QRP Labs “ProgRock” programmable crystal

1. Introduction

Thank you for purchasing this QRP Labs kit. This kit comprises two modules:

- Si5351A Synthesiser module kit
- ProgRock controller kit for the Si5351A Synthesiser

You need to download the assembly instructions for the Si5351A synthesiser module kit separately. The synth kit plugs into the ProgRock controller kit’s two 10-pin header sockets. This document describes the assembly and operation of the ProgRock kit.

This kit gets its name “ProgRock” from the potential use as a programmable crystal replacement. A lot of applications arise where one needs a crystal of an unusual frequency, which cannot be found at any supplier or in the junk box. Custom crystal manufacture is expensive and time-consuming. So “ProgRock can replace those unusual crystals! ProgRock provides THREE independent outputs, and 8 banks of selectable frequencies.

Additionally, by connecting a 1 pulse-per-second (1pps) signal from a GPS receiver (such as the QRP Labs QLG1 GPS receiver kit), the ProgRock controller will GPS-discipline the Si5351A PLL reference frequency, correcting for any calibration error and temperature-related frequency drift.

Programming of the ProgRock frequencies is by a 4-way DIP switch and button. You will enter the frequencies in binary coded decimal, one digit at a time. A little cumbersome, but you’ll soon get used to it and get fast at it. Anyway this kit is not a VFO – QRP Labs has a different kit for that – this kit is ProgRock, a crystal replacement which does not need to be reprogrammed very often. The good news of course is that all the settings you enter, are stored in the microcontroller’s EEPROM, so that they are ready next time you power up your project.

A short summary of ProgRock features:

- 3 independent 3.3V p-p squarewave outputs (2 if you use GPS discipline)
- 8 selectable “banks” of frequencies, chosen by 3 input control signals
- Frequency range 3.5kHz to 200MHz
- Extended frequency range up to approx. 300MHz if you don’t mind violating the Si5351A datasheet specifications
- GPS frequency discipline using 1pps from a GPS receiver
- Unique power supply noise filter circuit designed by Alan Gray G8LCO
- Power supply voltage 5V, or using LM317LZ regulator (supplied), range is 5-12V DC
- Minimalist configuration user interface using 4-way DIP switch, push button, and LED
- Easy construction, no surface-mount components to solder (Si5351A already pre-soldered)

Recommended approach to building the kit: This is a simple design but there are several features which provide a great deal of flexibility. **Read this WHOLE manual and understand it!** BEFORE you start to build it – so that you know what hardware options you want. Follow the assembly instructions carefully!

2. Parts list

2.1 Resistors

R1	1K (colour coding brown-black-red-gold) – supplied from Si5351A Synth kit
R2	3K (colour coding orange-black-red-gold)
R3, R4, R6	4.7K (colour coding yellow-purple-red-gold)
R5	39K (colour coding orange-white-orange-gold)
R7, R8	560-ohm (colour coding green-blue-brown-gold)

2.2 Capacitors

C1, C3	0.1uF ceramic (body marking “104”) – one supplied from Si5351A Synth kit
C2, C4, C5	22uF electrolytic
Cx	Not required or supplied – see text

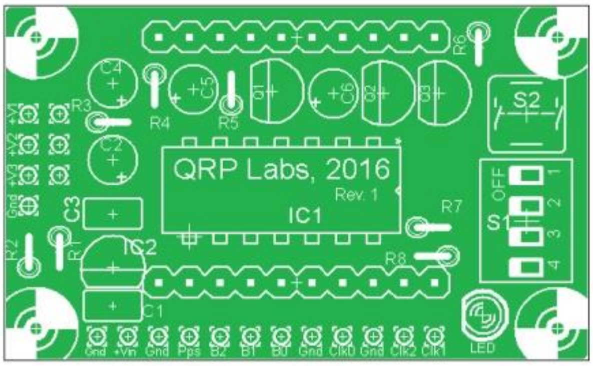
2.3 Semiconductors

LED	3mm red LED
Q1, Q3	2N3904 TO92-package NPN transistor
Q2	2N3906 TO92-package NPN transistor
IC1	ATtiny84 microcontroller, 14-pin DIP package
IC2	LM317LZ voltage regulator, TO92-package – supplied from Si5351A Synth kit

2.4 Miscellaneous

4-way DIP switch, S1
Push button, S2
Printed circuit board, 1.9 x 1.15 inches (48 x 29mm) size
Socket for IC1
Two 10-pin female header sockets
16-pin male header plug

Note: Certain components listed above are not supplied in the ProgRock kit bag, they are in the Si5351A Synth module kit bag. This is because the recommended way of assembling the ProgRock kit is to build the noise filter circuit (Q1-Q3) on the ProgRock PCB, instead of the 3.3V regulator circuit on the Si5351A Synth PCB. In this case, the LM317LZ from the Si5351A Synth kit is used on the ProgRock PCB as a 5V regulator. The power supply options are discussed further, below.



PCB 48 x 29mm

2pcs 560-ohm 1/4W

3pcs 4.7K 1/4W

1pcs 3K 1/4W

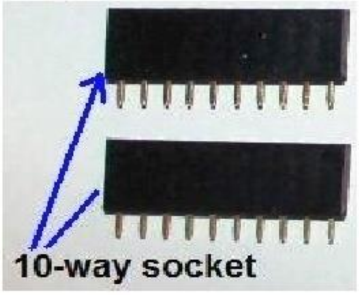
1pcs 39K 1/4W

16-way header

6 x 6 x 6mm button

DIP14 socket

4-way DIP switch



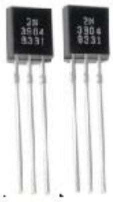
10-way socket

1pcs 0.1uF

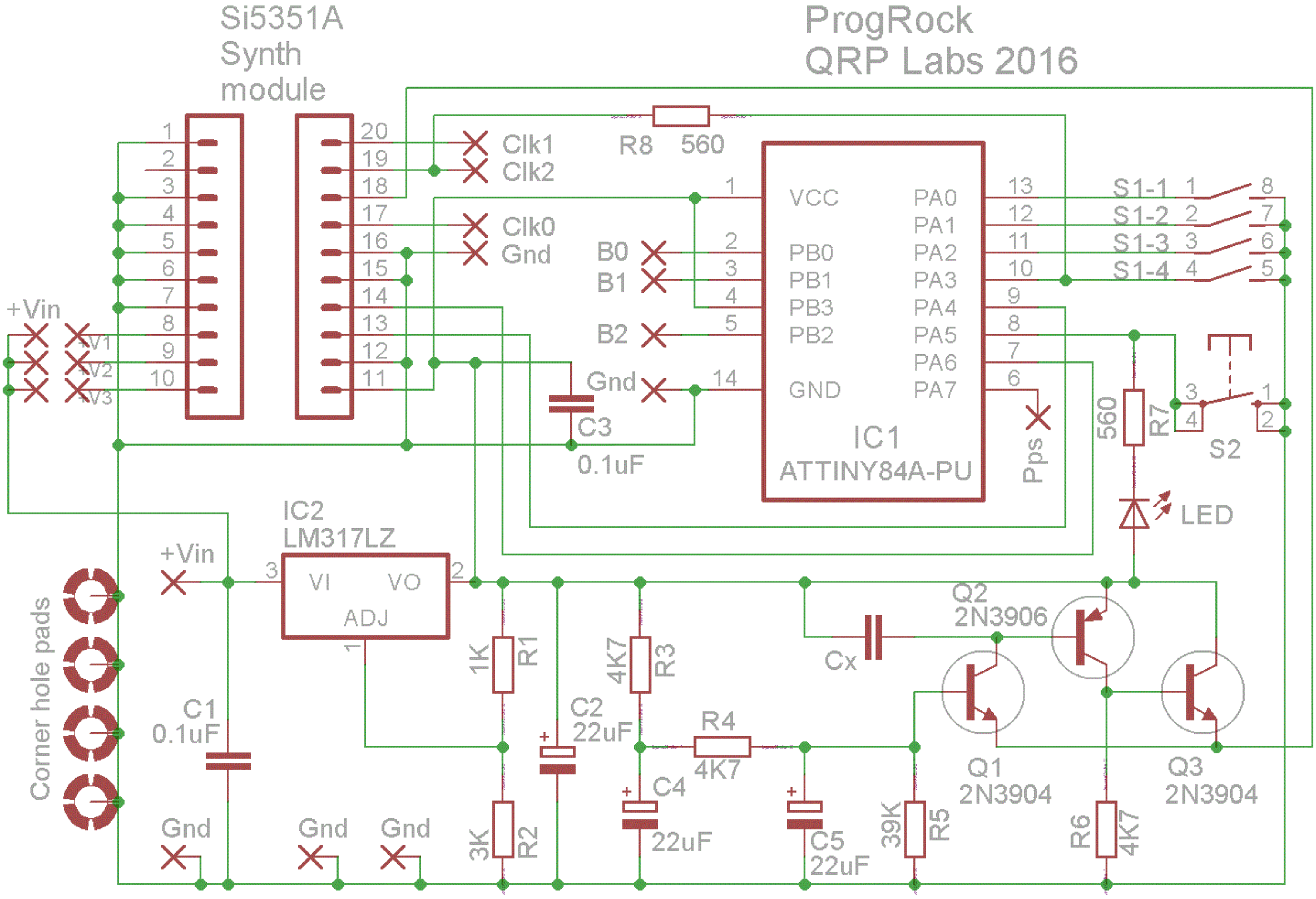
3pcs 22uF

1pcs 2N3906

3mm red LED



2pcs 2N3904



3. Design

Please refer to the circuit diagram on the previous page. We can divide this diagram neatly into 4 pieces.

At top left, are the two 10-pin header sockets for the Si5351A Synth module kit. On the actual PCB, many of the other components such as the microcontroller IC1, sit underneath the Si5351A Synth module, to save space. The Si5351A chip uses a 27MHz reference crystal, multiplied up by a phase-locked VCO to 600-900MHz, then divided back down to produce the output. There are two PLLs on-board and three output division stages. Since the VCO feedback and division stages have fractional divisors, the output frequencies can be set with high precision. The Si5351A chip is controlled by commands over an I2C bus. More description of this Si5351A Synth module kit is provided in the associated assembly manual for that kit.

At top right, is the microcontroller IC1, an ATtiny84, 8-bit Atmel AVR processor. This is the smallest (physically, and logically i.e. memory space etc) AVR processor that would do the job in this ProgRock application. The internal 8MHz RC oscillator is used, as timing is non-critical. The microcontroller is pre-programmed by QRP Labs with firmware to control the Si5351A. The operator sets up the frequencies using the 4-way DIP switch to select a single binary-coded-decimal digit, then press the button briefly to enter it into the microcontroller. The LED provides a visual feedback of the button press. Frequencies are stored in the non-volatile EEPROM memory of the microcontroller, so that they are preserved when the power is switched off. The operating instructions below discuss the set-up procedure in more detail.

At bottom left, is the LM317LZ voltage regulator, IC2. In the recommended assembly approach, this component is taken from the Si5351A Synth kit, it is NOT installed on the Si5351A Synth PCB. Instead it is used on the ProgRock PCB as a 5V regulator. Please see section below for further discussion on power supply options. It is important to decide which option you will use, BEFORE you start assembly of the kit.

At bottom right, is Alan Gray G8LCO's wonderful noise filter circuit. This circuit goes beyond the popular 3-component (resistor-capacitor-transistor) capacitor-multiplier technique of active filtering for power supplies, and provides higher performance: better filtering and better voltage regulation.

In Alan's own words:

When ProgRock was started Hans had the wish to minimise the phase noise due to noisy supply lines modulating the synthesiser. It is fairly well known that some IC regulators can be noisy at LF and that some batches are better than others.

So ProgRock has an IC regulator to provide voltage regulation and current limit protection and an additional circuit to further smooth the supply to the synthesis IC. The regulator output goes to a potential divider which then feeds the base of Q1. Q1 then drives Q2 and Q3, the ring-of-three transistors acts as a unity gain voltage follower with a high input impedance and a low output impedance. The top resistor in the potential divider is split into two resistors so that we can use two filtering caps to form a low pass filter to reduce noise to the input of the voltage follower which then becomes the output. Filtering starts at a few Hz, -60dB around 80Hz and increasing attenuation at 12 dB per octave.

Note that the capacitor Cx is NOT fitted (or supplied). The place on the board is provided just in case there is found to be some performance advantage of including this capacitor.

4. Construction

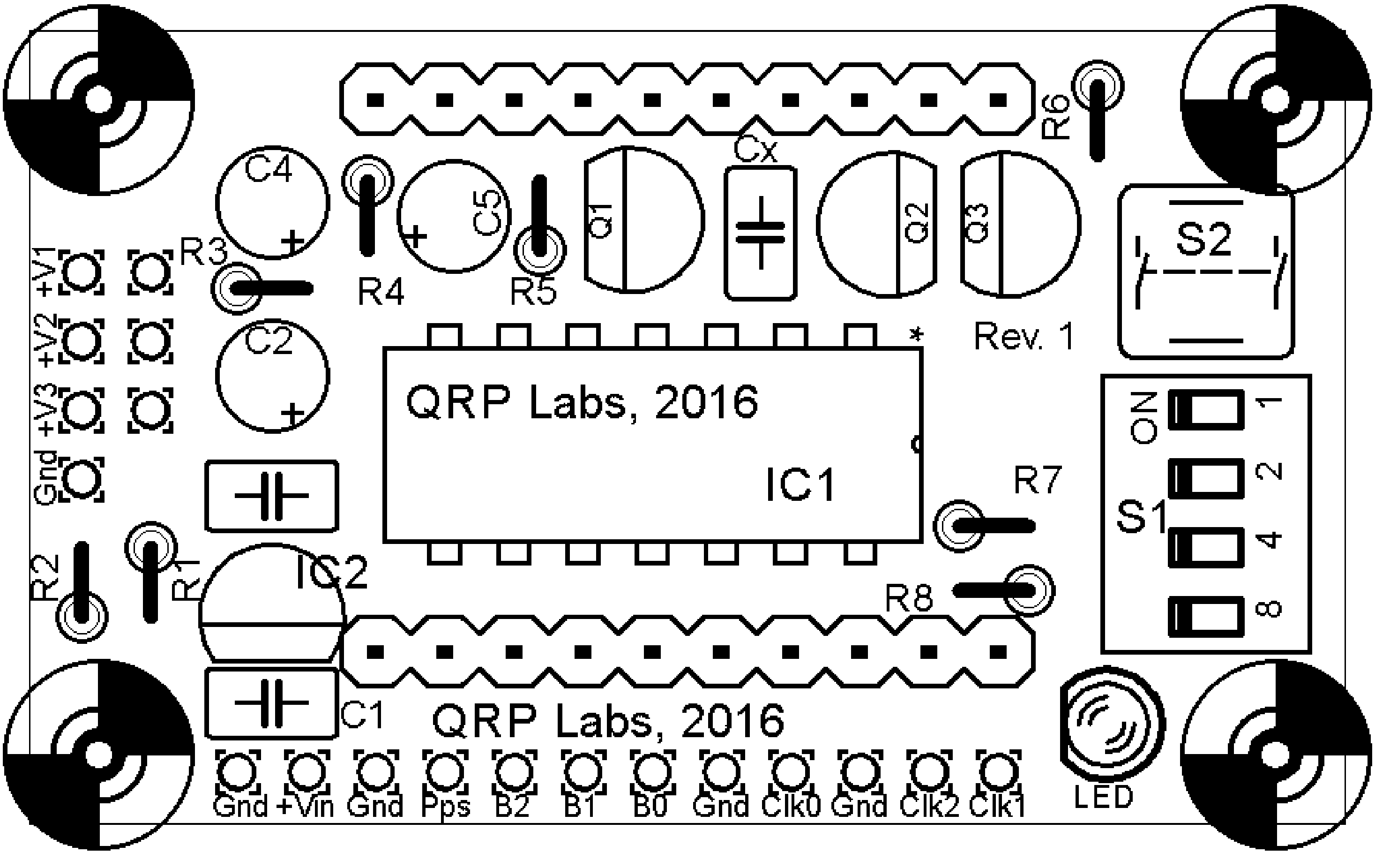
4.1 General construction tips

Parts placement is defined by the silkscreen printed legend on the PCB, so please observe it carefully, paying particular attention to the correct orientation of the semiconductors.

The PCB is quite small and the parts are close together. You are recommended to use a low wattage iron with a fine tip, and fine solder e.g. 1mm diameter or less. Take care not to overheat the PCB and risk damaging it. A well-lit area and magnifying glass are useful. Be careful not to bridge solder across closely packed connections. Some of the joints are close to each other. I recommend checking with a DVM to make sure no solder bridges have been created.

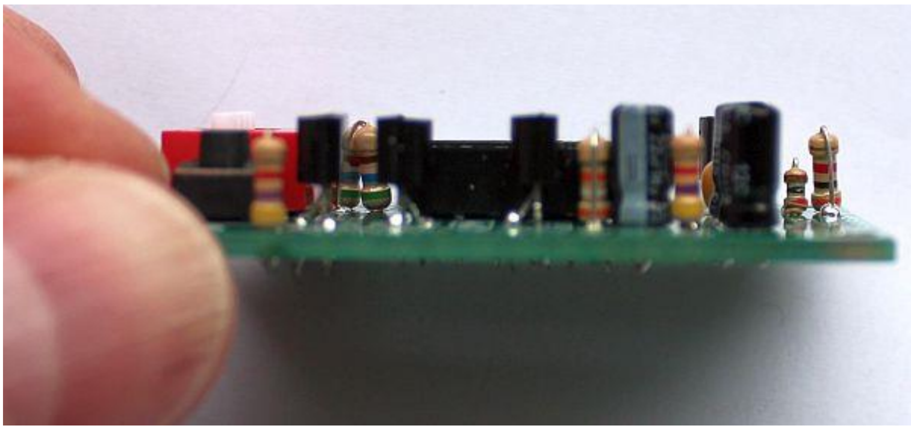
4.2 Parts placement and general construction notes

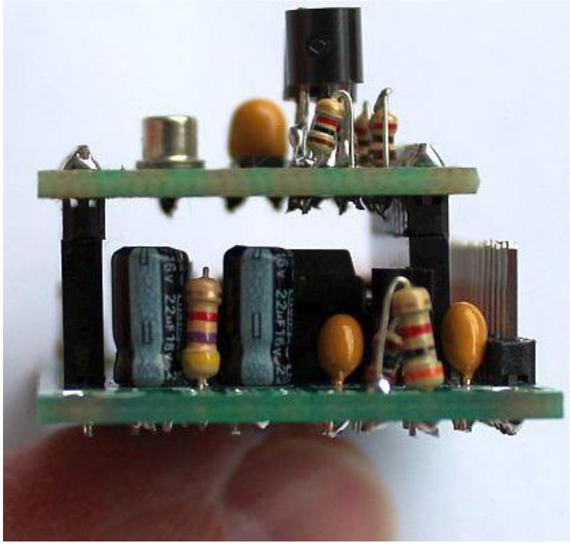
Please refer to the parts placement diagram below, in all the following steps.



When assembling ProgRock, it is important to ensure that the components are pushed all the way to sit nearly on the PCB. This will ensure that the height of the components over the board is not too much: remember that the Si5351A Synth module will be plugged into ProgRock. Many of the components installed on the ProgRock PCB reside underneath the Si5351A Synth module PCB. There needs to be enough clearance to ensure no short-circuits with component leads on the underside of the Synth PCB.

These two photographs will give you an idea of how the finished assembly will look, and fit together. You should keep this in mind while building the kits!





4.3 Power supply options

Originally, the Si5351A Synth module kit has an LM317LZ voltage regulator on-board, that produces the 3.3V required by the Si5351A chip from a 5V supply. There are also a pair of BS170 transistors which act as level converters between the 3.3V Si5351A I2C bus and the microcontroller's 5V data lines, where the module is used with 5V systems.

The ProgRock is designed to run the microcontroller at 5V. The recommended assembly is to leave out the LM317LZ and associated components from the Si5351A Synth PCB, and instead use them on the ProgRock PCB as a 5V regulator. The 3.3V required by the Si5351A is then produced by the noise filter circuit (actually it produces about 3.5V, which is within the Si5351A's specification). The advantage of this approach is that the noise filter produces an exceptionally clean supply for the Si5351A. Furthermore, it means ProgRock can be used with higher supply voltages, in the range 7 to 37V. Practically speaking, the LM317LZ will need to supply around 30mA current, mostly to the Si5351A; if you use the LM317LZ to drop the voltage from 37V to 5V the power dissipation will be almost 1W which will be too high for this tiny TO92-packaged voltage regulator. So limit the supply voltage to 12V, in practice.

You may wish to power ProgRock using a 5V supply. In this case you would short out the LM317LZ with a wire jumper.

ProgRock is also compatible with the QRP Labs OCXO/Si5351A Synth module kit. This is a kit-built 27MHz Oven Controlled Crystal Oscillator reference version, of the Si5351A Synth kit. The heater circuit for the oven consumes quite a lot of power (too much for an LM317LZ), so in this case you should wire ProgRock for 5V operation, and build the OCXO kit exactly as described in its assembly instructions!

The following lists the various power supply options you may want to consider.

4.3.1 5V LM317LZ regulator on ProgRock, noise filter provides 3.3V to Si5351A (RECOMMENDED)

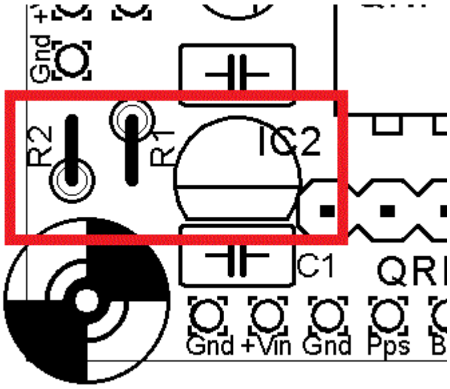
This is the recommended method for powering ProgRock and the Si5351A Synth module kit!

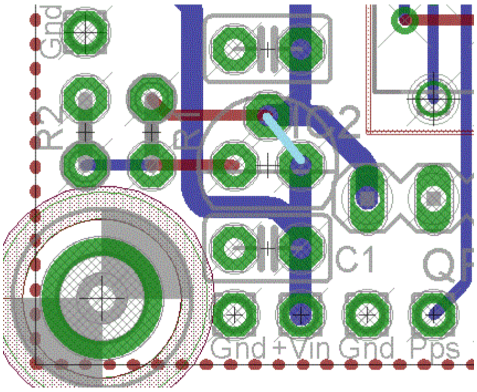
Here you should leave off the LM317LZ voltage regulator, two resistors and 0.1uF capacitor that produce the 3.3V on the Si5351A PCB. Instead, 3.3V will be provided by the noise filter circuit on the ProgRock PCB. In the Si5351A kit, the components to leave off, are labelled IC2, R5, R6 and C1. Some of these components are then used in the assembly of the ProgRock PCB.

The rest of this assembly guide assumes that this is the option you choose.

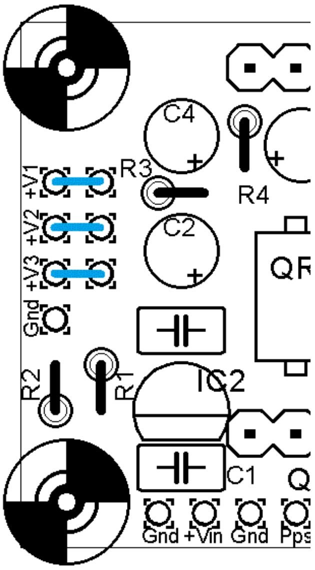
4.3.2 Power ProgRock with 5V, and use the noise filter for 3.3V to Si5351A

Use this option if you want to power ProgRock with a 5V supply. In this case, leave off the 3.3V regulator circuit from the Si5351A Synth PCB, as described in section 4.3.1. But do not install the voltage regulator on the ProgRock PCB. Components R1, R2 and IC2 are not installed on the ProgRock PCB. These are the components in the red rectangle below-left. The vacant LM317LZ should be jumpered as shown by the pale blue line in the diagram below-right, so that the 5V supply reaches the rest of the ProgRock circuit.





4.3.3 Power ProgRock with 5V, and use the OCXO/Si5351A Synth kit



When using the OCXO/Si5351A Synth kit for very high frequency stability, you should always power it with 5V because the oven heater circuit is designed to work at 5V. In this case, build the OCXO/Si5351A kit exactly as described in its manual. Omit R1, R2, C1 and IC2 from the ProgRock PCB. Jumper IC2, as described in the above section 4.3.2. Also omit all the components associated with the noise filter circuit, since the OCXO/Si5351A Synth kit has its own 3.3V regulators onboard, which will supply its Si5351A chip.

Additionally, three jumper wires should be used feed the 5V supply to the OCXO/Si5351A's 5V supply pins. The three jumper wires are shown in blue (right), and the pads on the PCB are labelled +V1, +V2 and +V3.

4.3.4 Other supply options

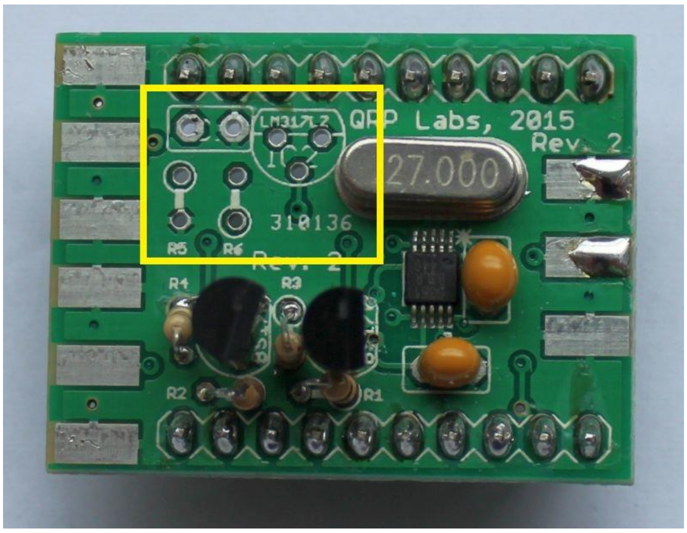
There are other combinations of power regulator options that can be considered. For example you could actually use the noise filter for the Si5351A power supply in the OCXO/Si5351A Synth kit, by modifying that kit. Provision is made to jumper each of the three 5V pins to the OCXO/Si5351A Synth kit separately so that if you wish, you can power them with separate 5V supplies (for absolutely the best frequency stability this is a good idea).

All these options are advanced topics, to be attempted only by the experienced constructor after studying and understanding all the circuit diagrams etc.

4.4 Assembly

The order of construction is not very important. However, a good principle to follow is to install the smaller components first, so that the larger ones do not prevent easy access. One suggested order of construction is described below. I recommend following it carefully.

1) **Build the SI5351A Synth module kit first**

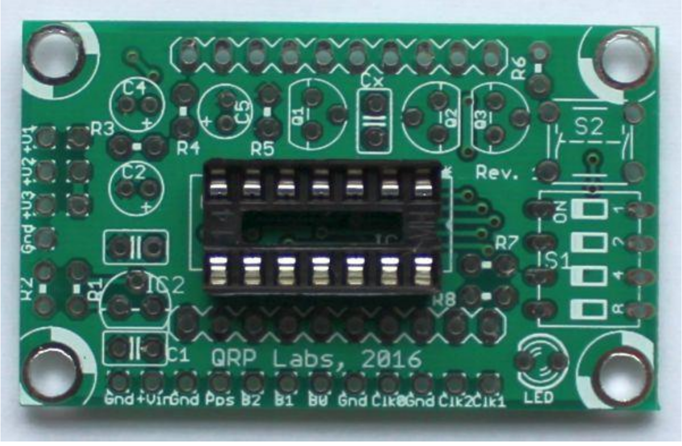


Re-read section 4.3 (power supply options) and make sure you have decided which way to build the kits.

This example assumes you will be installing the LM317LZ on the ProgRock PCB not the Si5351A Synth PCB. The components in the yellow square have not been fitted – C1, R5, R6 and IC2 in the Si5351A Synth kit numeration.

Please ignore the solder blobs on the right-hand SMA socket pads – just my testing!

2) **Solder IC1 socket**

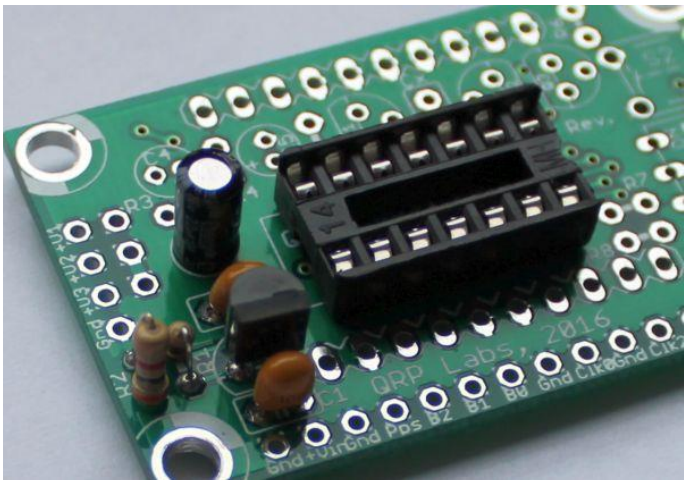


I always fit the IC1 socket first, the 14-pin DIP socket. The pins are quite delicate and may have become bent – just straighten them out, to fit them in the PCB holes.

To avoid confusion or mistakes later, align the dimple at one end of the socket, with the dimple illustrated on the PCB. The dimple should be at the end nearest the right-hand edge of the PCB.

3) **Install voltage regulator components**

Here again, I assume that you are using the recommended power supply option, that is, fitting the LM317LZ voltage regulator on the ProgRock PCB, and using it as a 5V regulator.



Fit components R1, R2, IC2, C1, C2 and C3.

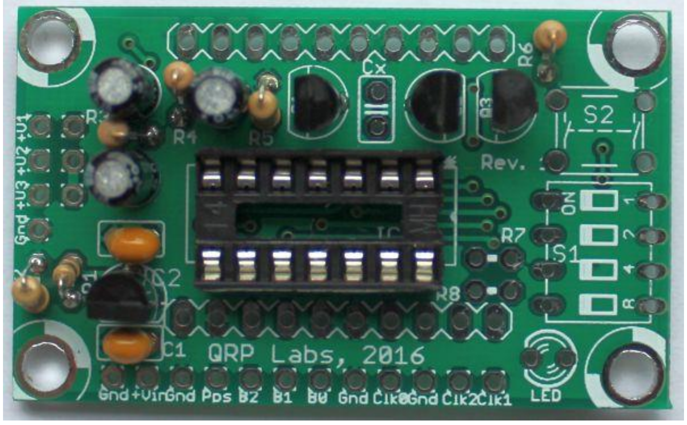
Note that R1 (1K resistor), C1 (0.1uF ceramic capacitor), and IC2 (LM317LZ voltage regulator) are to be taken from the components supplied with the Si5351A Synth kit. It will be necessary to bend the central lead of the LM317LZ to fit the triangular pad placement on the board. Align the shape of the LM317LZ body to the shape drawn on the PCB silk screen.

Don’t mix up the 1K and 3K resistors! R2 (3K, colour code orange-black-red-gold) is the one right next to the board edge.

For some reason the silk screen legend for C3 seems to have been missed. C3 is the ceramic capacitor (0.1uF) right above IC2 in the diagram; between IC2 and C2.

When fitting C2 (22uF electrolytic) capacitor, PAY ATTENTION to the polarisation of this capacitor! There is a white stripe along one side of the capacitor, this is the negative wire. The positive wire is indicated by a "+" symbol on the PCB silk screen. Don't get the capacitor the wrong way around!

4) **Install noise filter circuit components**

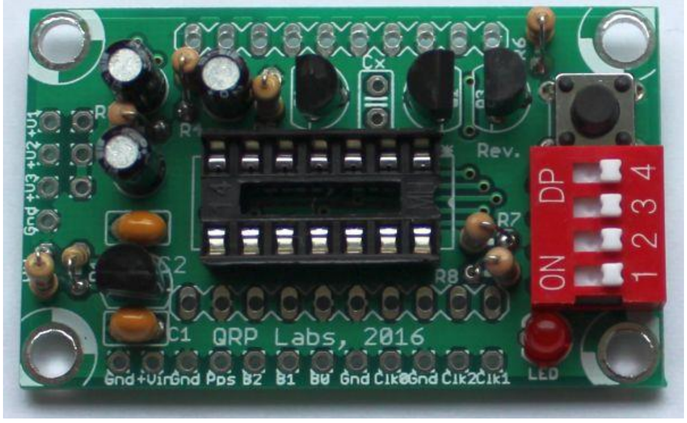


Install components R3, C4, R4, C5, R5, Q1, Q2, Q3 and R6. These components are in a line along the top of the board parts layout diagram. For convenience I recommend working from left to right. These components are all part of the noise filter circuit. This circuit takes 5V input and produces about 3.5V output for the Si5351A chip, resulting in a very highly filtered low-noise supply.

Pay careful attention to the polarity of C4 and C5 (22uF capacitors). The positive lead is indicated by a "+" symbol on the PCB silk screen, and the negative lead on the capacitor is indicated by a white stripe.

When installing Q1, Q2 and Q3, be sure to correctly identify which are the 2N3904 (NPN) and 2N3906 (PNP) transistors! They look the same – so carefully check the writing on the side of the transistor. The 2N3906 PNP transistor is fitted in the Q2 position, the central one of the line of three transistors. All three of the transistors will need their central wire bending outward to fit the triangular pad pattern on the PCB. Make sure the transistors body shapes are aligned with the shapes printed on the PCB silk screen. Note that Q2 (the 2N3906) is "backwards" compared to Q1 and Q3.

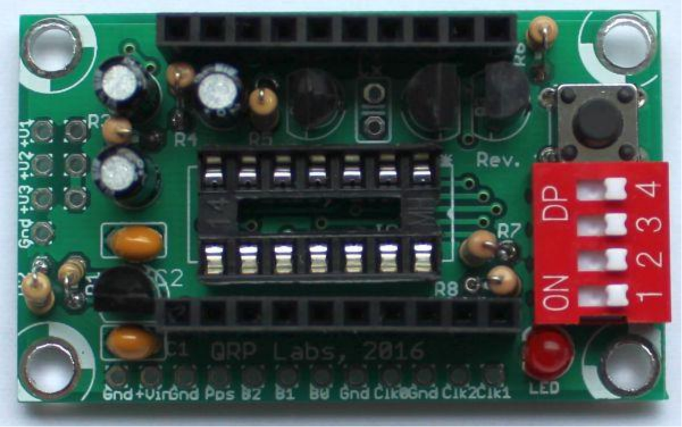
5) **Fit switches, button, LED and resistors**



Fit the switches S1 (4-way red-coloured DIP switch), the push-button S2, 3mm red LED, and remaining resistors R7 and R8 (both 560-ohm).

Make sure the "ON" printed on the 4-way DIP switch is aligned as shown in the photograph (right). Do not worry about the fact that the PCB silk-screen printing identifies the switches as "8 4 2 1" whereas they are labelled "1 2 3 4" on the switch body itself; and "ON" is in a different place on the silkscreen compared to the actual switch!

6) **Fit two 10-way header sockets**

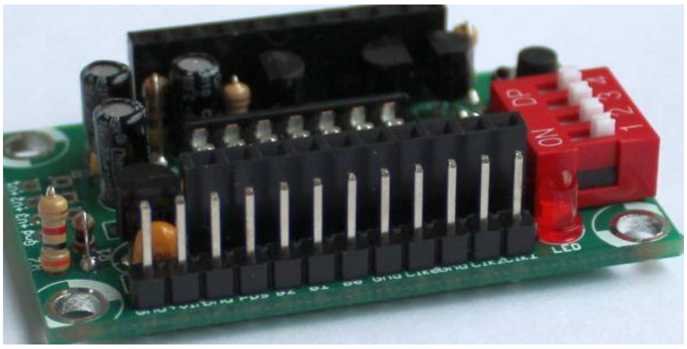


Here the two 10-way header sockets should be fitted and soldered. They are a little loose in the holes so it is important to align them well so that the Si5351A Synth module kit will plug in cleanly without stress. I find that the best way to do this, since you have already assembled the Si5351A Synth module, is to plug the Synth module into the sockets, then fit the sockets into

...

the ProgRock PCB, then solder them. This way, the Synth module will hold the sockets in perfect alignment while you solder them.

7) Fit a 12-way pin header



The ProgRock kit includes a 16-pin header. You can snap away 4 pins, to create a 12-pin header, and install this in the PCB's external connection pad holes, along the bottom edge of the board.

This is OPTIONAL. You might find it easier and prefer to solder your wires directly into these holes. But some people prefer the header pins, so that connectors can be plugged on. Some people prefer to solder wires directly to the header pins, which are easier to repair if wires break, than cleaning out and re-using holes on the PCB. It's up to you!

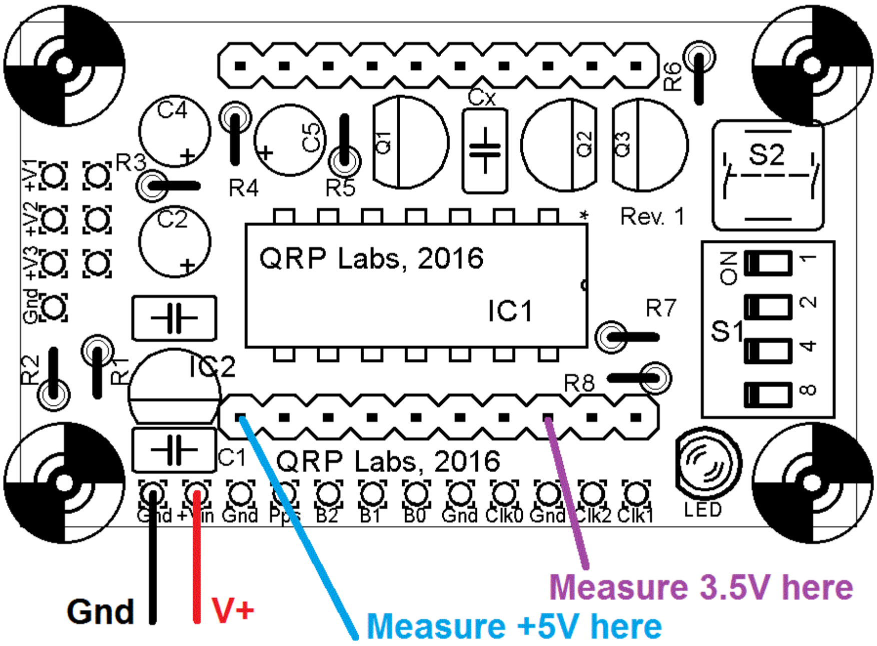
The photograph (above right) shows the 12-pin header installed on ProgRock.

8) Test the voltage regulator and noise filter circuits

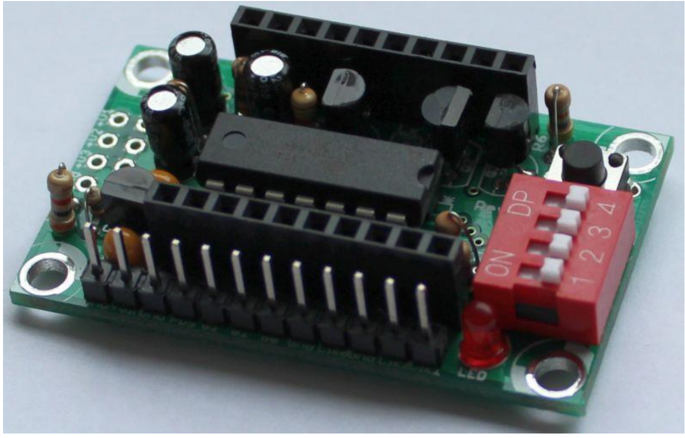
Again assuming you are using the recommended power supply option (refer to section 4.3), it is a REALLY good idea to test the voltages now with a digital volt meter (DVM) before plugging in the microcontroller IC1, or the Si5351A Synth module. IC1 will be damaged if the supply voltage exceeds 5V and the Si5351A requires 3.3V supply (3.5V as supplied here, is fine). It's best to make sure the voltages are correct now, before frying any of these ICs!

To test the voltage regulator and noise filter, apply the supply voltage to the Gnd and +Vin pins. I recommend a voltage in the range 7-12V. You should then be able to measure +5V and +3.5V at the points indicated in the diagram below. You could use the spare 4-pin header section, inserted into the 10-way header socket, to provide a pin to touch with the DVM probe for the measurement. Measure the voltages between the indicated points, and ground. They should measure around 5V and 3.5V respectively (within 5%).

If there is any problem with the measured voltages then you have GOT to investigate and find the problem before going any further!



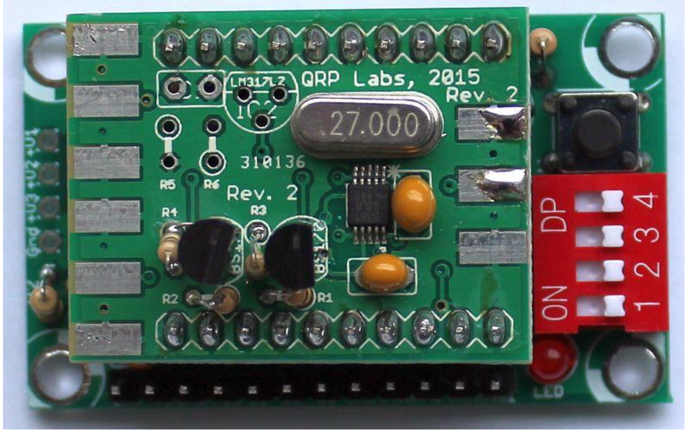
9) **Insert IC1, the ATtiny84 microcontroller**



It is normal for the IC pins to be angled slightly too widely. You need to squeeze the pins together in order to make them fit in the 14-pin DIP socket.

Take great care to ensure that IC1 is plugged in the correct way around. The dimple on the chip should match the dimple on the 14-pin DIP socket, and the dimple on the PCB silk screen. All the dimples should be closest to the right-hand side of the PCB (when held as photographed throughout this document).

10) **Plug in the Si5351A Synth module**

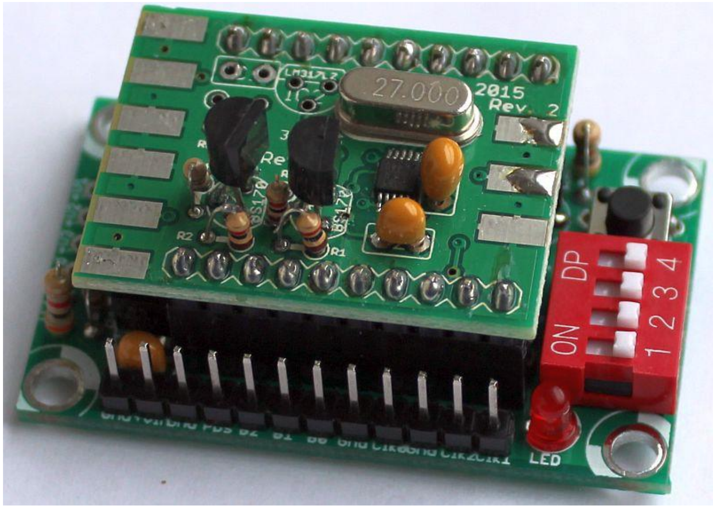


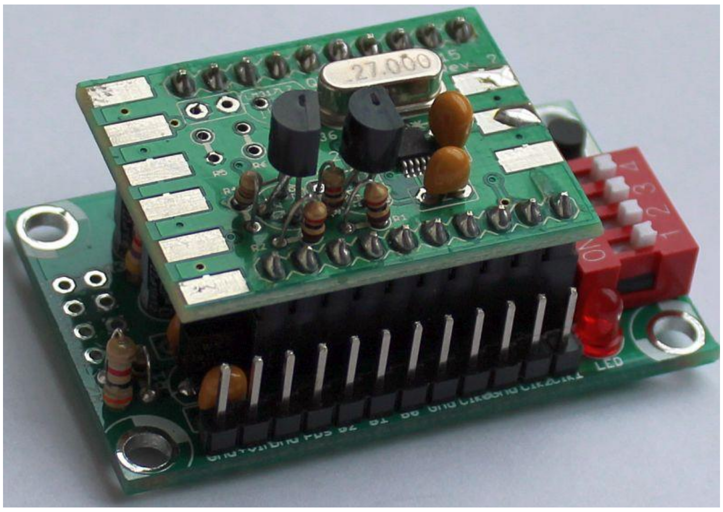
Finally, the Si5351A Synth module can be plugged in to the ProgRock PCB.

Ensure that the orientation of the Si5351A Synth module is correct! It should look like this photograph (right).

11) **Assembly is finished!**

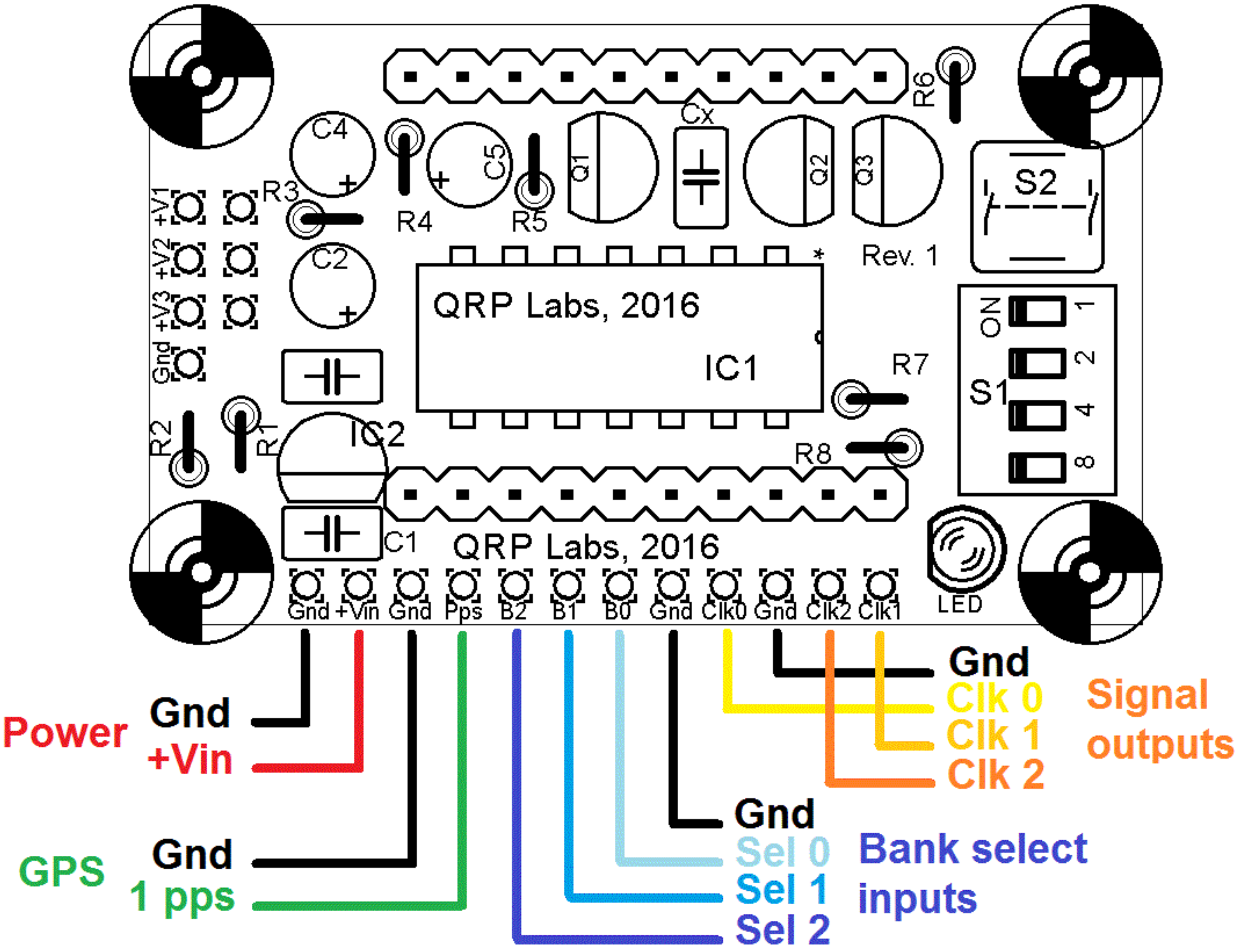
Now the assembly of ProgRock and the Si5351A Synth module is complete! The assembly should look like these photographs.





5. Connections

All the connections to ProgRock are made at the row of 12 pads along the bottom edge of the PCB. There are four groups of connections to discuss: Power, GPS, Bank select, and signal outputs.



5.1 Power connections

Connect a well-smoothed, clean DC power supply. Refer to your chosen power supply option (see section 4.3). For the recommended assembly with the LM317LZ voltage regulator acting as 5V regulator on the ProgRock board, the input supply voltage should be in the range 7-12V.

5.2 GPS connection

If you connect a positive-going, 1 pulse-per-second (pps) signal from a GPS receiver such as the QRP Labs QLG1 GPS receiver kit, then ProgRock will automatically use this to discipline the Clk0 and Clk1 outputs. **IMPORTANT: When not in use, connect the 1pps input to Ground! Receivers with a very short pulse e.g. 10us will not work (for example, Trimble Thunderbolt).**

The Si5351A Synth module’s Clk2 output is used by the microcontroller to count the frequency of the Si5351A’s 27MHz reference crystal. The processor makes corrections to the configuration of the Si5351A synthesiser to compensate for any error or drift in the 27MHz crystal oscillator. It takes a few minutes to get the ProgRock output frequencies accurate within 0.1ppm. Calibration is faster on subsequent uses.

When using GPS discipline, the Clk2 output is NOT available for normal use. The configurations which are entered for the Clk2 output are simply ignored. To use Clk2 again, you must disconnect the GPS 1pps signal, and switch off/on the power supply to ProgRock to cause a system reset.

5.3 Bank select inputs

ProgRock contains stored frequencies for 8 banks of frequencies, with 3 output frequencies in each bank. If you wish to use this feature, you should select the bank you want to use, using the three input signals Sel 0, Sel 1 and Sel 2. Immediately the setting on these three input signals changes, ProgRock will reprogram the Si5351A to the frequencies stored in its non-volatile EEPROM memory.

By default, if you leave the signals disconnected, Bank 0 is used.

These signals have the internal pullup activated within the microcontroller. The signals are active-low. “Low” voltage means 0V (ground), “High voltage” means 5V – or just disconnect completely (since the internal pull-ups will make the pin high).

The mapping of input signals to bank selected, is ordinary binary. For clarity, the mapping is listed in the table below.

Sel 2	**Sel 1**	**Sel 0**	**Bank**
High	High	High	0
High	High	Low	1
High	Low	High	2
High	Low	Low	3
Low	High	High	4
Low	High	Low	5
Low	Low	High	6
Low	Low	Low	7

The frequency banks could be used for many purposes. You may wish to program different “crystal” frequencies for different bands in your radio, for example; or for upper/lower sideband. You could also use one of the Bank inputs as a “key” signal to enable the frequency output. For example imagine programming the frequencies you want, in Bank 0. Then use the “Sel 0” input as a “keying” signal. Assuming you have left all three Bank 1 frequencies as 0Hz, the effect will be that when Sel 0 is taken Low, the frequency outputs are all disabled. Of course, you could reverse Bank 1 and Bank 0 in this example, if you want the Sel 0 “keying” input to be active high instead.

ProgRock allows you a LOT of flexibility!

5.4 Signal outputs

The raw outputs of the Si5351A Synth IC in the synth module kit are available in two places. Firstly, there are pads on the Si5351A Synth kit where you may solder edge-type SMA connectors if you wish. Secondly, if you prefer, there are also output pins on the connector, as shown previously in the connection diagram.

The outputs are 3.3V peak-peak squarewaves, with 50-ohm impedance. The maximum drive current setting in the Si5351A register configuration is used.

Note that if the GPS discipline option is used, by connecting a GPS 1pps timing signal to the 1pps input pin, then the Clk2 output will be used by the microcontroller to measure the 27MHz reference crystal oscillator frequency, to permit calibration and drift correction. In this case you should not use or connect anything to the Clk2 output.

5.5 Limitations of the Clk2 output

Read this section very carefully if you intend to use ProgRock's Clk2 output!

The Si5351A chip contains two PLL circuits and three output "MultiSynth" divider circuits. The feedback loop of the PLLs are fractional dividers, and all three output dividers are also fractional. For best performance (low jitter) the Si5351A datasheet recommends even integers in the "MultiSynth" dividers even though they are capable of fractional use. However if the PLL A and PLL B are set up for the Clk0 and Clk1 outputs, the only way to permit a general frequency on the Clk2 output, is to use fractional division in the MultiSynth for Clk2. There are also other limitations on the use of Clk2 because it is used for calibration when the GPS is connected.

Please take into account the following when considering the use of the Clk2 output:

1) The Clk2 output is fed via a resistor into the T0 timer input of the microcontroller. However this pin is also shared with one of the DIP switches S1. When the push-button S2 is pressed, Clk2 is temporarily disabled so that the state of the DIP switch can be read without interference.

2) When using GPS to calibrate and discipline ProgRock, Clk2 cannot be used. A single 1pps signal pulse on the 1pps input is enough to switch into GPS discipline mode. Clk2 is then used by the microcontroller for measuring the frequency of the 27MHz reference oscillator. You must switch off the power and back on again, to get back the use of Clk2. It is strongly recommended to Ground the 1pps input when not using the GPS option, to prevent accidental triggering of GPS mode.

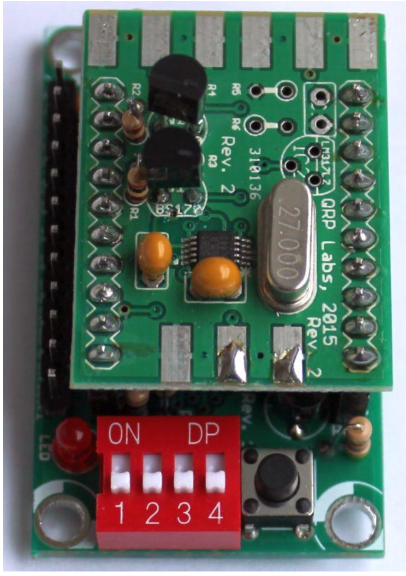
3) Inside the Si5351A, Clk0 is mapped to PLL A. Clk1 is mapped to PLL B. Therefore when you wish to use Clk2 you MUST ensure that there is a valid frequency set up for Clk1!

4) There are some frequency limitations imposed by this arrangement. PLL B operates internally in the Si5351A chip at a frequency of nominally 600-900MHz. The exact PLL VCO frequency will have been determined by the algorithm that sets up the Si5351A registers for the specified Clk1 output frequency. For a Clk2 output, the microcontroller specifies a division ratio to divide PLL B's output down by a fractional amount to get to the desired Clk2 output frequency. The division ratio must be in the range 8..900. If it is outside this range, Clk2 will be switched off by the microcontroller. Practically speaking, this limits the frequency of Clk2 to the range (approximately) 1MHz to 112MHz.

6. Programming

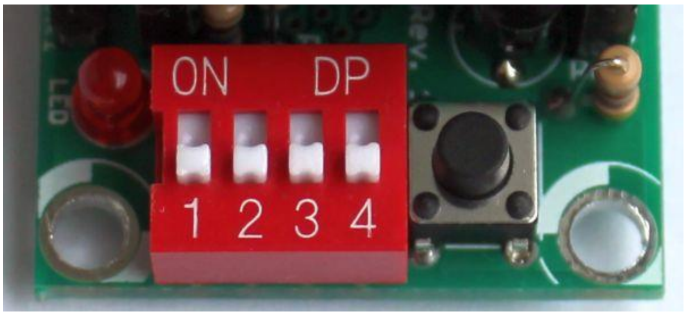
6.1 User Interface

The ProgRock frequencies list is programed in Binary Coded Decimal, which is input on the 4-way DIP switch, and button. Each time you press the button briefly, the 3mm red LED will illuminate for a short time. This will give you positive feedback that your button press actually occurred and the number programmed on the DIP switch has been accepted by the microcontroller. You cannot program another digit until the red LED goes off. These basic



controls (DIP switch, button, and LED) are your whole User Interface to ProgRock!

You will need to move the DIP switches with a screwdriver blade-tip – they are too small to set with your fingers!



The rest of this document assumes holding the ProgRock PCB for programming purposes in the orientation shown in these photographs.

When programming ProgRock please remove the 1pps connection to the GPS, if you are using the GPS. It can interfere with the button debounce logic. The 1pps input should be grounded during programming of ProgRock.

The DIP switches should always be set to OFF at Power-up and when you have finished programming! ProgRock cannot test the Si5351A (see start-up procedure below), or do GPS calibration, when the DIP switch on the left (labelled "1" in the above photograph), is ON!

6.2 Power-up procedures

On applying power to ProgRock, the red LED is illuminated for a short time.

The first operation undertaken by the ProgRock microcontroller on power up, is to check the connection to the Si5351A chip via the I2C bus. It then checks that it can read the 27MHz reference frequency (divided down) on the Clk2 output. If the I2C communication fails or the reference frequency cannot be read, then the red LED will remain illuminated. No further actions will be taken, and the switches, buttons, and bank select inputs are all ignored. If the microcontroller cannot operate the Si5351A chip, then nothing else can be done. You will need to investigate the issue and resolve it, before proceeding. The system must be reset by removing and re-applying power, to try again. **The DIP switches must be set to OFF at power-up and when using the GPS calibration option!**

You might wish to program the ProgRock frequency list without the Si5351A Synth module plugged in. For example if you have fitted SMA connectors to the Si5351A Synth module, then the 4-way DIP switch and button will be a bit hard to access with your fingers. When the Si5351A Synth module is not plugged in, the above error check will fail, and ProgRock will be stuck in the "error state" with the red LED illuminated.

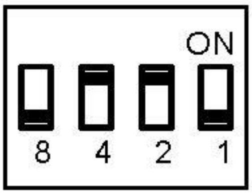
To avoid this problem, and allow you to program the frequency list without the Si5351A Synth module plugged in, you can BYPASS the error check. To bypass the error check, press the button (S2) while powering up ProgRock. When you have done this, the microcontroller will not attempt to communicate with the Si5351A chip again, until power is removed and re-applied.

Finally if all is well, you have programmed frequencies, and the Si5351A Synth module is all working properly – the microcontroller will set up the Si5351A chip to output the frequencies you chose.

6.3 A crash course in Binary Coded Decimal

You can skim over this section if you already know about Binary Coded Decimal.

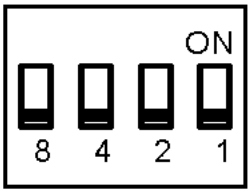
You have 4 switches in the 4-way DIP switch. Each switch can be either on or off. If it is on, the weight is 8, 4, 2 or 1 as indicated by the silk screen printed legend on the ProgRock PCB. Unfortunately the actual red switch is only labelled "1 2 3 4" and you will have to ignore that.



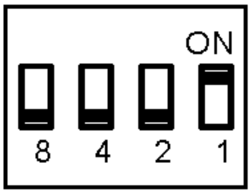
To make a number, you switch the required switches to "on" that add up to the required number.

For example, the switch configuration to (above-right) means "6", because the "4" switch and the "2" switch are the only switches which are "on", and 4 + 2 = 6.

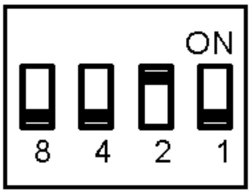
The table below shows the switch configurations for digits 0 to 9, for your reference.



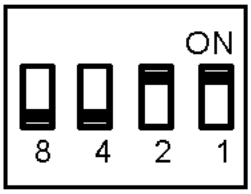
0



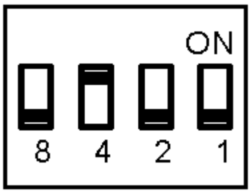
1



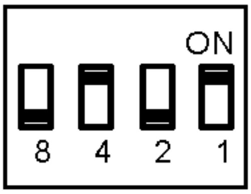
2



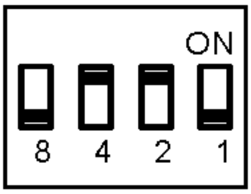
3



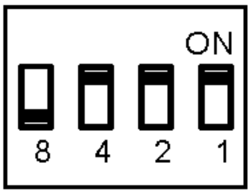
4



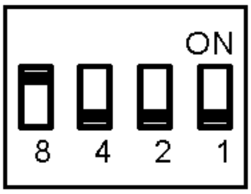
5



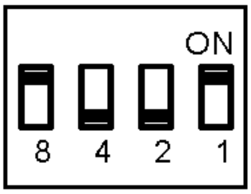
6



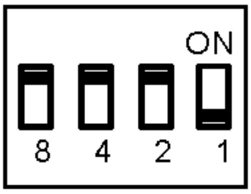
7



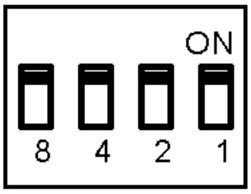
8



9



Cancel



Enter

Note the two special codes, which represent "Cancel" and "Enter". (These are hexadecimal E and F respectively, if you want to consider them this way).

If you configure the switch for the "Cancel" command and press the button, then the current number entry is cancelled.

If you configure the switch for the “Enter” command and press the button, then the current number entry is actioned by the microcontroller – the selected frequency register is programmed with the entered number, stored in EEPROM, and programmed to the Si5351A chip if it is in the active frequency bank.

6.4 ProgRock register configuration

ProgRock contains 29 registers, referred to as Register ID 00 to 28. Each register stores a 32-bit number, which you can program from the switches and button. The table below lists the register mapping. To enter a frequency into a register, you must first specify the register, then the required frequency register. The digits are entered into ProgRock one digit at a time.

You set up a digit on the 4-way DIP switch then press the button briefly. The LED illuminates for a short time. Wait for the LED to extinguish. Then set up the next digit, press the button, and continue until you have completely entered all the digits needed. Finally set up the “Enter” code (all 4 switches “on”) and press the button again. This causes the entered number to be stored at the specified register location.

For example, suppose you wish to set the Clk 0 output in bank 0 to 1.5MHz. The sequence of digits you need to enter is:

0 4 1 5 0 0 0 0 0 Enter

Because “04” is the register location for Clk 0 Bank 0, and “1500000” is the desired frequency, in Hz. The total number of digit set-ups and button presses you will need to make, is 10, in this case.

Make sure you give the button a decisive short (brief) press. Make sure that you wait for the LED to go out, before pressing the button repeatedly! Button presses are ignored while the LED is lit. It’s a kind of very simple switch debounce, as well as an indication of the button press. This is particularly important to remember if you are entering a repeated sequence of digits, such as the zeroes in this example. You don’t need to alter the 4-way DIP switch settings after each digit entry. It is tempting to just keep hitting the button 5 times, to enter the 5 zeroes. But you MUST WAIT after each button press for the red LED to go out, first!

If you try to enter a register value into a register number higher than 28, your entry will simply be ignored.

If you enter an invalid frequency, which is to say, the frequency is outside the possible range of the Si5351A, then the frequency obtained at the output will not be what you want. The most likely output will be none at all.

At the end of this document is a template which you can print and use to help program the frequencies.

The following table shows the register mapping in ProgRock.

Register	**Purpose**
00	Reserved, do not use this register
01	EEPROM version ID. See SECTION 6.5
02	27MHz reference clock value. See SECTION 6.6
03	GPS correction threshold. See SECTION 6.7
04	Bank 0, Clk 0 – default value 10.000MHz
05	Bank 0, Clk 1
06	Bank 0, Clk 2
07	Bank 1, Clk 0
08	Bank 1, Clk 1
09	Bank 1, Clk 2
10	Bank 2, Clk 0
11	Bank 2, Clk 1
12	Bank 2, Clk 2
13	Bank 3, Clk 0
14	Bank 3, Clk 1
15	Bank 3, Clk 2
16	Bank 4, Clk 0
17	Bank 4, Clk 1
18	Bank 4, Clk 2
19	Bank 5, Clk 0
20	Bank 5, Clk 1
21	Bank 5, Clk 2
22	Bank 6, Clk 0
23	Bank 6, Clk 1
24	Bank 6, Clk 2
25	Bank 7, Clk 0
26	Bank 7, Clk 1
27	Bank 7, Clk 2
28	Calibration configuration register. See SECTION 6.8

6.5 Factory Reset using EEPROM Version, register 01

Register 01 contains the EEPROM version ID. If you set this register to 0, and power down/up the system, then the EEPROM contents will be erased and returned to the factory default. The factory default sets all output frequency registers to 0, the Reference clock register 02 to “27,004,000”, GPS Correction Threshold register 03 to “5”, and the Calibration configuration register 28 to its default factory value.

If you mess everything up accidentally, you can always force this factory reset using this method, to return your microcontroller to the default factory-installed state. Key the digit sequence

0 1 0 Enter

Then cycle the power. This will enter value 0 into register 01, and when the system is rebooted, the factory reset will occur.

6.6 27MHz reference clock value, register 02

The 27MHz crystal used in the Si5351A Synth Module crystal does not oscillate at exactly 27.000000MHz. It varies from one crystal to the next, but in general the frequency will be 3-5 kHz too high, e.g. 27,004,000Hz. **This is why the default reference frequency in register 02 is 27,004,000.**

If you are using the OCXO/Si5351A Synth module kit, the frequency of this oscillator tends to be about 2kHz too low, e.g. 26,998,000Hz.

If you are using GPS discipline by applying a 1pps signal from a GPS receiver, the kit will self-calibrate in less than half a minute. This will update register 02 in EEPROM. Next time ProgRock is powered up the calibrated value will be used. Thereafter, the GPS makes corrections to the reference oscillator value in register 02 to compensate for any measured temperature-induced drift in the crystal frequency.

The reference oscillator value is used in the calculation of the Si5351A registers. So it is important that it is accurate, if you need the output frequencies of ProgRock to be accurate.

If you are not using a GPS, you can manually calibrate ProgRock by entering the correct value into register 02.

It is possible that you have an accurate means to measure the 27MHz reference oscillator frequency itself by probing the correct point on the Si5351A module. However most kit builders will not have the necessary equipment, and such a measurement is not easy to make without itself risking disturbing the frequency.

The easiest method is to set the output frequency to something convenient such as 13.500000 MHz, and measure it. Measurement can either be by using an accurately calibrated frequency counter, or by setting up an accurately calibrated receiver with Argo and monitoring the output signal frequency that way.

Once you have measured the actual output frequency, you can calculate the required correction to the 27MHz reference frequency and enter it in the “Ref. Frq.” configuration setting. For example, suppose you set the output frequency to 10.000000 MHz but you actually measure 10.000075. Your output frequency is 75 Hz too high. Assuming the default reference frequency (27,004,000), you should change this to 27,004,000 * 10.000075 / 10.000000. So enter the new value 27,004,203 in the “Ref. Frq.” setting. Be sure to write down somewhere the value you enter. If you need to re-calibrate in future, you will need to know what value you entered! Otherwise it will be necessary to do a “factory reset” (see section 6.5).

6.7 GPS correction threshold, register 03

This register has a default value of 5Hz. But you can set it to anything you like. It controls how and when the microcontroller makes updates to the Si5351A output frequencies. For example, at the default setting of 5Hz, the microcontroller will only correct the Si5351A output frequencies when the 27MHz reference value has changed by more than the 5Hz threshold. The threshold is always referred to the 27MHz reference frequency, not the output frequencies.

If you set the register to have a value of 0, then the microcontroller will adjust the Si5351A output frequencies once per second, every second, depending on whether it thinks the 27MHz frequency has drifted upwards or downwards. This continuous change in output frequency may not be desirable. It is more practical to have small jumps in output frequencies, less often – i.e. only when the reference oscillator has changed by more than a threshold.

The QRP Labs website has some notes on how to obtain excellent frequency stability with the Si5351A Synth module – please refer to the resources section below.

6.8 Calibration configuration register, register 28

The value in this register is several smaller numbers, bit-packed into the available 32-bit number. These parameters control the way the GPS calibration and drift correction processes are done.

In the future I may describe these parameters and their effect. It may be possible to further optimise the way the calibration and drift correction processes are carried out.

For the time being, just leave this register alone!

6.9 Default frequency output of ProgRock

At switch-on after factory reset, the Clk0 output is set up to default 10.000MHz. Clk1 and Clk2 are switched off. The reference frequency is set by default to 27.004MHz which should be approximately correct +/- 1kHz. Further calibration will be needed for your crystal as described in section 6.6 above.

7. Template for programming ProgRock

On the following page is a template for programming registers in ProgRock. You could print it out and use it to assist with programming the registers for the frequencies you need.

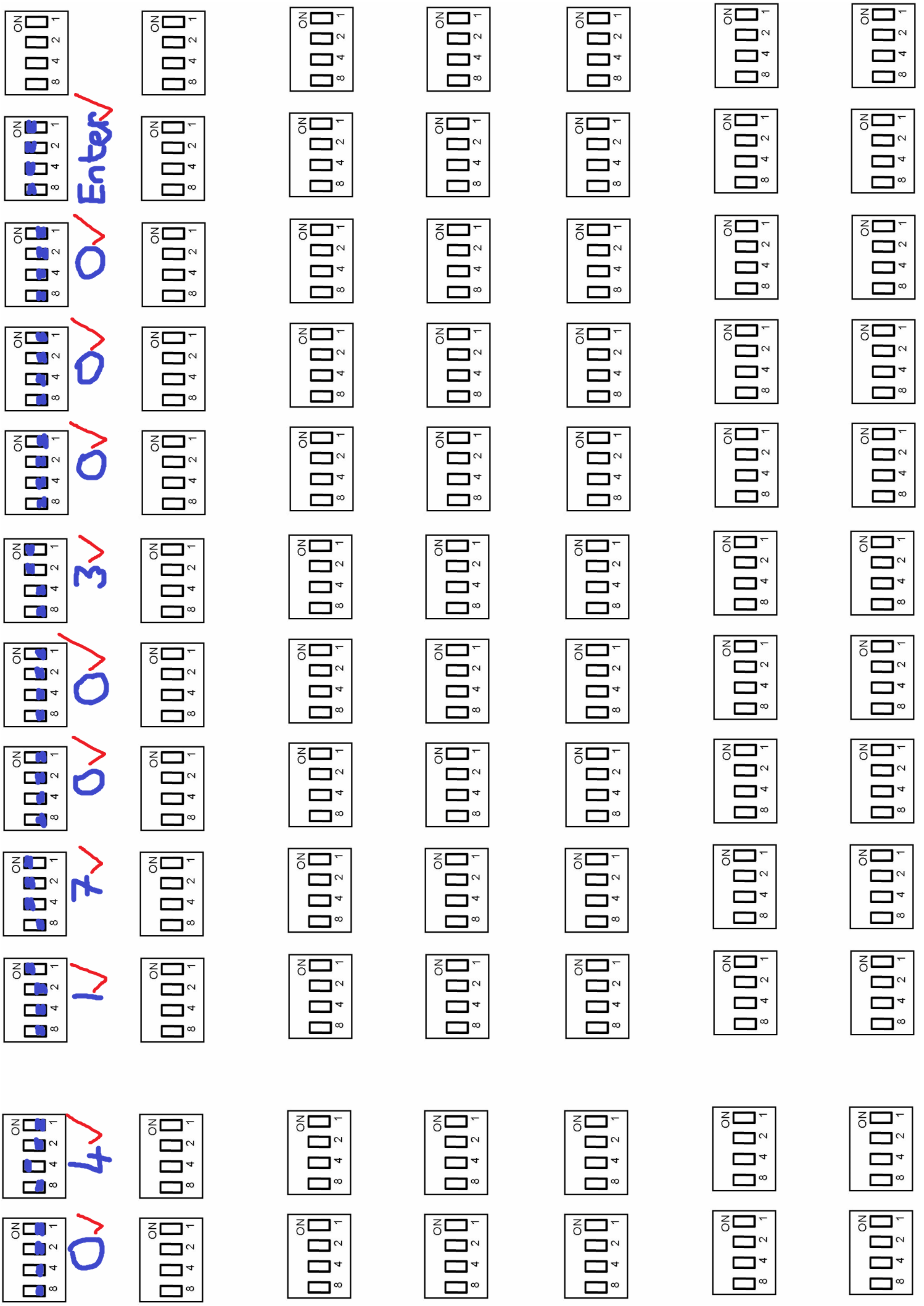
On the top row is an example. Programming the Bank 0, Clk 0 output to 17.003MHz.

The first two digits are the register number. The register number for the Bank 0, Clk 0 frequency is 04 (refer to the table in section 6.4).

The next 8 digits are the frequency, specified in Hz: 17003000.

Finally, the “Enter” code is input (all 4 switches are “on”).

The ProgRock operator can write out the numbers required, as done here on the top line. Then write the switch configurations for each digit. Refer to the Binary Coded Decimal table in section 6.3 if necessary. For each digit, the operator should set the 4 switches of S1 as needed, then press the button briefly (S2). It will help to write a tick next to the digits which have been input, to avoid forgetting where you are, in the number.


ON
1
2
4
8
Enter
0
0
0
3
0
0
7
1
4
0

8. Resources

- Please see the kit pages at http://qrp-labs.com/progrock for information on latest updates and issues.
- Notes on getting excellent frequency stability from the Si5351A Synth kit: http://qrp-labs.com/synth/freqstab.html

9. Version History

1 20-Apr-2016

- First version

2 27-May-2016

- Added Alan G8LCO description of the noise filter circuit
- Added clarification that capacitor Cx is neither required nor supplied, and should not be fitted
- Clarified orientation of the 4-way DIP switch; the silkscreen does not match the component print
- Corrected several typos, missing words, incorrect words etc.

3 24-Jun-2016

- Added note that very short 1pps pulse widths (e.g. 10us) do not work with this kit.

4 25-Jun-2016

- Re-write of the last paragraph of section 6.6 to make the calibration procedure more clear

5 27-Jun-2016

- Section 6.6: emphasise the importance of writing down the reference clock value, in case it is necessary to re-calibrate some time in future

6 11-Aug-2016

- Section 4.3.3: Add section about not installing noise filter components when you use the OCXO

7 04-Oct-2016

- Added clarification that the button press should be a short one.

8 12-Jan-2017

- Corrected pin numbers of Si5351A Synth module in the circuit diagram