Quad RF Synthesizer

QRF041/QRF241

Version 0.2.0, Rev 2 hardware

Limitation of Liability

MOG Laboratories Pty Ltd (MOGLabs) does not assume any liability arising out of the use of the information contained within this manual. This document may contain or reference information and products protected by copyrights or patents and does not convey any license under the patent rights of MOGLabs, nor the rights of others. MOGLabs will not be liable for any defect in hardware or software or loss or inadequacy of data of any kind, or for any direct, indirect, incidental, or consequential damages in connections with or arising out of the performance or use of any of its products. The foregoing limitation of liability shall be equally applicable to any service provided by MOGLabs.

Copyright

Contact

For further information, please contact:

MOG Laboratories P/L
49 University St
Carlton VIC 3053
AUSTRALIA
+61 3 9939 0677
info@moglabs.com

MOGLabs USA LLC
419 14th St
Huntingdon PA 16652
USA
+1 814 251 4363
www.moglabs.com

Preface

Acousto-optic modulators (AOMs) are an integral part of many modern laser-based experiments. They are used for frequency shifting, amplitude modulation, and laser frequency stabilisation. The MOGLabsQRF quad RF synthesizer provides a user-friendly device that can simultaneously control four AOMs at once, with direct output of up to 2 W per channel.

The wide frequency range (10 to 200 MHz) is compatible with most common AOMs, and it provides high-resolution frequency, amplitude and phase control. The QRF provides analogue modulation of each channel with 70 kHz bandwidth, ergonomic front-panel controls, and ethernet/USB interface. The included table mode allows generation of predefined time-dependent waveforms. The QRF is is single small (half-rack width) box which connects directly to AC mains power.

As you delve into this manual you will uncover more and more capability, but the microcontroller at the heart of the QRF allows software improvements to add new features, so please check the MOGLabs website for updates, example code, and assistance.

We hope that you enjoy using the QRF, and please let us know if you have any suggestions for improvement in the QRF or in this document, so that we can make life in the lab better for all.

MOGLabs, Melbourne, Australia
www.moglabs.com

Safety Precautions

Safe and effective use of this product is very important. Please read the following safety information before attempting to operate. Also please note several specific and unusual cautionary notes before using the MOGLabs QRF, in addition to the safety precautions that are standard for any electronic equipment.

CAUTION To ensure correct cooling airflow, the unit should not be operated with cover removed.

WARNING High voltages are exposed internally, particularly around the mains power inlet and internal power supply unit. The unit should not be operated with cover removed.

NOTE The MOGLabs QRF is designed for use in scientific research laboratories. It should not be used for consumer or medical applications.

Protection Features

The MOGLabs QRF includes a number of features to protect you and your device.

Open/short circuit Each RF output should be connected to a $50\,\Omega$ load where possible. The high-power output can tolerate open-circuit, but should not be short-circuited.

Mains filter Protection against mains transients.

Temperature Several temperature sensors control the fan and will trigger a shutdown if the temperature exceeds a safe limit.

RoHS Certification of Conformance

MOG Laboratories Pty Ltd certifies that the MOGLabs Diode Laser Controller (Revision 3) is RoHS-5 compliant. MOG Laboratories notes, however, that the product does not fall under the scope defined in *RoHS Directive 2002/95/EC*, and is not subject to compliance, in accordance with *DIRECTIVE 2002/95/EC Out of Scope; Electronics related; Intended application is for Monitoring and Control or Medical Instrumentation*.

MOG Laboratories Pty Ltd makes no claims or inferences of the compliance status of its products if used other than for their intended purpose.

Contents

Getting started

1. **Connect to mains power** using the supplied IEC cord.
2. **Power on the device** using the rocker switch on the rear.
3. Wait for the device to boot and the menu system (§2.2) to be displayed.
4. **Set power limits** for each channel using the Options menu. Devices are shipped with a 30 dBm limit that should be adjusted as required. The power limits should be chosen to ensure attached devices are not damaged by accidentally outputting too much power.
5. **Attach devices** to the RF SMA connectors, which are nominally $50\,\Omega$ loads. Under no circumstances should the RF output be short circuited.
6. **Switch on** output using the LHS front-panel push-buttons. Channels without attached loads should not be enabled.
7. **Modulation inputs** can be used to control the RF with an analog signal (chapter 4). Each channel has an associated SMA input that accepts ± 1 V.
8. **External control** of the signal is achieved with TTL SMA inputs, which can be used to generate sharp pulses (§4.5).

Host connection

The recommended mode of operation is using the Windows™ host application (chapter 3) which provides a simple interface for controlling device functionality. Instructions for connecting via ethernet and USB are provided in Appendix D.

1. Introduction

The MOGLabs QRF consists of an AD9959, which provides four independent direct digital synthesizer (DDS) sources, and a 2 W amplifier for each channel. The frequency, amplitude and phase of each output is software-controlled via a microcontroller, which can be adjusted in real-time using the front-panel control knobs, or via a scripting language over ethernet or USB. The RF parameters can be defined in a lookup table (loaded via ethernet or USB) to enable complex sequences with fast transitions.

The block diagram below shows the key components. A microcontroller programs the DDS, which generates RF output that is low-pass filtered, pre-amplified, and then further amplified with a custom GaN hybrid high-power output stage (QRF241 only).

The microcontroller provides a front-panel display with rotary encoders (knobs) and push-buttons, as well as external interfaces over TCP/IP and USB communications.

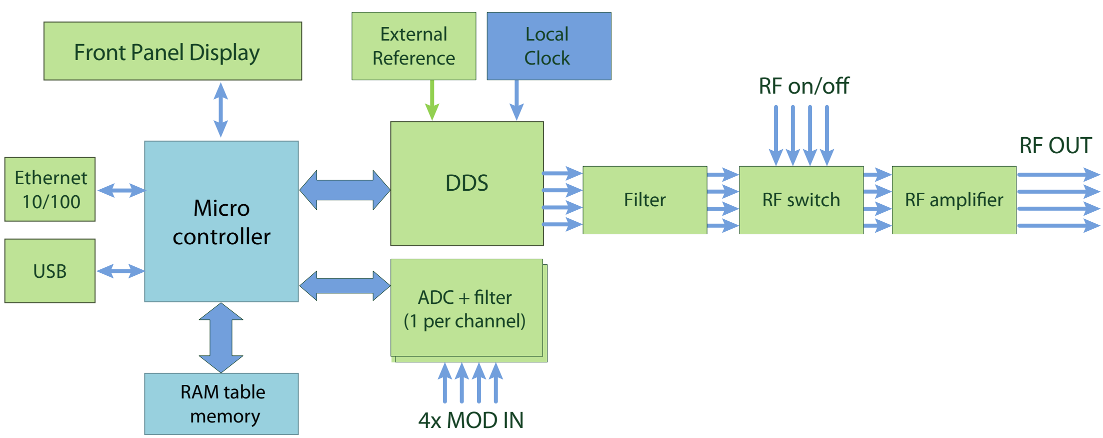

Figure 1.1: Block diagram of major components in the QRF.

The device allows analogue modulation through analogue-to-digital converters (ADCs) with anti-aliasing filters. When modulation is enabled, the microcontroller reads the value of the modulation signal and uses that value to adjust the DDS frequency, power and/or phase.

The QRF includes memory for storing complex waveform sequences, where each step in the sequence defines the frequency, power, phase and duration of that step. This capability can be accessed via either TCPIP or USB communications. See Chapter D for information on communications options and setup.

Once communications are established, the QRF is controlled with simple text commands. The commands can be very basic, for example to define the frequency or power, or they can define complex dynamic sequences. Appendix C provides a summary of the available commands.

1.1 Operating modes

The QRF can be used at varying levels of complexity, as either a free-running RF source or to follow pre-determined instructions defined in a table. The modes of operation are outlined below, and the current operational mode of each channel can be individually set using the `MODE` command.

NSB: Basic mode

Default state on power-up. In this mode, each channel acts as a simple single-tone RF source. The frequency and power of the signal can be controlled via the front panel, using simple instructions over the computer interface (e.g. `FREQ` or `POW`), or using the modulation inputs. Basic mode is convenient for driving AOMs and other single-frequency devices, with the flexibility of modulation.

NSA: Advanced mode

Advanced mode provides direct user-control of the DDS and its internal registers via the DDS command. Direct programming of each DDS is complex and not necessary for most applications; it requires careful reference to the AD9959 datasheet and manual calculation of the hardware registers.

TSB: Simple table mode

In table mode, the RF parameters are automatically sequenced by the microcontroller using a table stored in SDRAM. The table entries are defined by simple text commands from the host computer which define the RF frequency, amplitude, phase as detailed in chapter 5. The minimum duration of a TSB entry is 5 μs and each table can comprise up to 8191 instructions.

1.2 Feature compatibility

The QRF provides a wide range of functionality, but not all features are compatible with each other. The following table summarises which features can be used in which modes.

	NSB	NSA	TSB
Front-panel controls	✓	✗	✗
External modulation (AM/FM/PM)	✓	✗	✗
PID control	✓	✗	✗
Direct TTL on/off control	✓	✓	✗
Direct DDS control	✗	✓	✗
Autonomous execution	✗	✗	✓
External TTL trigger	✗	✗	✓

Table 1.1: Summary of feature compatibility

2. Connections and controls

2.1 Front panel controls

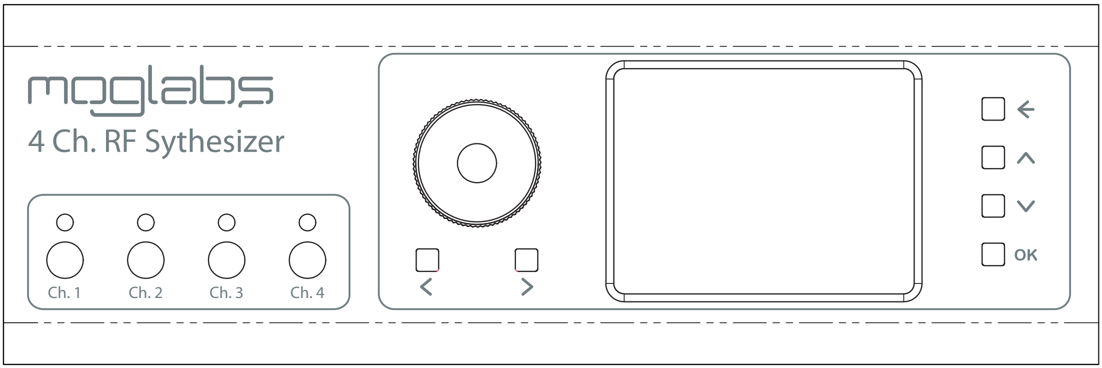

Figure 2.1: Front-panel layout of QRF devices.

The front-panel includes an interactive menu system for displaying the device state and controlling its settings. Each channel has an individual on/off button on the left-hand side, and an associated multicolour status LED that indicates the current output state of the channel as shown in the following table.

Colour	DDS active	Amplifiers
Off	✗	✗
Green	✓	✓
Yellow	✓	✗
Blue	✗	✓
Purple	Debug mode	
Red	Error state	

2.2 Menu system

The main menu (Figure 2.2) shows the current mode and status of each channel. In basic (NSB) mode, the current frequency and power of each channel is displayed, as well as any active modulation. Pressing the OK button with a channel selected will open the sub-menu to adjust settings for that channel (Figure 2.3).

The buttons on the right-hand side navigate through the menu structure and the encoder wheel is used to edit values. The ∧ and ∨ keys change between menu items, ← exits to the previous menu, and OK enters the selected menu or activates the selected command.

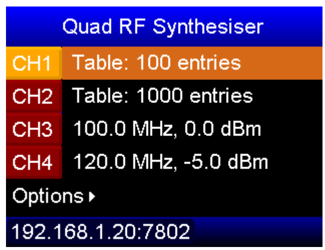

Figure 2.2: The main menu shows the current state of each channel. CH1 and CH2 are in table mode, with the number of entries in the table shown. CH3 and CH4 are in basic mode.

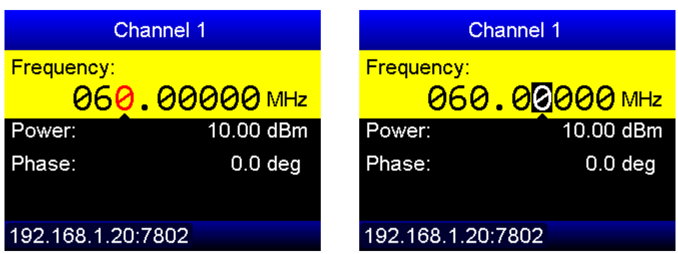

Figure 2.3: The basic parameters of each channel can be edited directly. Turning the encoder wheel modifies the selected digit of the current value (left) as indicated by the arrow. Pressing the encoder wheel changes to *digit select mode* (right), allowing the selected digit to be changed by turning the encoder.

The color of each menu item represents its purpose, as listed below.

White Static value, displayed for diagnostic purposes.

Yellow Adjustable value, modified using the encoder wheel.

Orange Currently selected channel.

Blue Submenu, entered with the OK button.

Green Command, executed by the OK button.

When an editable (yellow) value is selected, turning the encoder wheel changes the value of the selected digit as identified by the arrow and red text. To change the digit of interest, either use the < or > buttons or press the encoder wheel to change to *digit selection mode*. In this mode, the currently selected digit is shown on a black background, and is changed by turning the encoder wheel. Pressing the encoder again returns to *value modification mode*.

The options menu (Figure 2.4) allows the device configuration to be adjusted, such as the power limit of each channel (see also the `LIM` command). The ethernet settings of the device can also be set using this interface, including the static IP address of the device. When in use, the network status is displayed on the display footer, and once connected displays the current IP address.

Note that the "Restart ethernet" command must be used before changes in the ethernet menu will take effect.

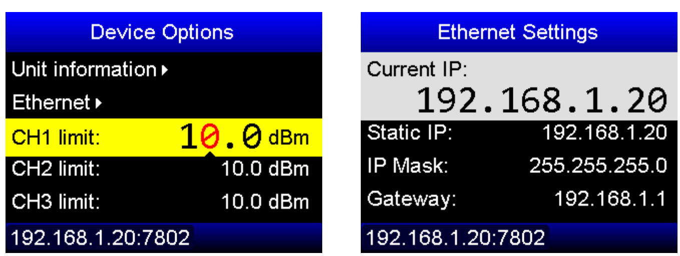

Figure 2.4: The options menu allows configuration of various settings, such as the maximum output power (left) and ethernet options (right).

The overall brightness of the display can be set with the *contrast* value in the Options menu. The display also includes a *sleep* timer that dims the display if it hasn't received input in a given period of time. This can be disabled by setting the sleep timer value to 0.

2.3 Rear panel controls and connections

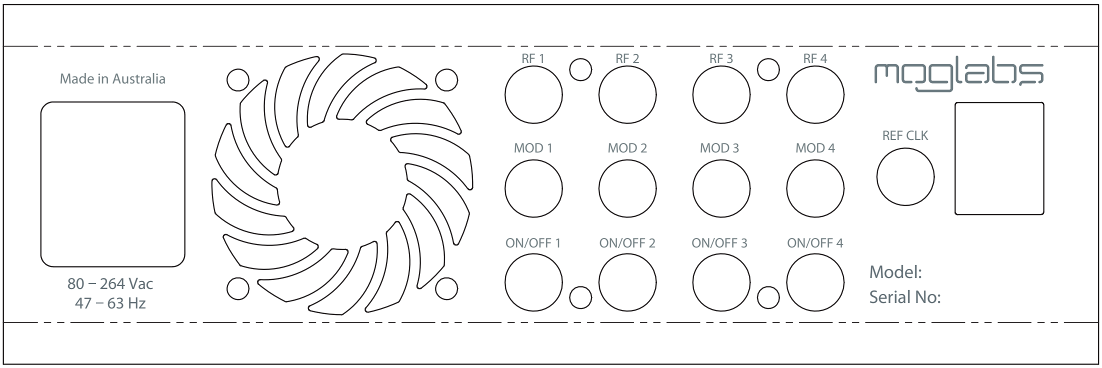

Figure 2.5: QRF rear panel layout.

IEC power in The QRF is compatible with all standard AC power systems, from 90 to 264 V and 47 to 63 Hz.

Fan The QRF has five temperature-controlled fans directing air flow over the RF power amplifiers, exhausting through the rear vent. Ensure that the vent does not become blocked.

RF SMA connectors for the primary RF outputs. Nominally connected to a 50Ω load, **must not be short-circuited**.

MOD Each channel has an associated SMA analog input, for AM/FM/PM/PID applications (see chapter 4).

ON/OFF SMA digital inputs for controlling the RF switch (see §4.5), compatible with both 3.3 V and 5 V logic. Also used for external triggering in table mode.

CLK IN The QRF can be synchronised to a high-performance external clock input via this SMA connector. The input is 50 Ω terminated, and the provided 25 MHz reference must be between 0 and +5 dBm.

RJ45/USB-A Ethernet (TCP/IP 10/100 Mb/s) and USB communications jacks.

2.4 Internal DIP switches

Four DIP switches are provided to assist in diagnosis and recovery of the QRF units. They should all be left in default configuration (set to OFF) for regular operation.

WARNING There is potential for exposure to high voltages inside the QRF. Take care around the power supply and ensure that objects, particularly electrically conducting objects, do not enter the unit.

CAUTION The cover should be replaced before powering on to ensure proper airflow and cooling.

	Function
DIP 1	Enter firmware update mode
DIP 2	Disable amplifiers
DIP 3	Use factory network settings
DIP 4	Initiate factory reset

3. MOGRF host software

The `mogrf` software package provides a simple user interface to the basic behaviour of QRF devices, with the ability to issue commands, run scripts, control tables, and apply firmware updates. This is the same software used to control ARF/XRF devices.

Please note: It may be necessary to install a firmware update (see Appendix B) to use the software described in this section.

3.1 Device discovery

Upon starting the application, a device discoverer (Figure 3.1) is initiated. This program scans the COM ports of the host computer looking for a QRF device, and then scans the local network subnet. Starting the application is then as simple as selecting the device and clicking *Connect*. If your device is not listed, recheck your connection and network settings.

If the network and/or firewall blocks device discovery, enter the IP address of the unit in the *Device address* box directly.

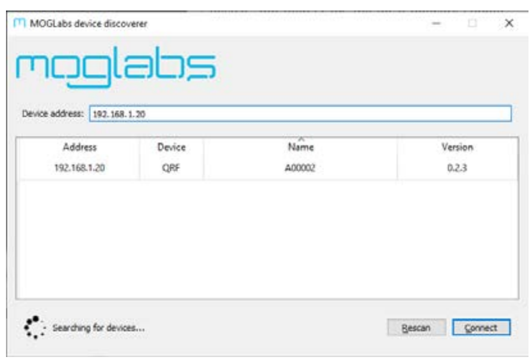

Figure 3.1: Example of the *Device discoverer* window, showing that one networked device was detected.

3.2 Device commander

The *Device commander* is an interactive terminal for issuing commands and queries to your QRF device and displaying the result (Figure 3.2). The accepted commands and their functions are listed in Appendix C. Type statements into the *Command* box and execute them by pressing the ENTER key or clicking Send. The window contains a history of recently executed commands.

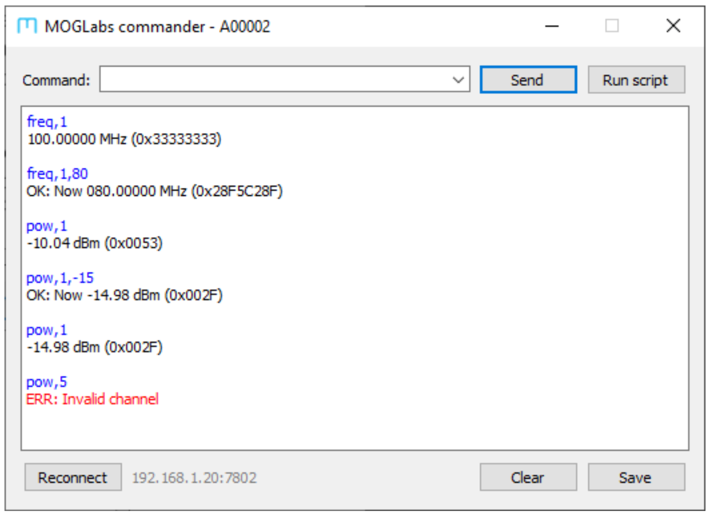

Figure 3.2: The *Device commander* window, which permits the execution of individual instructions or of text files containing scripts.

Scripts are ASCII text files where each line corresponds to a command to be executed (see Appendix E). Clicking *Run script* triggers stepwise execution of such a script, where the success of each statement is checked before executing the subsequent line. If an error occurs, execution of the script is aborted and an error message is displayed.

If the device is restarted or the connection is lost, clicking *Reconnect* will attempt to reestablish communication.

3.3 MOGRF main window

The main window of `mogrf` is shown below. The channels are displayed side-by-side, with information and controls that depend on the current operational mode of each channel.

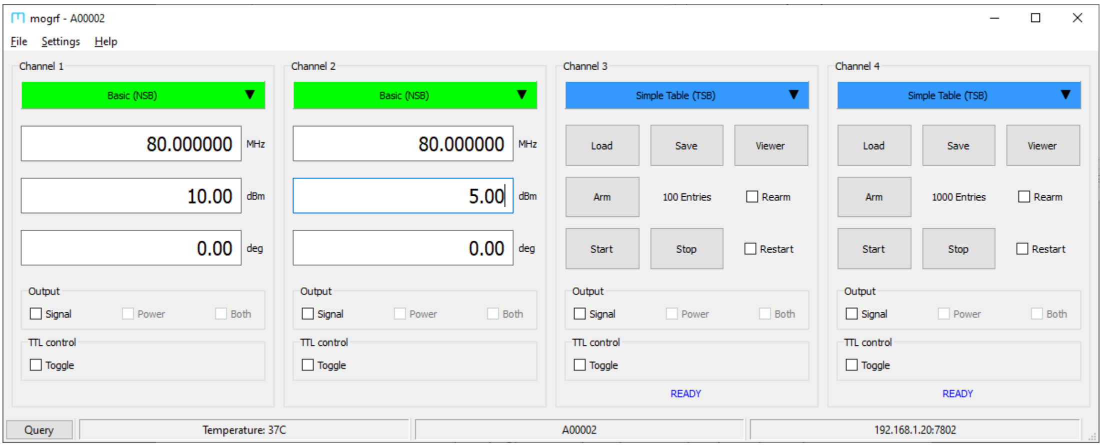

Figure 3.3: The main window of `mogrf`, showing CH1 and CH2 are in basic mode, and CH3 and CH4 are in table mode. Different options are presented in different operating modes.

Each channel is controlled individually as follows:

- The operational mode of the channel can be changed by clicking on the coloured box.
- In basic mode the frequency, amplitude and phase are displayed. Changing the value immediately updates the output.
- In table mode, buttons are provided to trigger specific functionality.
- Checkboxes control the RF switch (signal), RF amplifiers (power) or both (241-series only).

3.3.1 File menu

Device command Starts the *Device commander* (§3.2) for interactive execution of instructions to control the device.

Upload firmware Starts the firmware update application to upload and install updates on the device. The procedure for applying firmware updates is described in detail in Appendix B.

3.3.2 Settings menu

Ethernet Allows configuration of network connection settings (IP address, mask, gateway and port). Particularly useful for configuring the network settings over USB. Note that changing the *Static IP* only has an effect if DHCP is disabled, or if DHCP name resolution fails.

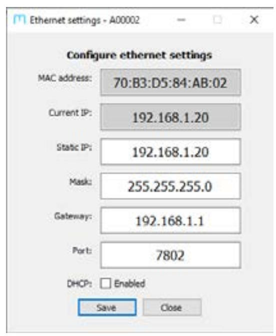

Figure 3.4: Ethernet configuration interface.

Note that changing the ethernet settings will require the application to be restarted, and may also require the device to be rebooted. The port should be unchanged at 7802 to ensure that the `mogrf` suite of programs can continue to communicate with the device.

Modulation The QRF supports a variety of modulation options as detailed in chapter 4. For each channel, the individual modulation types can be enabled/disabled and their gains adjusted using this dialog window (Figure 3.5).

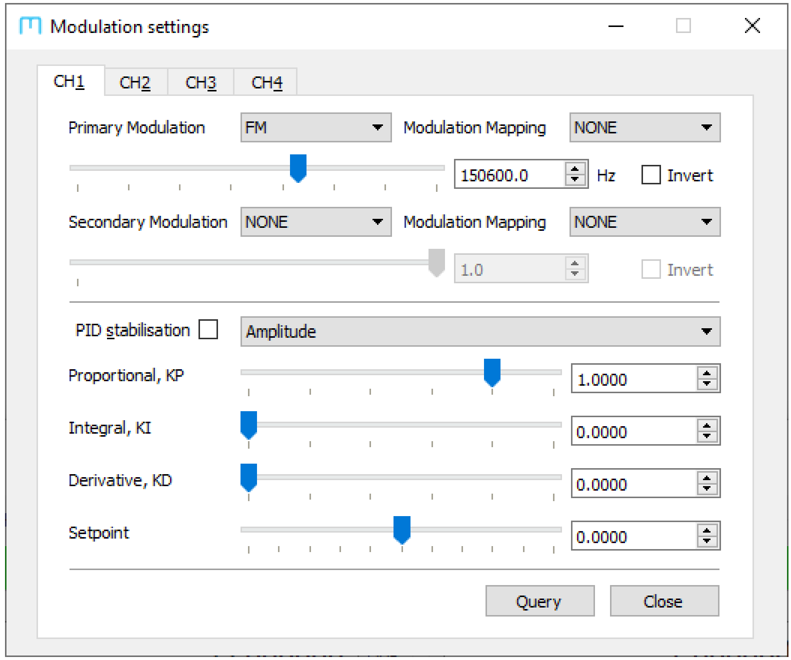

Figure 3.5: Use the Modulation dialog in the `mogrf` application to change the modulation settings for each channel

Each channel can have a primary and secondary modulation enabled (§4.3), which use the associated physical modulation input mapping. The gain can be set using the slider or by entering a value directly. Negative gain is specified by ticking the *Invert* check box. Similarly, PID can be enabled on the primary channel (§4.4) and the gains adjusted in the same way.

4. External modulation

The QRF supports external modulation of the RF in NSB mode through the modulation input SMA connectors on the back-panel. Frequency, amplitude and phase modulation of the RF are supported.

WARNING: The modulation inputs are nominally ±1V, and can be permanently damaged by applying higher voltages. Disconnected inputs are floating and can cause unexpected results when modulation is enabled.

4.1 Operational principle

Modulation is performed by digitising the analogue input signal, which is then multiplied by the modulation gain and added to the internal control value associated with the particular modulation mode ("*frequency tuning word*" for frequency, "*amplitude scale factor*" for power or "*phase offset word*" for phase). Limits are applied to ensure that the value remains within the bounds.

The ADC has 12-bit resolution (±1 V range), has 2nd-order anti-aliasing filters at 100 kHz and a measured 3-dB bandwidth of 70 kHz. Modulation is enabled/disabled with the `MOD` command[1]. For example, to enable AM on channel 1, use the command `MOD,1,AM,ON`.

Note than no amplitude-compensation is applied in FM mode, so changes in amplitude with frequency are expected in applications that sweep the frequency by a large amount. Consult the test report shipped with your QRF for an indication of the response curve, or use dual-modulation with AM to maintain a desired output power.

Note that modulation is not available in table mode.

[1] The "`MDN`" command is a synonym for compatibility with other products.

4.2 Modulation gain

The modulation depth is controlled by the "gain" and is set using the GAIN command. Each channel has a separate gain control for each modulation mode (amplitude, frequency or phase), which can be negative to indicate that the modulation action is inverted.

The gain can be specified with physical units, which corresponds to the modulation produced for +1V input. For example, when +1V is applied to the associated modulation input, GAIN,1,FREQ,-10MHz will shift the output frequency down by 10 MHz and GAIN,1,PH,180deg will invert the phase. Alternatively the gain can also be specified as a signed 32-bit integer in hexadecimal with negative values represented using two's complement.

The amplitude gain is represented as a percentage of **full-scale** amplitude. Therefore a gain of 100% will bring the output to **full power** at +1 V input (as restricted by the channel's power LIMIT). To prevent unexpected results, it is recommended to verify the modulation gain by disconnecting the load and checking the amplitude of the RF output on an oscilloscope with 50Ω termination.

4.3 Dual modulation

The QRF supports dedicating two modulation inputs to a single channel in "dual modulation" mode, enabling applications such as simultaneous AM and FM. In particular, this permits PID stabilisation to dynamically correct for the frequency-dependence of an attached load (such as the diffraction efficiency of an AOM).

In order to use this mode, the physical modulation inputs on the back-panel must be mapped to the desired destination channel. Each channel can have assigned "primary" and "secondary" inputs, which can be reconfigured using the control application (Figure 3.5) or the MAPMOD command.

Each physical input can only be used for one purpose at a time, so if an input is used for primary modulation on a channel, it cannot *simultaneously* be used for secondary modulation on a separate channel. Furthermore, to prevent confusion it is recommended that the primary modulation channel remain assigned to the RF channel with the same number.

Once configured, the primary and secondary modulation inputs can then be activated using the MOD command as shown in the example below.

```
# this example shows dual modulation on CH1
# we configure the MOD1 input for FM and MOD2 for AM
MAPMOD,1,1,2
MOD,1,FM,AM
# configure the modulation gains
GAIN,1,FM,10 MHz # 10 MHz/V
GAIN,1,AM,10% # 10% of full—scale amplitude
```

Listing 4.1: Demonstration of simultaneous AM+FM on CH1.

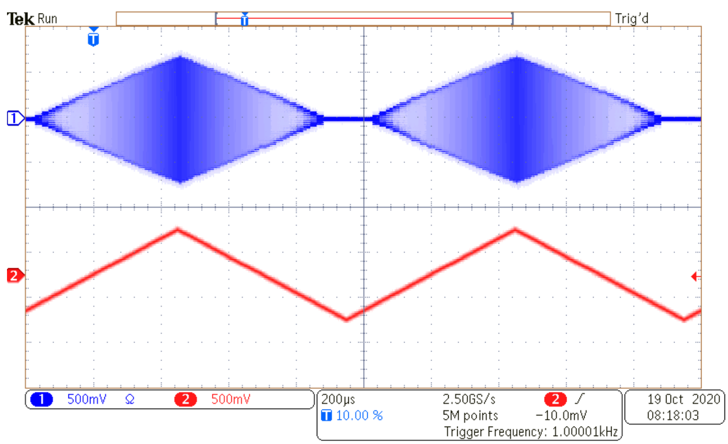

Figure 4.1: Example waveform (blue) showing a chirped pulse generated with the dual-modulation example by feeding a triangle wave (red) into both primary and secondary modulation inputs.

The modulation mapping is stored in non-volatile memory, but it is strongly recommended that any setup scripts explicitly include the modulation input mapping to prevent confusion.

The active purpose of the modulation inputs can be verified using the `MAPMOD` command. For the example above CH1 uses both MOD1 and MOD2 inputs, which are both presently active. Querying the `MOD` command shows that these are being used for FM and AM respectively.

```
>> mapmod,1
<< 1 (ACTIVE), 2 (ACTIVE)
>> mod,1
<< FM, AM
```

Listing 4.2: Example verifying the modulation configuration of CH1.

There are several restrictions in dual-modulation mode:

- Simultaneous modulation of a single parameter is not permitted (e.g. two inputs both controlling FM of one channel).
- Mapping a second modulation input to the same modulation mode will override the first (e.g. attempting to assign two inputs to CH1 primary).
- If a single modulation input is enabled for primary and secondary modulation on different channels, the secondary modulation will be disabled.
- PID can only performed on the primary modulation input.

4.4 PID stabilisation

In addition to direct modulation, the QRF also provides PID control loops which can be used in conjunction with an AOM to perform intensity (also called *noise eating*) or frequency stabilisation of a laser. Each channel has an independent PID controller, which acts to drive an *error signal* provided to the modulation input towards zero by adjusting the frequency, power or phase as appropriate.

AN001: An application note is available from the MOGLabs website that describes PID control in more detail, and provides instructions for tuning the PID loop.

Upon activating the PID controller, the channel's modulation input is treated as an *error signal* instead of a control voltage, and the action of the controller is to drive this signal towards zero. Locking to a non-zero value can be achieved with the `PID,SETPOINT` command. However, to make best use of the analog-to-digital converter's dynamic range, it is recommended to remove the set-point before digitisation using a *signal conditioning* circuit (Figure 4.2).

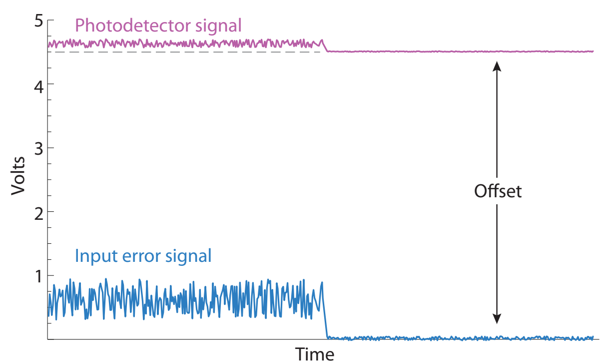

Figure 4.2: Photodetector signal (magenta) and conditioned error signal (blue) before and after activating noise-eater functionality. The conditioning board removes the DC offset and applies gain, improving sensitivity and ensuring the signal remains between ±1 V.

A clamping circuit should be used to ensure that the signal fed into the QRF does not exceed the ±1 V modulation input tolerance, as this can damage the input ADCs.

For convenience, MOGLabs produces a signal-conditioning board (B3122) available as an optional extra, which provides:

1. Manual offset adjustment using a potentiometer
2. Analog offset subtraction (e.g. from DAC output)
3. Variable analog gain
4. Monitor outputs for both photodetector and error signals
5. Output protection, to prevent exceeding ±1 V.

The error signal is then converted into a control value using the standard PID function:

$$u(t) = Gk_p e(t) + Gk_i \int_0^t e(\tau) d\tau + Gk_d \frac{de}{dt},$$

where $e(t)$ is the input error signal, $u(t)$ is the feedback response, and G is the overall modulation gain. The gain constants k_p, k_i, k_d are floating-point values in the range $[0, 1]$ which tune the proportional, integral and differential responses respectively. Typical values depend on the application, but reasonable starting values are often $k_p = 0.1$, $k_i = 0.01$ and $k_d = 0$.

When optimising a PID control loop, it should be kept in mind that the achievable loop bandwidth is limited by the propagation delay of the entire signal processing chain, not just the modulation bandwidth. This includes the impulse response of the AOM, photodetector and signal-processing electronics, as well as the QRF. Typical results achieved with standard AOMs are shown in the application note.

4.5 TTL switching

A versatile feature of the QRF is the ability to switch the RF in response to an external input such as a tactile switch or a TTL trigger for device synchronisation. The switching bypasses the microcontroller, enabling very low latency switching to be achieved (Figure 4.3).

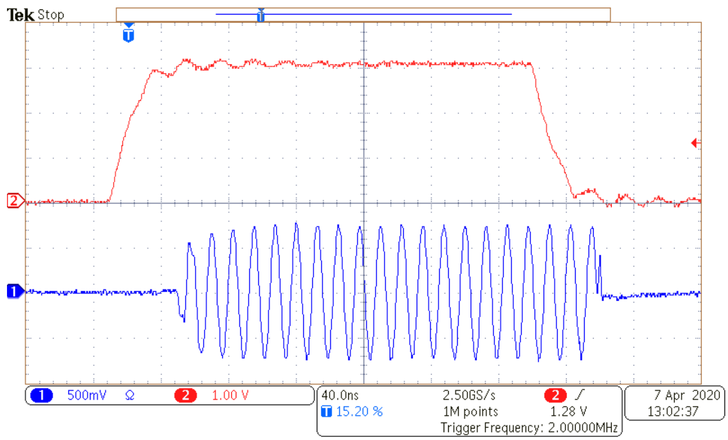

Figure 4.3: Example of pulse generation using the ON/OFF input, showing a response time of $< 40\,\mathrm{ns}$. Red is the TTL signal; blue is the RF signal.

Each channel has one TTL input on the rear panel labelled ON/OFF. In basic (NSB) mode, pulling the TTL LOW will rapidly disable the RF switch, and returning it HIGH will reenable the switch. In table mode it is used as a hardware trigger for synchronising the table execution.

Note: Digital inputs are pulled *high*, meaning that a disconnected input pin is equivalent to supplying a TTL high to that input.

5. Table mode

Table mode performs sequential execution of up to 8191 instructions with precise timing. This enables generation of complicated pulse sequences, custom envelope shapes, and automated control of experiment sequences through digital I/O.

5.1 Operational principle

Each channel has its own independent table comprising a number of entries that describe the frequency, amplitude and phase of the rf output at each step. Once the sequence has been defined using the `TABLE,ENTRY` or `TABLE,APPEND` commands (see §C.7), it is readied for execution using the `TABLE,ARM` command. The table is checked for errors, and will fail if an incompatibility is detected.

Once the table is armed, execution is started by either a hardware TTL trigger on the channel's ON/OFF input or using the `TABLE,START` command. The table can be automatically restarted after completion by enabling the `TABLE,RESTART` option, and execution can be stopped mid-sequence using the `TABLE,STOP` command.

When a table is armed, the RF is switched on and upon completion of the table the final RF state remains ongoing. If it is required that the output be disabled when the table is complete, the final entry should set the power to `0x0` (zero amplitude, not 0 dBm).

A simple example of a table mode shape is shown below with the resulting output demonstrated in Figure 5.1. The channel it put into table mode and any existing table is cleared. The steps in the sequence are then added one by one, and the table is armed. Once the arming is complete, the table is executed. Although is it not necessary to separate the "arm" and "start" steps, it can be preferable to do so to assist in debugging.

```
MODE,1,TSB
TABLE,CLEAR,1
TABLE,APPEND,1,20MHz,0dBm,0,0x1
TABLE,APPEND,1,50MHz,5dBm,0,0x1
TABLE,APPEND,1,100MHz,10dBm,0,0x1
TABLE,APPEND,1,50MHz,-5dBm,0,0x1
TABLE,APPEND,1,20MHz,5dBm,0,0x1
TABLE,APPEND,1,20MHz,0x0,0,0x1
TABLE,ARM,1
TABLE,START,1
```

Listing 5.1: Demonstration of simple table mode commands.

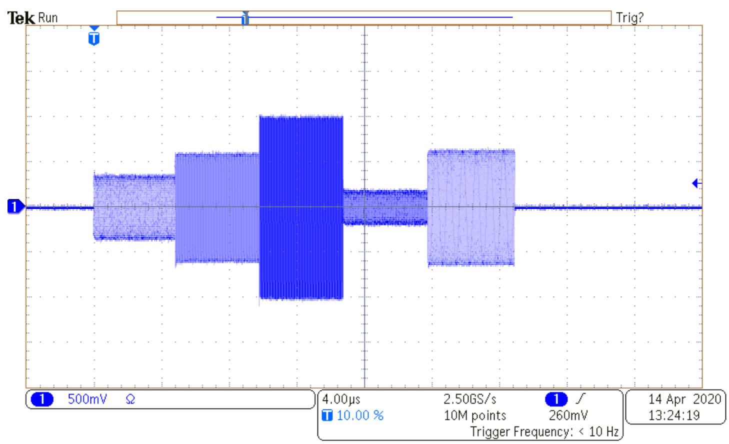

Figure 5.1: Output waveform generated by simple table mode example.

Typically more complicated pulse sequences are of interest, such as generated shaped envelopes with frequency sweeps. These tables are often generated by user-created scripts, or as part of a control system.

An example of such a pulse is shown in Figure 5.2 where the RF frequency is swept under a Gaussian envelope. Examples of how to do generate this pulse in several programming languages are provided in Appendix E.

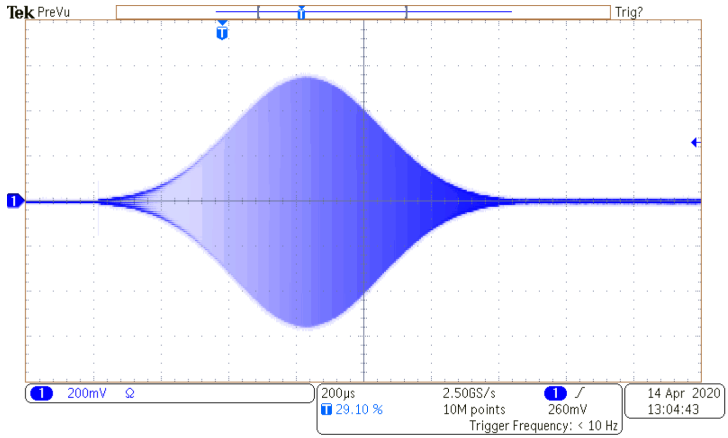

Figure 5.2: Demonstration of a more complicated table sequence involving a shaped pulse with frequency chirp.

5.2 External trigger

An external trigger can be used in table mode to synchronize with other devices. In thise mode the instruction will be repeated until a falling trigger is detected on the TTL input. Trigger mode is activated by specifying a duration of zero, or with the `TRIG` flag.

After a table is armed with the `TABLE,ARM` command, it will automatically begin executing upon a falling external trigger on the associated channel's TTL input. It is therefore typically not necessary to use the `TRIG` flag on the first table entry for such operation.

If a trigger is not received, the `TABLE,STOP` command should be used to abort operation.

5.3 Re-arm and restart

The QRF can be instructed to automatically re-arm the table after a successful execution using the `TABLE,REARM` command. This automatically prepares the table for execution from either hardware or software trigger once execution has finished.

Furthermore, the table can be repeated continuously by enabling the `TABLE,RESTART` option. This will cause the table to immediately re-execute after it has been rearmed, although an initial hardware or software trigger must be provided to begin the first execution.

A. Specifications

Parameter	Specification
RF characteristics	
Max output power	+33 dBm/+12 dBm (241/041 models)
Output impedance	50Ω
Amplitude control	10-bit resolution
Frequency	10 to 200 MHz
Frequency control	32-bit resolution; 0.116 Hz steps
Frequency stability	±25 ppm (0 to 50°C)
Phase	0 to 2π (14-bit resolution)
Phase noise	TBD
Signal to noise	TBD
Intermodulation and spurious	TBD
Channel crosstalk	< -70 dBc (off), < -50 dBc (on)
Power on, RF off	< -70 dBm
Analogue input	
Inputs	1 per channel (4 total)
Function	FM, AM, ϕ or PID
Sensitivity	±1 V
Bandwidth	70 kHz with 2nd order anti-alias
Resolution	12-bit
Input impedance	5 kΩ

Digital input (per channel)	
RF on/off	TTL hardwired, positive logic only
Trigger input	TTL input to continue table execution
TTL input high	2.2 V
TTL input low	0.6 V
Absolute max in	6.5 V
Absolute min in	-0.5 V

Table mode	
Min. step size	5 µs
Max. table length	8191 instructions per channel
Trigger options	Software, or TTL via rear panel
Channel sync	Configurable independent or synchronised

Mechanical & power	
Display	320x240 px colour LCD
Fans	5 x temperature controlled fans
IEC input	90 to 264 Vac, 47 to 63 Hz
Dimensions	W×H×D = 250 × 79 × 292 mm
Weight	2 kg

B. Firmware upgrades

From time to time, MOGLabs will release firmware updates that enable new functionality or address issues. This section provides instructions on how to apply firmware updates to your device.

> **WARNING:** Do not attempt to communicate with the device while a firmware upgrade is being applied, and do not interrupt an upgrade (or factory reset) in progress.

B.1 Upgrade via `mogrf`

The recommended way to install updates is using the "`mogrf`" application, accessed through the File menu. Information about the current firmware will be shown and you will be given the option to install an update (Figure B.1). Note that **the device should not be used while upgrading firmware**.

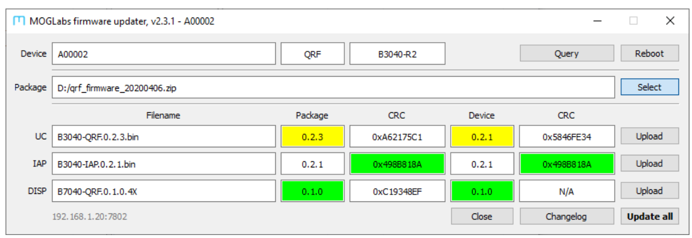

Figure B.1: The firmware update application connected to a unit, showing the device information (top) and current firmware versions (middle). Ensure that the model numbers are correct before continuing.

Update packages are available from the MOGLabs website and are loaded into the application by pressing the "Select" button. The firmware in the package is compared against the currently running version to determine which upgrades are required.

Each component of the firmware is compared against the package version and colour coded as follows:

Green: Component matches package version and is up-to-date.

Yellow: Package contains an update which should be applied.

Magenta: Package contains an **older** version than currently installed. Installing this component will **downgrade** the firmware.

Red: Currently installed version conflicts with package version and may be damaged. Installation of package version is strongly recommended.

Click on *Update all* to install all detected upgrades in sequence, otherwise individual components can be installed by clicking the *Upload* option next to each item.

The upload process proceeds through four stages:

1. The FLASH memory is erased to make room for the new image.
2. The data is uploaded to the device.
3. The device checks the data is received for consistency, to ensure the upload was successful.
4. The device is rebooted to load the new firmware.

The device will reboot after every individual component upgrade, to ensure the upload was successful before moving on to the next component. The software will automatically try to reconnect to the device.

B.2 Factory reset

If a firmware upgrade fails and the device subsequently cannot boot (even with DIP1 set to ON), a factory reset can be applied. Booting the device with DIP4 set to ON will then restore the configuration of the device as it was shipped. Once complete, the device will display a message on the front-panel LCD screen requesting that you return DIP4 to OFF and reset the device.

Note that all device settings will be overridden with factory defaults, including network settings, power limits, frequencies, calibrations, and so on. Ensure that relevant values are corrected after the reset.

Once the reset is complete, upgrade can be reattempted to gain access to newer features, or a different firmware can be applied. Please contact MOGLabs if you encounter any difficulties during firmware upgrade.

C. Command language

Please note: The command language is being continuously updated across firmware releases to improve functionality and add features. When upgrading firmware, please refer to the most recent version of the manual available at `http://www.moglabs.com`

C.1 Arguments

Most commands accept a comma-separated list of parameters, typically including a channel number "`ch`" (1, 2, 3 or 4). Optional parameters are listed in square brackets. Only the first few characters of the command name are significant.

Units can be specified for physical values as appropriate, namely:

Frequency Hz, kHz, MHz (default)

Power dBm (default), dB, mW, W

Phase deg (default), rad

Time ns, us (default), ms, s

Note that when requesting a particular value to change, the *actual* value may differ from the *requested* value because of discretisation and/or parameter limits. Typically the command will respond with the closest permitted value.

If required, values corresponding to internal representation can be specified directly using hexadecimal format with a "0x" prefix. This is potentially useful for stepping through the discrete values that the DDS is capable of generating.

All commands respond with a string that begins with either "OK" or "ERR" to indicate whether it was successful. It is **strongly recommended** that commands be checked for success.

C.2 General functions

REBOOT Initiate microcontroller reset, causing unit to reboot. Note that all communications links will be immediately closed.

INFO Report information about the unit.

VERSION Report versions of firmware currently running on device. Please include this information in any correspondance with MOGLabs.

TEMP Report measured temperatures and fan speeds.

C.3 Basic control

STATUS `STATUS[,ch]`
Reports the current operational status of the specified channel, describing whether the signal and amplifiers are switched on. Returns an error if the device is not operational.

MODE `MODE,ch[,type]`
Controls the operational mode of the given channel; `type` is one of NSB, NSA or TSB (see §1.1). Note that some options are only available in particular modes.

Note: this command automatically switches off the output of the specified channel.

OFF/ON `OFF,ch[,mode]` `ON,ch[,mode]`
Enable or disable the RF output of the specified channel. The signal and amplifiers can be individually controlled, which allows for more rapid switching response (see §4.5). `mode` is one of:

SIG Turn off/on the RF signal only.

POW Turn off/on the RF high-power amplifier only.
Note that the amplifiers take 2 s to completely power on.

ALL Turn off/on both the RF signal and high-power amplifier (default).

SLEEP `SLEEP,dt`
Pause microcontroller operation for `dt` milliseconds. Intended for use in simple scripts to wait for a short amount of time, e.g. for tables to finish execution. Note that the microcontroller will be unresponsive during this time.

C.4 Primary RF control

FREQUENCY `FREQ,ch[,value]`
Set channel `ch` to specified frequency as permitted by the DDS's 32-bit frequency resolution.

POWER `POW,ch[,value]`
Set channel `ch` to specified output power as computed using a factory calibration. To prevent damaging attached components, a power limit is applied with the `LIMIT` command.

LIMIT `LIMIT,ch[,value]`
Defines the maximum RF power for given channel `ch`, with the same syntax as the `POW` command. If the limit is set below the current power level, the current power is reduced to the limit.

PHASE `PHASE,ch[,phase]`
Set channel `ch` to have the specified phase offset.

ALIGNPHASE `ALIGNPH[,ch] ... [,ch]`
Simultaneously resets the phase accumulators of the associated channels. This may be necessary to re-establish a stable phase relationship between two channels after their frequencies are adjusted. If no channels are specified, all channels are simultaneously reset.

PHRESET `PHRESET[,onoff][,ch] ... [,ch]`
Enable or disable automatic phase alignment on the given channel(s), which performs an automatic phase alignment after a `FREQ` command is issued. This feature may be useful for applications such as IQ-modulation. If no channels are specified, it applies to all channels.

C.5 Modulation

MOD `MOD,ch,type[,onoff][,gain]`
`MOD,ch[,type1][,type2]`
`MOD,ch,PRI[,type]`
`MOD,ch,SEC[,type]`
Enables (or disables) modulation type of a given type (amplitude, frequency, or phase) for the selected channel.

type One of F[REQ] (frequency), A[MPL] (amplitude), P[HAS] (phase) or N[ONE].

onoff Can be ON or OFF. Reports whether this modulation mode is currently enabled if not supplied.

gain Optionally specify gain for this modulation type; equivalent to subsequently using the `GAIN` command.

If two types are specified, dual-modulation is enabled with the given primary and secondary modes (§4.3). Alternatively the primary and secondary modes can be configured separately by specifying PRI or SEC.

GAIN `GAIN,ch,type[,gain]`
Sets the modulation gain for the specified modulation type on the given channel. `gain` is a floating-point number with units or a signed hexadecimal integer that controls the depth of the modulation (§4.2). If `gain` is negative then the modulation action is inverted.

MAPMOD `MAPMOD,ch[,pri,sec]`
Configures the primary and secondary modulation inputs for the specified RF channel (see §4.3). The parameters `pri` and `sec` are 1–4 corresponding to the associated numbered input on the back-panel, or "*" for no change. Note that changing the modulation mapping may disable modulation on those channels, after which they may need to be re-enabled.

C.6 Clock reference

The DDS operates from an internal clock (SYSCLK) at 500 MHz, which is derived either from the on-board crystal oscillator at 25 MHz ("internal" mode) or from the back-panel SMA input labelled CLK IN ("external" mode).

The DDS can only accept a limited range of reference frequencies, including 25 MHz, 50 MHz, 100 MHz, 125 MHz and 500 MHz, but *not* 10 MHz. In applications where minimal phase noise is critical, it is recommended to directly supply a stable 500 MHz reference to run the SYSCLK without the PLL engaged[1].

CLKSRC `CLKSRC[,source][,freq]`

Query or set the current clock source. `source` is either `INT` to use the internal 25 MHz oscillator, or `EXT` to use the reference provided to the CLK IN connector. When using `EXT` mode, the parameter `freq` must be the frequency in MHz. An error will be generated if the clock frequency is not supported. Specifying the `freq` as either zero or 500 will disable the DDS's PLL and it will clock directly from the provided signal.

Configuring an external clock but not providing a stable clock signal results in undefined behaviour. It is strongly recommended that the external clock signal be provided before powering on the device.

> **Note:** The external reference clock must have power between 0 dBm and +5 dBm. **Never operate the QRF in external clock mode without providing a valid reference clock, as undefined behaviour can result.**

[1] Available in Rev3+ hardware only

C.7 Table mode

Table mode gives access to the powerful sequencing functionality of the QRF devices (chapter 5). The syntax is compatible with ARF/XRF devices where applicable.

STATUS `TABLE,STATUS,ch`
Reports the current execution status of the table.

CLEAR `TABLE,CLEAR,ch`
Stops and clears the table entries from the specified channel.

ENTRY `TABLE,ENTRY,ch,num,[freq,pow,phas,dur,flags]`
Configure the specified table entry. If only `ch` and `num` are given, the current entry of the table is returned.

ch The channel to edit.

num The entry to edit (1 to 8191).

freq Frequency to output during this step.

pow Output power during this step.

phas Phase offset of the RF for this step.

dur Duration of this step (discretised at 5 us). If zero, the entry is held until a hardware trigger is received.

HEXENTRY `TABLE,HEXENTRY,ch,num`
Queries the specified table entry, returning the internal hexadecimal representation of the associated frequency, amplitude and phase.

ENTRIES `TABLE,ENTRIES,ch[,num]`
Alternative: `TABLE,LENGTH,ch[,num]`
Defines the last table entry number for the given channel. Incorrectly setting the number of entries can result in undefined behaviour.

APPEND `TABLE,APPEND,ch,freq,ampl,phase,duration[,flags]`
Inserts the specified entry at the end of the table and increments the `TABLE,ENTRIES` counter.

COPY `TABLE,APPEND,src,dest`
Copies the table data from the `src` channel to the `dest` channel.

NAME `TABLE,NAME,ch,[‘‘name’’]`
Assigns a character string to the table for identification purposes. Use quotation marks to preserve case.

ARM `TABLE,ARM,ch[,...]`
Loads the table for execution and readies the output. The table then begins execution upon receiving a software trigger (`TABLE,START`) or hardware trigger on ON/OFF input. Multiple channels can be specified to arm them simultaneously.

START `TABLE,START,ch[,...]`
Provides a software trigger to initiate table execution. Multiple channels can be specified to start them simultaneously.

STOP `TABLE,STOP,ch[,...]`
Terminates an executing table at the end of the current step. Multiple channels can be specified to stop them simultaneously.

REARM `TABLE,REARM,ch[,on/off]`
Enables/disables the automatic re-arming (loading) of the table upon completion such that it can be started again from a hardware or software trigger.

RESTART `TABLE,RESTART,ch[,on/off]`
Enables/disables an automatic software-controlled restart of the table upon completion.

C.8 PID feedback

The QRF has an integrated PID-controller that can feed back to the amplitude, frequency or phase of the RF output in response to an error signal (see §4.4).

ENABLE `PID,ENABLE,ch,mode`
Enables the PID controller for the given channel in the specified `mode`, which is one of FREQ, AMPL or PHAS.

DISABLE `PID,DISABLE,ch`
Disables the PID controller for the given channel.

STATUS `PID,STATUS,ch`
Reports the current status of the PID controller and whether saturation occurred. The saturation flag is reset by this query.

SETPOINT `PID,SETPOINT,ch[,val]`
Applies a DC offset to enable locking to a non-zero set-point voltage. If specified, the voltage `val` must be between -1 and +1 V. Note that the lock may become unstable close to these limits due to clipping of the error signal.

INVERT `PID,INVERT,ch[,on/off]`
Inverts the controller action for the given channel.

GAIN `PID,GAIN,ch,name[,value]`
Sets the PID gain constant for `name`, which is one of `P` (proportional), `I` (integral) or `D` (derivative). `value` is a floating point number in the range [0,1]. The derivative gain can be negative, which inverts the sense of the derivative action with respect to the other components, and may be beneficial in some applications that require phase lead or lag.

ERROR `PID,ERROR,ch`
Returns the value of the error signal fed into the PID control loop, for diagnostic purposes.

C.9 Ethernet settings

INFO `ETH,INFO`
Reports current ethernet settings.

STATUS `ETH,STATUS`
Report the status of the ethernet connection.

WEB `ETH,WEB,onoff`
Enable or disable the integrated web server for device control.

STATIC `ETH,STATIC,ipaddr`
Set IP address based on the supplied dotted-quad string (for example "10.1.1.180"). The static address is used if DHCP fails, or is disabled.

MASK `ETH,MASK,ipmask`
Set IP mask based on dotted-quad string (for example "255.255.255.0").

GW `ETH,GW,ipaddr`
Set IP gateway based on dotted-quad string (for example "10.1.1.1").

PORT `ETH,PORT,port`
Set the TCPIP port number for device communication.

DHCP `ETH,DHCP,onoff`
Enable or disable DHCP.

D. Communications

The QRF can be connected to a computer by USB or ethernet (TCPIP). The software package `mogrf` (chapter 3) provides interactive functionality. Device control can also be integrated into existing control software following the protocol below using the commands in Appendix C. A detailed connection guide is available from the MOGLabs website.

D.1 Protocol

Device communication follows a query/response protocol using CRLF-terminated ASCII strings. Statements are either *commands* or *queries* depending on whether they cause an action to occur. A command will always respond with `OK` or `ERR` depending on whether the action succeeded.

It is strongly recommended that all software should wait for this response and check for success before continuing. The `python` and `LabVIEW` bindings provided take care of buffering and error checking automatically.

The *MOGLabs Device Commander* application is available as part of the `mogrf` package and provides a convenient interface for sending commands and receiving responses (see §3.2).

D.2 TCP/IP

The QRF can be accessed over ethernet via the IPv4 protocol. When ethernet is connected, the QRF will attempt to obtain an IP address by DHCP. If DHCP fails, an internally defined address will be used. In both cases, the address will be shown on the device display (for example, `10.1.1.190:7802`), showing the address and port number for communicating with the device.

D.2.1 Changing IP address

Depending on your network settings you may need to manually set the IP address. This is most easily done via the front-panel interface as detailed below.

1. From the main menu, open Options > Ethernet Settings.
2. Select **Static IP** and use the encoder wheel to set the IP address of the device as required. Pressing the encoder wheel changes between octets of the address.
3. Select **Gateway** and set the gateway address as required.
4. Select **DHCP** and set to OFF by turning the encoder anticlockwise.
5. Select **Restart ethernet** and press the OK button.
6. The new IP address will be displayed in the display footer.

In some situations it may be necessary to power-cycle the device to propagate ethernet changes.

D.3 USB

The QRF can be directly connected to a host computer using a USB cable (type A-male). The device will appear as a Virtual COM port, which behaves like an RS232 connection. The required STM32 Virtual COM Port Driver (VCP) device driver is available from the MOGLabs website for the Windows™ operating system. After installation, the QRF will appear as a COM port on the machine.

To determine the port number of the device, go to Device Manager (Start, then type `Device Manager` into the Search box). You should see a list of devices including Ports (Figure D.1).

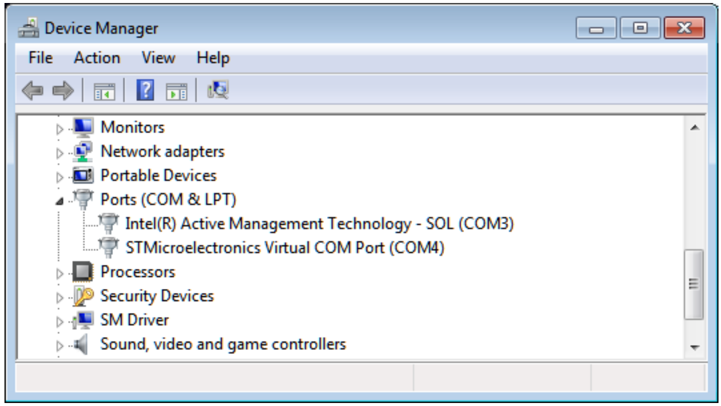

Figure D.1: Screenshot of Device Manager, showing that the QRF can be communicated with using COM4. The port number might change when plugging into a different USB port, or after applying a firmware update.

The device can be identified as a COM port with the following name,

`STMicroelectronics Virtual COM Port (COMxx)`

where `xx` is a number (typically between 4 and 15). In the example above, the device was installed as COM4.

Note that if the port appears in Device Manager with a different name, then the driver was not successfully installed. If this occurs, disconnect the device from the host computer, reinstall the VCP driver, then reconnect the USB cable.

E. Code examples

MOGLabs devices use standard protocols, allowing control from any programming language with appropriate standard libraries for network or serial communication. The following simple examples demonstrate how to communicate with the QRF over ethernet in several languages, using the helper bindings provided by MOGLabs.

E.1 python

Communication is handled by a "device" class, which provides convenience functions for sending commands and queries. The following example shows how to construct a Gaussian pulse using `numpy` and the `MOGDevice` class, with the resulting waveform shown in Figure E.1.

```
#--------------------------------------------
# Gaussian pulse example, (c) MOGLabs 2020
#--------------------------------------------
from mogdevice import MOGDevice
import numpy as np
# connect to the device
dev = MOGDevice('192.168.1.20')

# construct the pulse
N = 250 # number of points to generate
X = np.linspace(-1,1,N)
P = 5 - 35*X**2 # Gaussian amplitude, -30 to +10dBm
F = 100 + 50*X # linear sweep, 50 to 150 MHz

dev.cmd('MODE,1,TSB') # set CH1 into table mode
dev.cmd('TABLE,CLEAR,1') # clear existing table
for (f,p) in zip(F,P): # upload the entries
    dev.cmd('TABLE,APPEND,1,%.2f,%.2f,0,1'%(f,p))
dev.cmd('TABLE,START,1') # execute the table
```

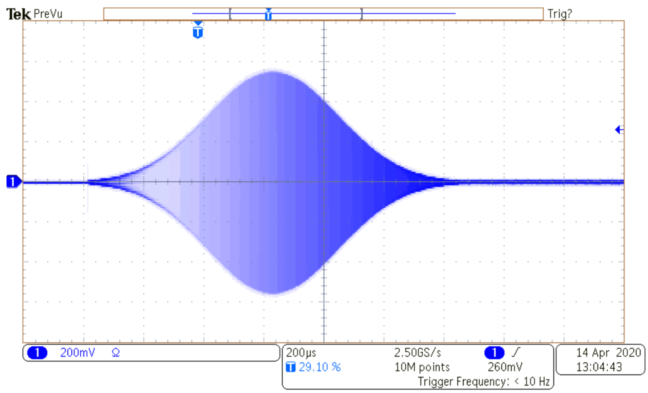

Figure E.1: Pulse with a Gaussian envelope, created in table mode using the example `python` code.

E.2 `matlab`

Similar to the `python` bindings, a device class is provided to make controlling the QRF easy using `matlab`. The listing below demonstrates how to generate the previous example in `matlab`.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% MATLAB pulse example, (c) MOGLabs 2020
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% create a device instance
dev = mogdevice();
% example: connecting by ethernet
dev.connect('10.1.1.31');
% print some information about the device
disp(dev.ask('INFO'));

% create a gaussian envelope
N = 250;
X = (1:N)/(N/2) - 1;
P = 5 - 35*X.^2;
F = 100 + 50*X;

% upload gaussian pulse in simple table mode
dev.cmd('MODE,1,TSB');
dev.cmd('TABLE,CLEAR,1');
for i=1:N
```

```
    % we can use printf notation when sending commands
    dev.cmd('TABLE,APPEND,1,%f,%f,0,0x1',F(i),P(i));
end
disp('Done')
dev.cmd('TABLE,ARM,1')
dev.cmd('TABLE,START,1')

% close the connection
delete(dev);
```

E.3 LabVIEW

The LabVIEW drivers provided make use of NI-VISA to provide a unified interface over both ethernet and USB. They perform automatic error checking, and are compatible with LabVIEW-2009 and later editions.

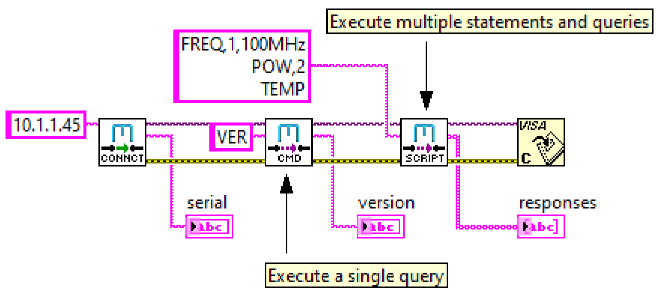

Figure E.2: Example LabVIEW program that connects to an QRF unit, and performs a number of queries

When using these drivers, it is strongly recommended that the automatic session close option be enabled, to prevent communications problems if the VI aborts or is interrupted. The option is found in Tools→Options→Environment→Automatically close VISA sessions.

MOG Laboratories Pty Ltd
49 University St, Carlton VIC 3053, Australia
Tel: +61 3 9939 0677 info@moglabs.com