RG50xQ&RM5xxQ Series
AT+QCRMTOOL Data Call Application Note

5G Module Series

Version: 1.0

Date: 2021-07-20

Status: Released

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://www.quectel.com/support/sales.htm.

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/technical.htm
Or email to support@quectel.com.

General Notes

Quectel offers the information as a service to its customers. The information provided is based upon customers' requirements. Quectel makes every effort to ensure the quality of the information it makes available. Quectel does not make any warranty as to the information contained herein, and does not accept any liability for any injury, loss or damage of any kind incurred by use of or reliance upon the information. All information supplied herein is subject to change without prior notice.

Disclaimer

While Quectel has made efforts to ensure that the functions and features under development are free from errors, it is possible that these functions and features could contain errors, inaccuracies and omissions. Unless otherwise provided by valid agreement, Quectel makes no warranties of any kind, implied or express, with respect to the use of features and functions under development. To the maximum extent permitted by law, Quectel excludes all liability for any loss or damage suffered in connection with the use of the functions and features under development, regardless of whether such loss or damage may have been foreseeable.

Duty of Confidentiality

The Receiving Party shall keep confidential all documentation and information provided by Quectel, except when the specific permission has been granted by Quectel. The Receiving Party shall not access or use Quectel's documentation and information for any purpose except as expressly provided herein. Furthermore, the Receiving Party shall not disclose any of the Quectel's documentation and information to any third party without the prior written consent by Quectel. For any noncompliance to the above requirements, unauthorized use, or other illegal or malicious use of the documentation and information, Quectel will reserve the right to take legal action.

Copyright

About the Document

Revision History

Version	Date	Author	Description
-	2021-04-28	Ozzy ANG	Creation of the document
1.0	2021-07-20	Ozzy ANG	First official release

Contents

Table Index

1 Introduction

As there is a default tool for data call on a Windows host, and the QMI interaction is completed via the driver layer, you can use **AT$QCRMCALL** to make a data call directly. However, on a non-Windows host, using **AT$QCRMCALL** alone is not enough because an external QMI client is required to participate in the interaction during a data call.

Unlike Windows OS, Linux OS has no data call tool ready for use. Therefore, it is impossible to make a data call with **AT$QCRMCALL** when no quectel-CM data call tool has been integrated into the system.

To make the data call practice handier, Quectel has developed the new AT command **AT+QCRMTOOL**, which enable users to make data calls on a Linux host with AT commands directly without the need to integrate the quectel-CM tool.

This document introduces how to use **AT+QCRMTOOL** and **AT$QCRMCALL** to make a data call in Linux OS.

1.1. Applicable Modules

Table 1: Applicable Modules

Module Series	Model
RG50xQ	RG500Q Series
	RG501Q Series
	RG502Q-EA
RM5xxQ	RM500Q Series
	RM502Q Series
	RM505Q-AE
	RM510Q-GL

2 AT+QCRMTOOL Introduction

2.1. AT Command Introduction

2.1.1. Definitions

- **<CR>** Carriage return character.
- **<LF>** Line feed character.
- **<...>** Parameter name. Angle brackets do not appear on the command line.
- **[...]** Optional parameter of a command or an optional part of TA information response. Square brackets do not appear on the command line. When an optional parameter is not given in a command, the new value equals to its previous value or the default settings, unless otherwise specified.
- **Underline** Default setting of a parameter.

2.1.2. AT Command Syntax

All command lines must start with **AT** or **at** and end with **<CR>**. Information responses and result codes always start and end with a carriage return character and a line feed character: **<CR><LF><response><CR><LF>**. In tables presenting commands and responses throughout this document, only the commands and responses are presented, and **<CR>** and **<LF>** are deliberately omitted.

Table 2: Types of AT Commands

Command Type	Syntax	Description
Test Command	**AT+<cmd>=?**	Test the existence of corresponding Write Command and return information about the type, value, or range of its parameter.
Read Command	**AT+<cmd>?**	Check the current parameter value of a corresponding Write Command.
Write Command	**AT+<cmd>=<p1>[,<p2>[,<p3>[...]]]**	Set user-definable parameter value.
Execution Command	**AT+<cmd>**	Return a specific information parameter or perform a specific action.

2.2. Declaration of AT Command Examples

The AT command examples in this document are provided to help you learn about the AT commands introduced herein. The examples, however, should not be taken as Quectel's recommendation or suggestions about how you should design a program flow or what status you should set the module into. Sometimes multiple examples may be provided for one AT command. However, this does not mean that there exists a correlation among these examples and that they should be executed in a given sequence.

2.3. AT+QCRMTOOL Description

2.3.1. AT+QCRMTOOL QCRMTOOL Data Call Tool Control

This command starts/stops the data call tool QCRMTOOL (Hereinafter referred to as QCRMTOOL), and configures/queries the parameters used by the tool.

AT+QCRMTOOL QCRMTOOL Data Call Tool Control	
Test Command **AT+QCRMTOOL=?**	Response **+QCRMTOOL: "CTL",(**list of supported **<status>**s**)** **+QCRMTOOL: "IP_TYPE",(**list of supported **<IP_type>**s**)** **+QCRMTOOL: "AGG_PROT",(**range of supported **<version>**s**)** **+QCRMTOOL: "DLAGG_GRAMS",(**range of supported **<DLAGG_grams>**s**)** **+QCRMTOOL: "DLAGG_SIZE",(**range of supported **<DLAGG_size>**s**)** **+QCRMTOOL: "ULAGG_GRAMS",(**range of supported **<ULAGG_grams>**s**)** **+QCRMTOOL: "ULAGG_SIZE",(**range of supported **<ULAGG_size>**s**)** **+QCRMTOOL: "EP_TYPE",(**range of supported **<ep_type>**s**)** **+QCRMTOOL: "IFACE_ID",(**range of supported **<iface_ID>**s**)** **+QCRMTOOL: "MUX_ID",(**range of supported **<MUX_ID>**s**)** **OK**
Write Command **AT+QCRMTOOL=<options>[,…]**	Response **OK** If there is any error:

	ERROR
Maximum Response Time	100 ms
Characteristics	This command takes effect immediately. The configurations will not be saved.

Parameter

<options>	String type. Indicates the parameter object to be set.	
	"CTL"	Start/Stop QCRMTOOL or query the status of QCRMTOOL
	"IP_TYPE"	Set the IP type used for QCRMTOOL data call
	"AGG_PROT"	Set the AGG protocol version used by QCRMTOOL
	"DLAGG_GRAMS"	Set the number of DL packets aggregated
	"DLAGG_SIZE"	Set the size of DL packets aggregated
	"ULAGG_GRAMS"	Set the number of UL packets aggregated
	"ULAGG_SIZE"	Set the size of UL packets aggregated
	"EP_TYPE"	Set the terminal type of QCRMTOOL
	"IFACE_ID"	Set the terminal interface ID of QCRMTOOL
	"MUX_ID"	Set the aggregation ID of QCRMTOOL

NOTE

1. The configurations set by this command will not be saved after the module reboots. Making the configurations take effect requires re-executing the corresponding commands after every reboot.
2. For details about **AT+QCRMTOOL=<options>,[…]**, see ***Chapter 2.3.1.1*** to ***Chapter 2.3.1.10***.

Example

```
AT+QCRMTOOL=?          //Test Command, list the supported range of parameter values.
+QCRMTOOL: "CTL",(0-1)
+QCRMTOOL: "IP_TYPE",(4,6,10)
+QCRMTOOL: "AGG_PROT",(5-9)
+QCRMTOOL: "DLAGG_GRAMS",(8-64)
+QCRMTOOL: "DLAGG_SIZE",(4096-31744)
+QCRMTOOL: "ULAGG_GRAMS",(8-64)
+QCRMTOOL: "ULAGG_SIZE",(1024-4096)
+QCRMTOOL: "EP_TYPE",(1-3)
+QCRMTOOL: "IFACE_ID",(1-5)
+QCRMTOOL: "MUX_ID",(129-137)

OK
```

2.3.1.1. AT+QCRMTOOL="CTL" Set/Query QCRMTOOL Service Status

This command queries QCRMTOOL service status, and starts/stops QCRMTOOL service.

AT+QCRMTOOL="CTL" Set/Query QCRMTOOL Service Status	
Write Command **AT+QCRMTOOL="CTL"[,<status>]**	Response If the optional parameter is omitted, get the current QCRMTOOL service status: **+QCRMTOOL: "CTL",<status>** **OK** If the optional parameter is specified, start/stop QCRMTOOL service: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect immediately. The configurations will not be saved.

Parameter

<status>	Integer type. Starts/stops QCRMTOOL service status. 0 Stop QCRMTOOL service 1 Start QCRMTOOL service

NOTE

1. After you execute **AT+QCRMTOOL="CTL",1** to start QCRMTOOL service, **OK** will be returned directly followed by an URC reporting service status. See ***Chapter 2.4.3*** for details on URC messages.
2. Do not execute **AT+QCRMTOOL="CTL",1** while the QCRMTOOL service has already been started, otherwise **ERROR** will be returned.

Example

```
AT+QCRMTOOL="CTL"          //Query the current status of the QCRMTOOL service.
+QCRMTOOL: "CTL",0

OK
```

```
AT+QCRMTOOL="CTL",1          //Start QCRMTOOL service.
OK
AT+QCRMTOOL="CTL",0          //Stop QCRMTOOL service.
OK
```

2.3.1.2. AT+QCRMTOOL="IP_TYPE" Set IP Type for QCRMTOOL Data Call

This command sets the IP type for making data calls with QCRMTOOL.

AT+QCRMTOOL="IP_TYPE" Set IP Type for QCRMTOOL Data Call	
Write Command **AT+QCRMTOOL="IP_TYPE"[,<IP_type>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "IP_TYPE",<IP_type>** **OK** If the optional parameter is specified, set an IP type for QCRMTOOL data call: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<ip_type>	Integer type. The IP type used for QCRMTOOL data call. 4 IPv4 6 IPv6 10 IPv4v6

Example

```
AT+QCRMTOOL="IP_TYPE"        //Query the IP type used for QCRMTOOL data call.
+QCRMTOOL: "IP_TYPE",10

OK
AT+QCRMTOOL="IP_TYPE",4      //Set an IP type for QCRMTOOL data call.
```

```
OK
```

2.3.1.3. AT+QCRMTOOL="AGG_PROT" Set AGG Protocol Version for QCRMTOOL Data Call

This command sets the AGG protocol version used for making data calls with QCRMTOOL.

AT+QCRMTOOL="AGG_PROT"	Set AGG Protocol Version for QCRMTOOL Data Call
Write Command **AT+QCRMTOOL="AGG_PROT"[,<version>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "AGG_PROT",<version>** **OK** If the optional parameter is specified, set an AGG protocol version for QCRMTOOL data call: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<version>	Integer type. The AGG protocol version for QCRMTOOL data call. Range: 5–9, corresponding to AGG protocol versions from V1 to V5. Default value: 9 (AGG protocol version V5).

NOTE

As **<version>** is related to the module and host driver, it is recommended not to modify it and just leave it in its default value.

Example

```
AT+QCRMTOOL="AGG_PROT"                  //Query the AGG protocol version for QCRMTOOL data call.
+QCRMTOOL: "AGG_PROT",9
```

```
OK
AT+QCRMTOOL="AGG_PROT",9        //Set an AGG protocol version for QCRMTOOL data call.
OK
```

2.3.1.4. AT+QCRMTOOL="DLAGG_GRAMS" Set the Number of DL Packets Aggregated

This command sets the number of DL packets aggregated for making data calls with QCRMTOOL.

AT+QCRMTOOL="DLAGG_GRAMS" Set the Number of DL Packets Aggregated	
Write Command **AT+QCRMTOOL="DLAGG_GRAMS"[,<DLAGG_grams>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "DLAGG_GRAMS",<DLAGG_grams>** **OK** If the optional parameter is specified, set the number of DL packets aggregated: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<DLAGG_grams>	Integer type. The number of DL packets aggregated. Range: 8–64. Default value: 32.

NOTE

As **<DLAGG_grams>** is related to the module and host driver, it is recommended not to modify it and just leave it in its default value.

Example

```
AT+QCRMTOOL="DLAGG_GRAMS"       //Query the number of DL packets aggregated.
+QCRMTOOL: "DLAGG_GRAMS",32
```

```
OK
AT+QCRMTOOL="DLAGG_GRAMS",32      //Set the number of DL packets aggregated.
OK
```

2.3.1.5. AT+QCRMTOOL="DLAGG_SIZE" Set the Size of DL Packets Aggregated

This command sets the size of DL packets aggregated for making data calls with QCRMTOOL.

AT+QCRMTOOL="DLAGG_SIZE"	Set the Size of DL Packets Aggregated
Write Command **AT+QCRMTOOL="DLAGG_SIZE"[,<DLAGG_size>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "DLAGG_SIZE",<DLAGG_size>** **OK** If the optional parameter is specified, set the size of DL packets aggregated: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<DLAGG_size>	Integer type. The size of DL packets aggregated. Range: 4096–31744. Default value: 31744.

NOTE

As **<DLAGG_size>** is related to the module and host driver, it is recommended not to modify it and just leave it in its default value.

Example

```
AT+QCRMTOOL="DLAGG_SIZE"      //Query the size of DL packets aggregated.
+QCRMTOOL: "DLAGG_SIZE",31744
```

```
OK
AT+QCRMTOOL="DLAGG_SIZE",31744    //Set the size of DL packets aggregated.
OK
```

2.3.1.6. AT+QCRMTOOL="ULAGG_GRAMS" Set the Number of UL Packets Aggregated

This command sets the number of UL packets aggregated for making data calls with QCRMTOOL.

AT+QCRMTOOL="ULAGG_GRAMS" Set the Number of UL Packets Aggregated	
Write Command **AT+QCRMTOOL="ULAGG_GRAMS"[,<ULAGG_grams>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "ULAGG_GRAMS",<ULAGG_grams>** **OK** If the optional parameter is specified, set the number of UL packets aggregated: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<ULAGG_grams>	Integer type. The number of UL packets aggregated. Range: 8–64. Default value: 11.

NOTE

As **<ULAGG_grams>** is related to the module and host driver, it is recommended not to modify it and just leave it in its default value.

Example

```
AT+QCRMTOOL="ULAGG_GRAMS"    //Query the number of UL packets aggregated.
+QCRMTOOL: "ULAGG_GRAMS",11
```

```
OK
AT+QCRMTOOL="ULAGG_GRAMS",11      //Set the number of UL packets aggregated.
OK
```

2.3.1.7. AT+QCRMTOOL="ULAGG_SIZE" Set the Size of UL Packets Aggregated

This command sets the size of UL packets aggregated for making data calls with QCRMTOOL.

AT+QCRMTOOL="ULAGG_SIZE"	Set the Size of UL Packets Aggregated
Write Command **AT+QCRMTOOL="ULAGG_SIZE"[,<ULAGG_size>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "ULAGG_SIZE",<ULAGG_size>** **OK** If the optional parameter is specified, set the size of UL packets aggregated: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<ULAGG_size>	Integer type. The size of UL packets aggregated. Range: 1024–4096. Default value: 4096.

NOTE

As **<ULAGG_size>** is related to the module and host driver, it is recommended not to modify it and just leave it in its default value.

Example

```
AT+QCRMTOOL="ULAGG_SIZE"          //Query the size of UL packets aggregated.
+QCRMTOOL: "ULAGG_SIZE",4096
```

```
OK
AT+QCRMTOOL="ULAGG_SIZE",4096      //Set the size of UL packets aggregated.
OK
```

2.3.1.8. AT+QCRMTOOL="EP_TYPE" Set the Terminal Type of QCRMTOOL

This command sets the terminal type of QCRMTOOL.

AT+QCRMTOOL="EP_TYPE" Set the Terminal Type of QCRMTOOL	
Write Command **AT+QCRMTOOL="EP_TYPE"[,<ep_type>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "EP_TYPE",<ep_type>** **OK** If the optional parameter is specified, set the terminal type of QCRMTOOL: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<ep_type>	Integer type. The terminal type of QCRMTOOL. 1 Reserved 2 USB Terminals 3 PCIe Terminals

NOTE

As **<ep_type>** is related to the module and host driver, it is recommended not to modify it.

Example

```
AT+QCRMTOOL="EP_TYPE"          //Query the terminal type of QCRMTOOL.
+QCRMTOOL: "EP_TYPE",2

OK
AT+QCRMTOOL="EP_TYPE",2        //Set the terminal type of QCRMTOOL.
OK
```

2.3.1.9. AT+QCRMTOOL="IFACE_ID" Set the Terminal Interface ID of QCRMTOOL

This command sets the terminal interface ID of QCRMTOOL.

AT+QCRMTOOL="IFACE_ID" Set the Terminal Interface ID of QCRMTOOL	
Write Command **AT+QCRMTOOL="IFACE_ID"[,<iface_ID>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "IFACE_ID",<iface_ID>** **OK** If the optional parameter is specified, set the terminal interface ID of QCRMTOOL: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<iface_ID>	Integer type. The terminal interface ID of QCRMTOOL. Range: 1–5. Default value: 4.

NOTE

As **<iface_ID>** is related to the module and host driver, it is recommended not to modify it and just leave it in its default value.

Example

```
AT+QCRMTOOL="IFACE_ID"          //Query the terminal interface ID of QCRMTOOL.
+QCRMTOOL: "IFACE_ID",4

OK
AT+QCRMTOOL="IFACE_ID",4        //Set the terminal interface ID of QCRMTOOL.
OK
```

2.3.1.10. AT+QCRMTOOL="MUX_ID" Set the Aggregation ID of QCRMTOOL

This command sets the aggregation ID of QCRMTOOL.

AT+QCRMTOOL="MUX_ID" Set the Aggregation ID of QCRMTOOL	
Write Command **AT+QCRMTOOL="MUX_ID"[,<MUX_ID>]**	Response If the optional parameter is omitted, query the current setting: **+QCRMTOOL: "MUX_ID",<MUX_ID>** **OK** If the optional parameter is specified, set the aggregation ID of QCRMTOOL: **OK** If there is any error: **ERROR**
Maximum Response Time	100 ms
Characteristics	This command takes effect after QCRMTOOL service restarts. The configurations will not be saved.

Parameter

<MUX_ID>	Integer type. The aggregation ID of QCRMTOOL. Range: 129–137. Default value: 129.

NOTE

As **<MUX_ID>** is related to the module and host driver, it is recommended not to modify it and just leave it in its default value.

Example

```
AT+QCRMTOOL="MUX_ID"                //Query the aggregation ID of QCRMTOOL.
+QCRMTOOL: "MUX_ID",129

OK
AT+QCRMTOOL="MUX_ID",129            //Set the aggregation ID of QCRMTOOL.
OK
```

2.4. URC Messages

2.4.1. URC Introduction

During its interaction with the host, the module not only responds to the request from the host system, but also actively reports to the host of external events, such as incoming calls and SMS. Usually, the information that the module actively reports is called URC (Unsolicited Result Code).

Generally, RG50xQ&RM5xxQ series modules and the host communicate only via the UART or USB interface, which means URCs from the module can only be reported via the UART or USB interface. You can use **AT+QURCCFG** to configure the URC reporting port of the module.

NOTE

For details about **AT+QURCCFG**, see ***Chapter 2.25*** of *Quectel_RG50xQ&RM5xxQ_Series_AT_Commands_Manual*.

2.4.2. URC Message Description

The URC of **AT+QCRMTOOL** is reported to the host in the format **+QCRMTOOL: <stage>,<code>,** indicating the status of QCRMTOOL service.

2.4.3. URC to Report QCRMTOOL Service Status

URC to Report QCRMTOOL Service Status	
+QCRMTOOL: <stage>,<code>	Reports the current status of QCRMTOOL service to the host

Parameter

<stage>	Integer type. QCRMTOOL service running stage. 1 Service initialization 2 Parameter setting 3 Port binding 4 Normal operation 5 Service monitoring 6 Service processing
<code>	Integer type. The running status of each stage of QCRMTOOL service. 0 Running normally < 0 An error occurs in the current service stage, and the service stops.

NOTE

After QCRMTOOL service starts, the URC will report at one go all the running status of the four stages from “service initialization” through “normal operation”, indicating this is the initialization stage of QCRMTOOL. At this time, if the **<code>** reported in URC is not 0, it means that an initialization fails and the service stops. Only after the initialization completes can the QCRMTOOL service start to run normally; only when there is an error occurs in service running will the URC **+QCRMTOOL: <stage>,<code>** be reported again.

Example

```
AT+QCRMTOOL="CTL",1          //Start QCRMTOOL service.
OK

+QCRMTOOL: 1,0               //QCRMTOOL service initialization runs normally.

+QCRMTOOL: 2,0               //QCRMTOOL parameter setting initialization runs normally.

+QCRMTOOL: 3,0               //QCRMTOOL port binding initialization runs normally.

+QCRMTOOL: 4,0               //QCRMTOOL normal operation initialization runs normally.

+QCRMTOOL: 4,-1              //An error occurs in normal operation stage, causing QCRMTOOL
                             service to stop.
```

3 AT Data Call Operation

3.1. Environmental Preparation

3.1.1. Environment Requirements

1. **AT+QCRMTOOL** is only applicable to Linux-USB environment.
2. Before using **AT+QCRMTOOL**, you need to install the USB qmi_wwan_q driver provided by Quectel, though there is no need to integrate the quectel-CM tool.
3. It is necessary to make modifications to the values of certain parameters in the qmi_wwan_q driver according to their values in QCRMTOOL. For details, see ***Chapter 3.1.2***.

NOTE

1. At present, **AT+QCRMTOOL** is only applicable to Linux-USB environment, not to PCIe environment. For Windows, **AT$QCRMCALL** can be used to make data calls directly.
2. To obtain the USB qmi_wwan_q driver, contact Quectel Technical Support.

3.1.2. qmi_wwan_q Driver Parameter Modification

If the parameter values in the driver are inconsistent with those in QCRMTOOL, the network cannot work normally. Therefore, before using **AT+QCRMTOOL**, please confirm that the parameter values in the installed driver are consistent with their counterparts in QCRMTOOL. It is recommended to modify the parameter values in the driver against the default parameter values in QCRMTOOL.

While modifying parameters in the qmi_wwan_q driver, the parameters in **AT+QCRMTOOL** that should be referred to are as follows:

<DLAGG_size>, **<ULAGG_grams>**, **<ULAGG_size>** and **<MUX_ID>**

The locations of these parameters in the driver's configuration file *qmi_wwan_q.c* and the rules for assigning values to them are as follows:

1. **<DLAGG_size>**

   As shown below, in *qmi_wwan_q.c*, the value 31 represents the size of DL packets aggregated (**<DLAGG_size>**). This value multiplied by 1024 should equal the value of **<DLAGG_size>**

configured in **AT+QCRMTOOL**. The default value is 31 × 1024 = 31744.

```
2234: static const struct driver_info qmi_wwan_raw_ip_info_sdx55 = {
2235:     qmi_wwan_raw_ip_info
2236:     .data = (9<<8) | 31, //QMAPV5 and 31KB
```

2. **<ULAGG_grams>** and **<ULAGG_size>**

   In *qmi_wwan_q.c*, the value 1 represents the number of UL packets aggregated (**<ULAGG_grams>**); the value 1500 represents the size of UL packets aggregated (**<ULAGG_size>**). In the driver, **<ULAGG_grams>** defaults to 1 and **<ULAGG_size>** defaults to 1500. But you need to modify these two parameter values to the default values configured in **AT+QCRMTOOL**, namely to 11 and 4096 respectively.

```
1976: #if defined(QUECTEL_UL_DATA_AGG)
1977:                 pQmapDev->tx_ctx.ul_data_aggregation_max_datagrams = 1;
1978:                 pQmapDev->tx_ctx.ul_data_aggregation_max_size = 1500;
1979: #endif
```

3. **<MUX_ID>**

   In *qmi_wwan_q.c*, the value 0x81 (129 in hexadecimal) represents the aggregation ID (**<MUX_ID>**). Converted to decimal, this value should equal the value of **<MUX_ID>** configured in **AT+QCRMTOOL**.

```
123:
124: #if defined(QUECTEL_WWAN_QMAP)
125: #define QUECTEL_QMAP_MUX_ID 0x81
126:
```

3.1.3. Data Call Status Rewriting

For RG50xQ&RM5xxQ series modules, after the driver is loaded, the network card device is in the unconnected status by default, which indicates it has not connected to the host. At this time, the host will not send any data (such as DHCP data) to the device. This is to prevent abnormity of the module caused by the host's sending data to the module when the module has not performed any data call.

In cases where the quectel-CM tool is integrated, it will keep the device connected after a successful data call. In other cases where quectel-CM is not used, Quectel provides another tool to the same purpose. The method to use this tool is as follows:

Enter */sys/class/net/wwan0/link_state* to open the driver device file and write **1\n** into the file to connect the driver device or **0\n** to disconnect the driver device. See the sample code below for details.

NOTE

The characters "wwan0" in the path */sys/class/net/wwan0/link_state* is a default vaule. If the name of the

device loaded by the host is not wwan0, please modify the value to the name of the device actually loaded.

The sample code to rewrite data call status is as follows:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
#include <fcntl.h>
#include <signal.h>
#include <errno.h>
#include <unistd.h>

void main(int argc, char *argv[])
{
char link_file[128];
int fd;
int new_state = 0;
if(argc == 1)
new_state = 1;
else if(argc == 2)
{
if(atoi(argv[1]) == 0)
new_state = 0;
else if(atoi(argv[1]) == 1)
new_state = 1;
else
return;
}
else
return;
printf("set ql_set_driver_link_state new_state=%d\r\n", new_state);
snprintf(link_file, sizeof(link_file), "/sys/class/net/wwan0/link_state");
fd = open(link_file, O_RDWR | O_NONBLOCK | O_NOCTTY);
if (fd == -1)
{
printf("Fail to access %s, errno: %d (%s)\r\n", link_file, errno, strerror(errno));
return;
}
snprintf(link_file, sizeof(link_file), "%d\n", new_state);
if (write(fd, link_file, sizeof(link_file)) == -1)
{
printf("set ql_set_driver_link_state fail\r\n");
```

```
}
close(fd);
}
```

3.2. AT Data Call Process

3.2.1. Brief Introduction to AT Data Call Process

1. Send **AT+QCRMTOOL="CTL",1** to start QCRMTOOL service.
2. Check URC **+QCRMTOOL: <stage>,<code>** to confirm whether the QCRMTOOL service is running normally.
3. Send **AT$QCRMCALL** to make a data call.
4. Use the data call status rewriting tool to modify the driver status to 1 (the connected status).
5. Execute **ifconfig wwan0 up && ifconfig wwan0_1 up** to start the network card (if the network card has already been enabled, it is recommended to execute **ifconfig wwan0 down && ifconfig wwan0_1 down** and then **ifconfig wwan0 up && ifconfig wwan0_1 up**).
6. Use a tool like udhcpc to send a DHCP request to the network card to obtain IP address and other information.

3.2.2. Matters Needing Attention

1. If the network disconnects due to reasons like poor network or the expiration of IP address, QCRMCALL will not reconnect on its own. In such a case, you need to re-make a data call after addressing specific network issues.
2. If a data call is made via IPv4v6 dual stack, when **AT$QCRMCALL=1,1,3,2,1** is executed to disconnect IPv4 and IPv6 simultaneously, the URC **+QCRMTOOL: <stage>,<code>** will only be reported once to notify the QCRMTOOL service status; if you disconnect IPv4 and IPv6 with **AT$QCRMCALL=1,1,1,2,1** and **AT$QCRMCALL=1,1,2,2,1** respectively, two URCs in the same format of **+QCRMTOOL: <stage>,<code>** will be reported to notify QCRMTOOL service status. It is recommended to disconnect IPv4 and IPv6 separately.

4 Appendix References

Table 3: Related Documents

Document Name
[1] Quectel_RG50xQ&RM5xxQ_Series_AT_Commands_Manual

Table 4: Terms and Abbreviations

Abbreviation	Description
AGG	Aggregation
DHCP	Dynamic Host Configuration Protocol
DL	Downlink
ID	Identifier
IP	Internet Protocol
OS	Operating System
PCIe	Peripheral Component Interconnect Express
QMI	Qualcomm MSM (Mobile Station Modem) Interface
UL	Uplink
URC	Unsolicited Result Code
USB	Universal Serial Bus
UART	Universal Asynchronous Receiver/Transmitter