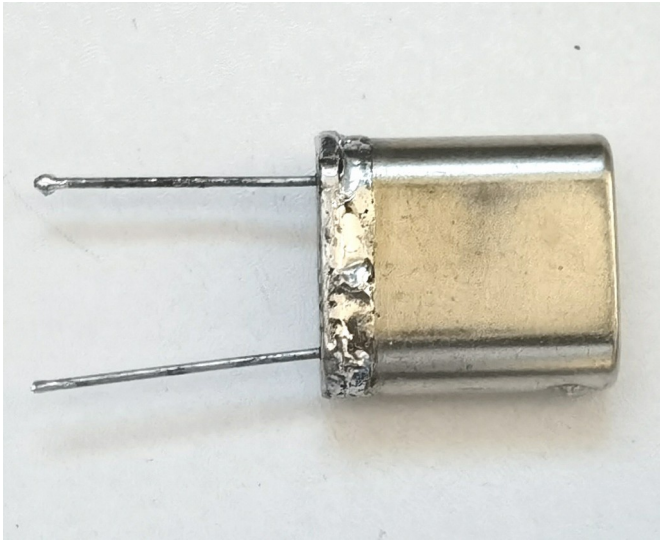
Install in an HC6 crystal case

The pin layout was carefully designed so that at the left end, furthest from the micro-USB connector, has pads for Gnd, +V and the CLK 0 output. +V and CLK 0 can be soldered conveniently to the HC6 crystal holder pins, and a wire for Ground connected to the metal HC6 case.









5 Connecting terminal emulator

Drivers

No additional drivers are required for operation with most Linux distributions, Apple Mac or MS Windows 10 or Windows 11.

For older versions of MS Windows, it may be necessary to install a driver for the serial port because this driver is not on your computer already by default. This driver is available from the ST Semiconductor website at https://www.st.com/en/development-tools/stsw-stm32102.html and is applicable to 98SE, 2000, XP, Vista®, 7, and 8.x Operating Systems. There is a description for installation on Windows 7/8 on the QRP Labs QLG2 page http://qrp-labs.com/qlg2 so if in doubt, please check this.

Linux special note

On Linux systems, a particular problem can occur. When the ProgRock2 (Serial) connection is detected, the PC thinks that a modem has been connected and starts trying to send it Hayes AT-commands dating back to 1981, implemented on Hayes' 300-baud modem. Yes! 40 years ago…

The Operating System attempting to send AT commands to your ProgRock2 will certainly mess everything up. Not least because when ProgRock2 receives a carriage return character, it will enter Terminal Applications mode; this will send all sorts of characters back to the PC (as ProgRock2 thinks it is now talking to a terminal emulator) and it will disable CAT command processing, so your PC digi modes software will not be able to talk to ProgRock2. Disaster.

To fix this you need to issue the following commands to disable ModemManager:

```
sudo systemctl stop ModemManager
sudo systemctl disable ModemManager
sudo systemctl mask ModemManager
```

This will permanently stop ModemManager. If for some reason, you actually DO need ModemManager operational, for some other reason… well there IS a way to stop it just for ProgRock2… but Google will be your elmer on this!

Additional information from Greg Majewski:

There is another Linux service, BRITTY, that does the same. BRITTY is a Braille service for access by sight impaired people. I have encountered the problem with the G90 and Ubuntu on a laptop (Ubuntu full version), Raspberry Pi 3 with Raspberry OS and the Orange PI 800. Here are commands that remove BRITTY:

sudo systemctl stop brltty-udev.service

sudo systemctl mask brltty-udev.service

note output: Created symlink /etc/systemd/system/brltty-udev.service

→ /dev/null.

sudo systemctl stop brltty.service

sudo systemctl disable brltty.service

These commands are similar as used for Modem Manager service.

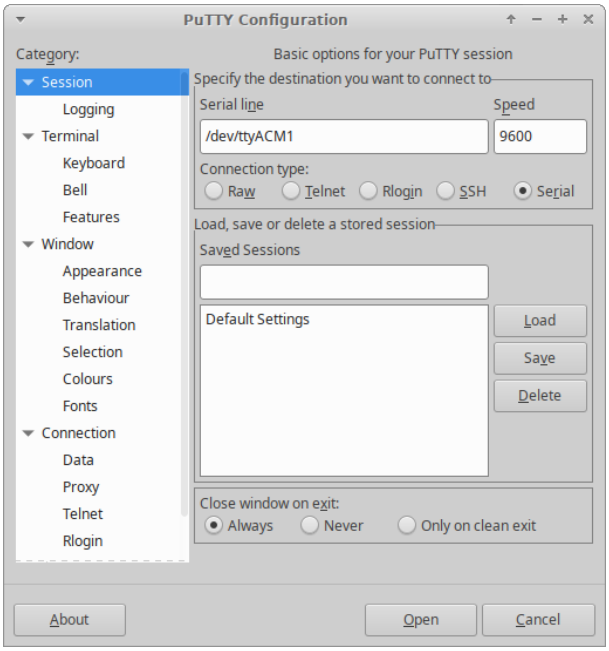
Terminal emulator

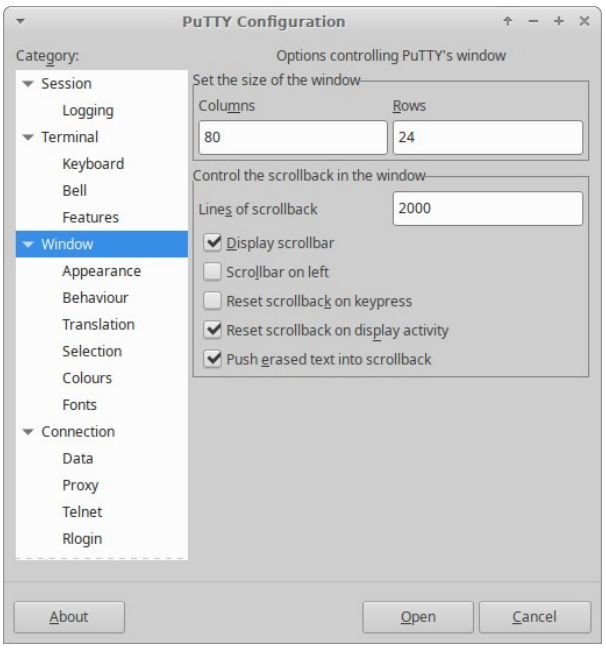
I use Linux (XUbuntu 18.04) and I'm using the PuTTY terminal emulator. There are numerous other terminal applications which will work fine. You may have your own favourite. They are all capable of correct operation with ProgRock2.

I start PuTTY using command line "sudo putty" then connect to ProgRock2 on /dev/ttyACM0. It is necessary to know which serial port is being used by ProgRock2. There is also a guide to identifying the serial port at http://qrp-labs.com/qlg2 (scroll down the page).

Note that only one PC application at a time can connect to Virtual COM Serial ports.

Set the size of the terminal window to 80 columns and 24 rows.





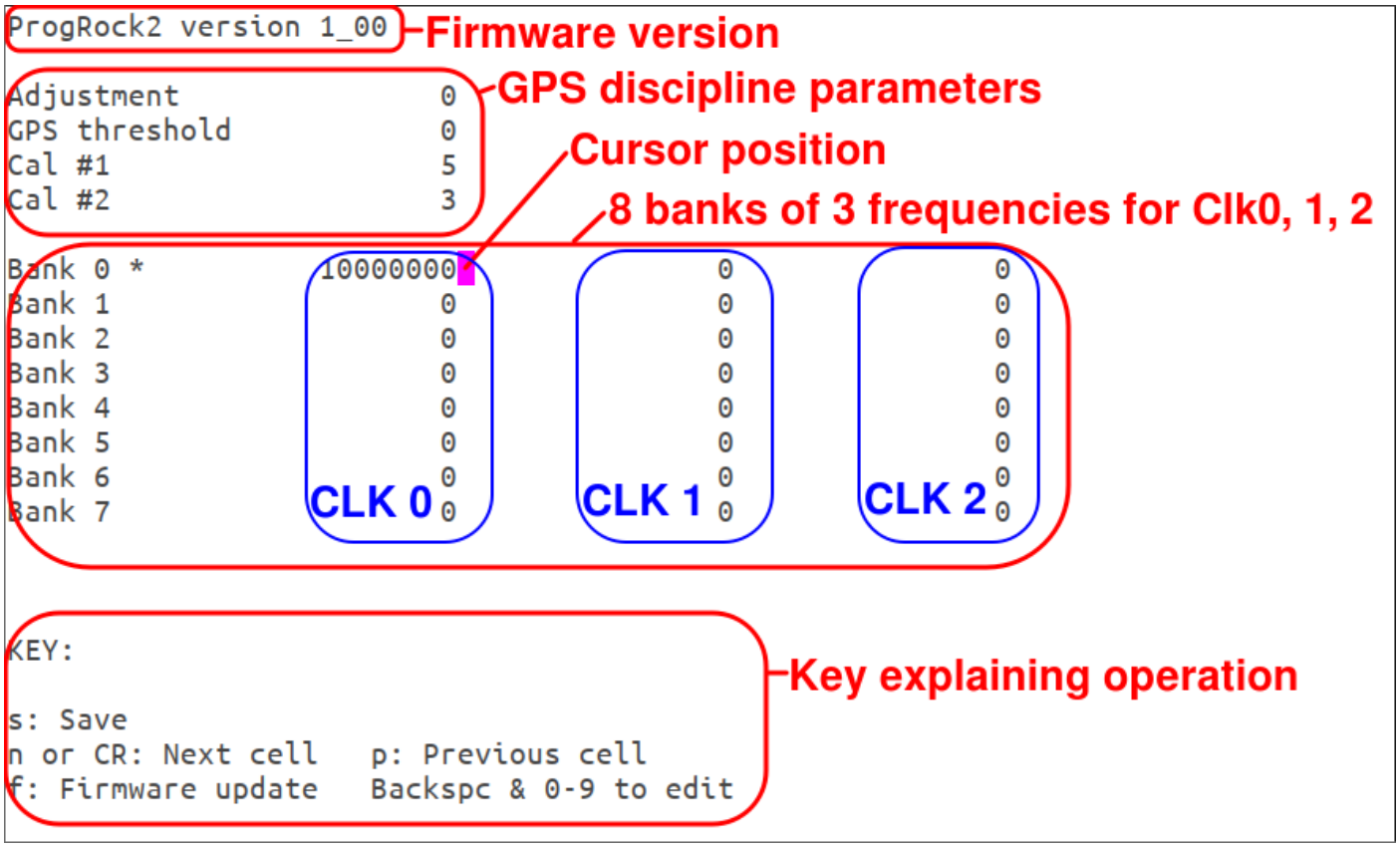
When you click "Connect", a blank screen should be displayed. Now press the ENTER key, to enter the ProgRock2 configuration utility.

Configuration utility

The single screen configuration utility for ProgRock is very minimal, due to the resource constraints of the microcontroller. It looks like this:

```
ProgRock2 version 1_00

Adjustment                0
GPS threshold             0
Cal #1                    5
Cal #2                    3

Bank 0 *           10000000                0                0
Bank 1                    0                0                0
Bank 2                    0                0                0
Bank 3                    0                0                0
Bank 4                    0                0                0
Bank 5                    0                0                0
Bank 6                    0                0                0
Bank 7                    0                0                0



KEY:

s: Save
n or CR: Next cell    p: Previous cell
f: Firmware update    Backspc & 0-9 to edit
```

The notable screen elements are identified in this annotated version:



Editing

Use the n or CR key to move to the next cell
Use the p key to move to the previous cell
Use backspace and the number keys to edit a cell

Editing may seem rather counter-intuitive because the cursor (arrow) keys do not function. The reason for this is that the ProgRock2 microcontroller has limited Flash memory so a very slimline editing tool was required, that does not take up too much code space.

The currently selected memory bank is indicated by the asterix next to the bank column – in the above example, Bank 0 is selected.

Frequencies set to zero cause the corresponding Si5351A output to simply be switched off.

When editing is complete, press the 's' key to save the configuration to non-volatile memory (Flash memory).

Phase offset modes

CLK 1 may be configured to operate on the same frequency as CLK 0 but have a defined (accurate) phase shift. This is useful for example, for directly driving quadrature sampling detector type mixers. To enable this mode, set the CLK 1 frequency to one of 90, 180, 270 or 360.

- 90: CLK 1 is on the same frequency as CLK 0 with a 90-degree phase shift
- 180: CLK 1 is on the same frequency as CLK 0 with a 180-degree phase shift (CLK 1 is the inverse of CLK 0 – useful for driving Push-Pull circuits such as amplifiers)
- 270: CLK 1 is on the same frequency as CLK 0 with a 270-degree phase shift – useful for selecting the opposite sideband to the 90-degree setting
- 360: CLK 1 and CLK 0 are identical in phase and frequency

Limitations:

There are certain limitations on frequency in the various modes, as follows:

1. The lowest frequency attainable is a little under 3.5 kHz
2. The highest frequency is 228 MHz
3. If one output is more than 150 MHz, no other outputs may be configured
4. If GPS discipline is applied, CLK 2 is set to 9,999,999.5 Hz regardless of any configuration settings
5. When using CLK 1 in 90-degree or 270-degree mode, the output frequency precision may be slightly lower than normal
6. When using CLK 1 in 90-degree or 270-degree mode, the minimum frequency for CLK 0 and CLK 1 is a little under 3 MHz.

GPS Discipline

GPS discipline of the frequency occurs when a positive-going 1pps signal from a GPS such as the QRP Labs QLG2 http://qrp-labs.com/qlg2 is connected to the ProgRock2 module. In this mode, CLK 2 is set to 9,999,999.5 Hz regardless of the CLK 2 configuration setting, and this frequency is measured using the 1pps leading edge as a frequency counter gate.

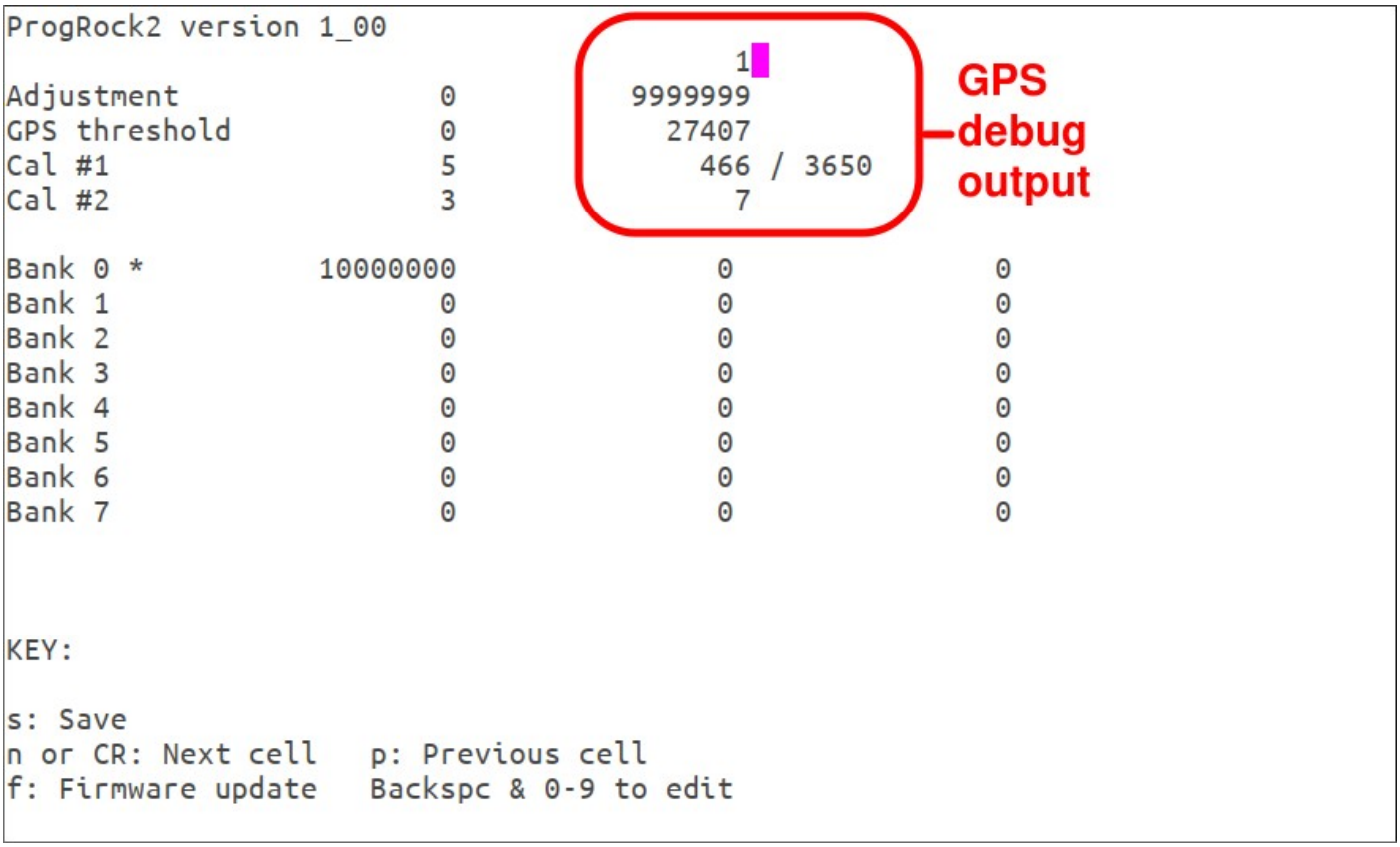
GPS discipline is controlled by the parameters on the configuration screen, as follows:

- **Adjustment:** this is the error of the reference output, expressed in units of 1 / 3650'ths of a Hz at 9,999,999.5 Hz. This value is loaded at power-up. However when GPS discipline is applied, the value is internally updated. It is not saved to memory; when you press 's' to save the configuration, the current live GPS discipline value is copied into the memory (startup) parameter.
- **GPS threshold:** When in 90-degree and 270-degree phase offset modes, this setting controls by how many 3650'ths of a Hz at 9,999,999.5 Hz the internally calculated adjustment parameter must change, since the last time output oscillator clocks are updated, until the next time output oscillator clocks are updated. When not in phase offset modes, there is always a very slight adjustment to the output frequencies, at every single 1 second interval.
- **Cal #1:** this is the minimum step, expressed in units of 1 / 3650'ths of a Hz at 9,999,999.5 Hz, applied at each GPS discipline measurement interval (1 second). You could use a smaller value, for example, if you have well enclosed ProgRock2 such that any temperature variations will be slow. Or in some cases, you might wish to increase the value, to cope faster with sudden extreme temperature variations. The size of the actual step at each interval is variable; when GPS discipline is initialized, the step is large (2,000) in order to achieve rapid convergence (calibration); thereafter it is gradually reduced until it hits the minimum step size. If the GPS discipline algorithm notes rapid drift of the oscillator frequency, it starts to increase the dynamic step size in order to rapidly correct for the drift.
- **Cal #2:** this controls an additional control loop which acts to keep the cumulative error on the 9,999,999.5 Hz signal to precisely zero. In other words the average long-term frequency tends to 9,999,999.5 Hz exactly. The parameter is the additional step size correction, which is applied on top of Cal #1, at each 1 second measurement interval where the cumulative error exceeds 0.5 Hz in total since GPS discipline was enabled. Cal #2 can be set to zero if you do not wish to use this secondary control loop feature.

GPS debug display

It is possible to enable a view of the live GPS discipline parameters for debug purposes by pressing the ‘g’ key once.

Then the screen appears as below.



To explain these five debug parameters in turn:

1. The first parameter (1 here) is the cumulative frequency measurement error since GPS discipline began. Remember that frequency measurement in a 1 second interval has a 1 Hz resolution. So when you see “1” here it does not mean that the frequency is 1 Hz wrong. But if you see a number larger than 1, this means that some accumulated frequency error has occurred; if the Cal #2 parameter is non-zero it will gradually act to remove any accumulated error, reducing it to 0.
2. Frequency measurement in the last 1 second interval. When GPS discipline has homed in on the correct adjustment amount, this measurement will alternate between 9,999,999 and 10,000,000 with an average therefore of exactly 9,999,999.5 Hz.
3. 27407 is just the current value of the internal 16-bit counter used for the frequency measurement. It will reduce by 1 every 2 seconds (0.5 Hz per second, to reflect the 0.5Hz offset from 10.000000 MHz).
4. Fourthly, 466 / 3650 (in this case) is the current adjustment amount, and the constant 3650 to remind you that the adjustment amount is expressed in 3650’th of a Hz at 9,999,999.5 Hz.
5. The last parameter (7 in this case) is the current step size, the amount the adjustment is adjusted, after every 1 second measurement period. The minimum value is the value expressed in Cal #1.

GPS debug Log

Pressing ‘g’ key one more time, produces a scrolling log of the GPS discipline debug parameters. With most terminal emulators (including PuTTY) you can enable a log file. So if you want to chart what happens to the GPS discipline under various conditions, this a useful tool.

When the GPS debug display is active or the GPS log is active, normal editing of the configuration parameters is disabled.

So pressing ‘g’ key cycles between:

- Ordinary parameter editing operation
- GPS debug display (parameter editing is disabled)
- GPS debug log (parameter editing is disabled)

6. Firmware update procedure

One of the absolute best features of ProgRock2 is that firmware updates can be done easily, by anyone, with no special hardware, software, tools or experience. This feature is called “QFU” (**Q**RP Labs **F**irmware **U**pdate) and provides the following features:

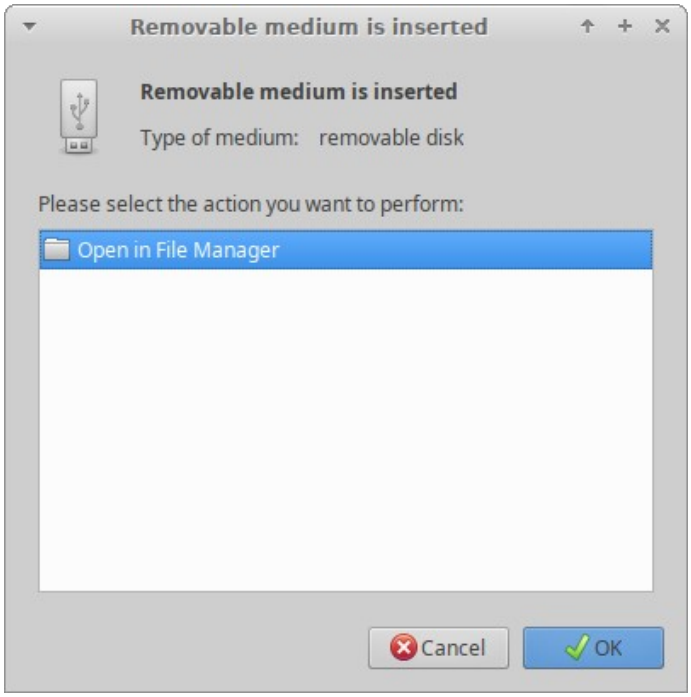
- **Easy** – anyone can do the firmware update
- **No additional hardware required:** only a standard USB A-B cable (or micro-USB cable if you have installed a micro-USB connector)
- **No additional software required:** just the standard file manager application that is already available on any PC
- **No drivers:** no special drivers need to be installed, the existing drivers on any modern operating system are used
- **Works on any PC Operating System:** and in the same way: Windows, Linux, Mac
- **Secure:** firmware files are published on the QRP Labs website and are encrypted using 256-bit AES encryption technology

USB Flash memory stick emulation:

In the firmware update mode, the ProgRock2 module pretends to be a USB Flash memory stick, having a 4MByte capacity and implementing a FAT16 file system. This virtual “Flash stick” contains a single file, the firmware program file in the ProgRock2 microcontroller. You may read the file from the ProgRock2, or write a new one, just by dragging files in your file manager application.

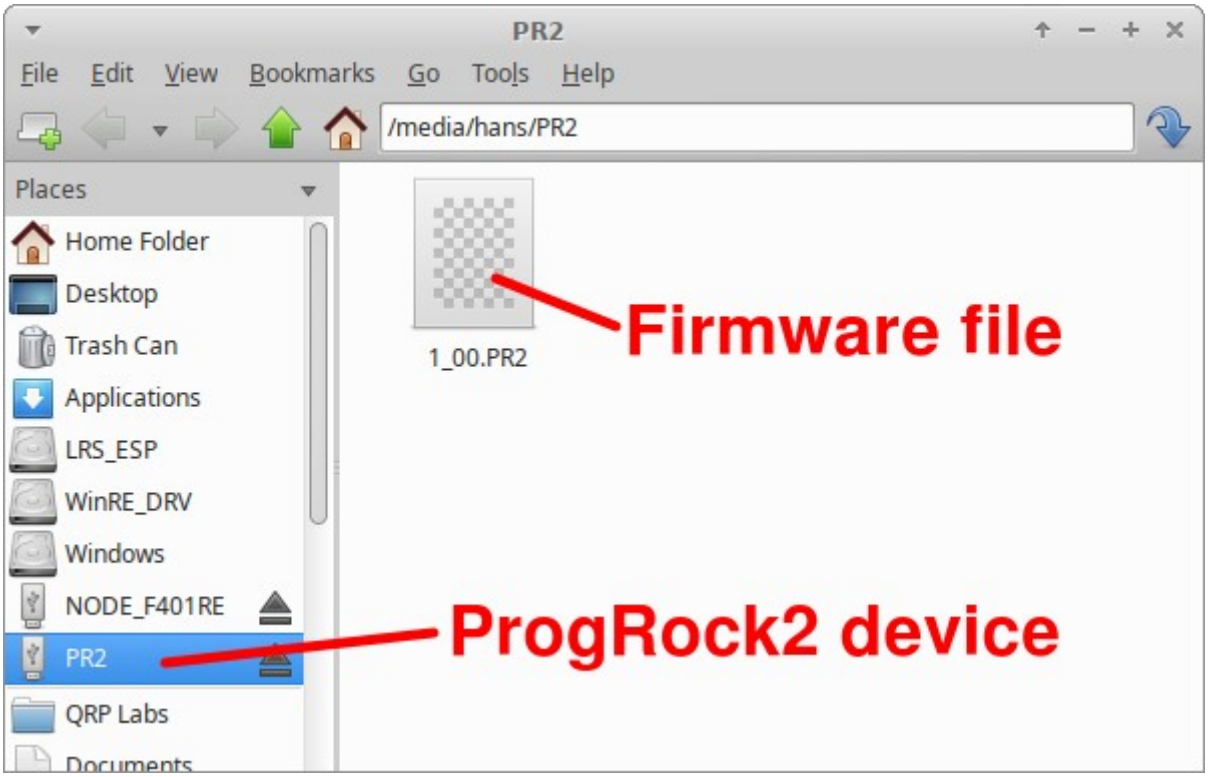
Entering bootloader (firmware update) mode:

To enter bootloader mode, simply press ‘f’ on the terminal configuration screen. When you press f, the terminal screen will close automatically, and a pop-up window should appear on your PC. On my system (Linux XUbuntu 18.04) it looks like this:



Click the OK button.

The File Manager window will then open, and on my system looks like this:



The ProgRock2 appears as a removable USB Flash device named “PR2”, and the folder shows a single file which is the firmware version file, 1_00.PR2 in this example.

The file name must not be longer than 8 characters, and cannot contain punctuation or spaces; the file extension must be no more than 3 characters (hence “PR2”). This is because the file system emulation is FAT16 and these are the specifications of the FAT16 format.

You may check the properties of the file and will note that it is a 23.5K file. ProgRock2 firmware images are always a 23.5K file. The creation date and modification date etc. have not been set, because it was important to minimize the size and complexity of the ProgRock2 QFU bootloader, in order to maximize the space available to the application firmware.

You may copy the existing firmware file to another directory of your computer. Crucially, to do the firmware update, all you need to do is copy the new firmware file to this “Flash disk”. Download the new firmware file from the QRP Labs website, unzip it, and simply drag it into the folder where the existing firmware file version is shown. Or copy and paste it, however you wish.

As soon as you copy the new file to the ProgRock2 “flash drive”, the ProgRock2 QFU bootloader erases the current program from its memory and installs the new one.

The ProgRock2 firmware is 256-bit AES encrypted and this means:

- The encrypted ProgRock2 firmware file will only work on a QRP Labs ProgRock2 board, it cannot be installed on any other board, even one containing the same processor.
- No other firmware file will work on the QRP Labs ProgRock2 board except an official QRP Labs encrypted ProgRock2 firmware file.

The procedure will vary slightly for different Operating systems but in all cases is just a simple matter of copying the new firmware file to the emulated ProgRock2 USB Flash drive.

The above firmware update procedure works on ANY modern OS because the QFU bootloader emulates a USB Flash memory stick with the USB Mass Storage Device (MSD) class, for which drivers are already present.

The QFU bootloader implements a USB device stack (Mass Storage Device class), emulated FAT16 file system, Flash erase/write, and 256-AES encryption. It occupies the first 10K of Flash memory leaving 21K for the application itself and 1K for the non-volatile storage of the configuration parameters.

Important notes about the ProgRock2 firmware implementation

1. The only way to enter firmware update mode, is to press ‘f’ on the terminal screen
2. The only way to get out of firmware update mode, is to update the firmware by copying in a new firmware file. Even power cycling doesn’t get you out of firmware update mode. If you enter firmware update mode by mistake, you can just to a “Copy and paste” of the current firmware in the directory. This will overwrite the firmware with itself, which is pointless except that it does get you out of the firmware update mode.
3. When you execute a firmware update, the stored configuration parameters are set back to their defaults; so effectively a firmware update is also a factory reset.

7. Resources

- For updates and tips relating to this kit please visit the QRP Labs QDX kit page http://qrp-labs.com/progrock2
- For any questions regarding the assembly and operation of this kit please join the QRP Labs group, see http://groups.io/g/qrplabs for details

8. Document Revision History

1.00	16-Feb-2023	First draft version version 1.00
1.00a	16-Feb-2023	Added some examples of installing ProgRock2
1.00b	17-Feb-2023	Added notes about GPS mode and 1pps signal being 3.3V