QConnectManager

Linux User Guide

Version: 1.0

Date: 2022-03-16

Status: Released

At Quectel, our aim is to provide timely and comprehensive services to our customers. If you require any assistance, please contact our 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 offices. 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 us at: support@quectel.com.

Legal Notices

We offer information as a service to you. The provided information is based on your requirements and we make every effort to ensure its quality. You agree that you are responsible for using independent analysis and evaluation in designing intended products, and we provide reference designs for illustrative purposes only. Before using any hardware, software or service guided by this document, please read this notice carefully. Even though we employ commercially reasonable efforts to provide the best possible experience, you hereby acknowledge and agree that this document and related services hereunder are provided to you on an “as available” basis. We may revise or restate this document from time to time at our sole discretion without any prior notice to you.

Use and Disclosure Restrictions

License Agreements

Documents and information provided by us shall be kept confidential, unless specific permission is granted. They shall not be accessed or used for any purpose except as expressly provided herein.

Copyright

Our and third-party products hereunder may contain copyrighted material. Such copyrighted material shall not be copied, reproduced, distributed, merged, published, translated, or modified without prior written consent. We and the third party have exclusive rights over copyrighted material. No license shall be granted or conveyed under any patents, copyrights, trademarks, or service mark rights. To avoid ambiguities, purchasing in any form cannot be deemed as granting a license other than the normal non-exclusive, royalty-free license to use the material. We reserve the right to take legal action for noncompliance with abovementioned requirements, unauthorized use, or other illegal or malicious use of the material.

Trademarks

Except as otherwise set forth herein, nothing in this document shall be construed as conferring any rights to use any trademark, trade name or name, abbreviation, or counterfeit product thereof owned by Quectel or any third party in advertising, publicity, or other aspects.

Third-Party Rights

This document may refer to hardware, software and/or documentation owned by one or more third parties ("third-party materials"). Use of such third-party materials shall be governed by all restrictions and obligations applicable thereto.

We make no warranty or representation, either express or implied, regarding the third-party materials, including but not limited to any implied or statutory, warranties of merchantability or fitness for a particular purpose, quiet enjoyment, system integration, information accuracy, and non-infringement of any third-party intellectual property rights with regard to the licensed technology or use thereof. Nothing herein constitutes a representation or warranty by us to either develop, enhance, modify, distribute, market, sell, offer for sale, or otherwise maintain production of any our products or any other hardware, software, device, tool, information, or product. We moreover disclaim any and all warranties arising from the course of dealing or usage of trade.

Privacy Policy

To implement module functionality, certain device data are uploaded to Quectel's or third-party's servers, including carriers, chipset suppliers or customer-designated servers. Quectel, strictly abiding by the relevant laws and regulations, shall retain, use, disclose or otherwise process relevant data for the purpose of performing the service only or as permitted by applicable laws. Before data interaction with third parties, please be informed of their privacy and data security policy.

Disclaimer

a) We acknowledge no liability for any injury or damage arising from the reliance upon the information.
b) We shall bear no liability resulting from any inaccuracies or omissions, or from the use of the information contained herein.
c) While we have made every effort to ensure that the functions and features under development are free from errors, it is possible that they could contain errors, inaccuracies, and omissions. Unless otherwise provided by valid agreement, we make no warranties of any kind, either implied or express, and exclude all liability for any loss or damage suffered in connection with the use of features and functions under development, to the maximum extent permitted by law, regardless of whether such loss or damage may have been foreseeable.
d) We are not responsible for the accessibility, safety, accuracy, availability, legality, or completeness of information, advertising, commercial offers, products, services, and materials on third-party websites and third-party resources.

About the Document

Revision History

Version	Date	Author	Description
-	2021-12-03	Carl YIN	Creation of the document
1.0	2022-03-16	Carl YIN/ Ozzy ANG	First official release

Contents

Table Index

1 Introduction

QConnectManager is a Quectel data call tool for connecting to network. Multiple connection modes, drivers and advanced features are supported, as shown below:

Connection modes:

- QMI protocol
- MBIM protocol
- **AT+QNETDEVCTL**

Quectel drivers:

- qmi_wwan_q
- GobiNet
- pcie_mhi
- pcie_mhi_mbim

Linux drivers:

- qmi_wwan
- cdc_mbim
- cdc_ncm
- RNDIS
- ECM

Advanced features:

- Aggregate connection
- Multi-APN connection
- Bridge connection

QConnectManager automatically detects Quectel modules and the data call protocol in use. This document explains how to use QConnectManager, some common issues and their solutions.

1.1. Applicable Modules and Drivers

Table 1: Applicable Modules and Drivers

Driver	Protocol/ AT Command	Aggregate	Multiple	Bridge	Module Series	Applicable Module
qmi_wwan	QMI	-	-	-	LTE Standard	● AG35-CEN ● EC20-CE ● EC21 Series ● EC25 Series ● EG21-G ● EG25-G ● EG91 Series ● EG95 Series ● EM05 Series
qmi_wwan_q	QMI	√	√	√		
GobiNet	QMI	√	√	√		
cdc_mbim	MBIM	√	√	-	LTE-A	● EM06 Series ● EG06 Series ● EP06 Series ● EG060K-EA ● EG12 Series ● EM12-G ● EG18 Series ● EM160R-GL ● EM120R-GL ● EG512R-EA ● EM121R-GL ● EG065K Series ● EM120K-GL ● EM060K-GL
					Automotive	● AG35 Series ● AG520R Series ● AG521R Series
					5G	● RG500Q Series ● RG501Q-EU ● RG502Q Series ● RM500Q Series ● RM502Q-AE ● RM505Q-AE ● RM510Q-GL
pcie_mhi	QMI	√	√	√	LTE-A	● EM160R-GL ● EM120R-GL ● EG512R-EA
pcie_mhi_mbim	MBIM	√	√	√		

						● EM121R-GL
					5G	● RG500Q Series ● RG501Q-EU ● RG502Q Series ● RM500Q Series ● RM502Q-AE ● RM505Q-AE ● RM510Q-GL
cdc_ncm	**AT+QNETDEVCTL**	√	-	√		
RNDIS	**AT+QNETDEVCTL**	√	-	√	UMTS/HSPA+	● UC200T Series ● UG89
ECM	**AT+QNETDEVCTL**	-	-	√	LTE Standard	● EC200T Series ● EC200S Series ● EG912Y-EU ● EC200U Series

NOTE

1. The interface protocol of modules supporting USB interface can be configured into different protocols with **AT+QCFG="usbnet"**.
2. qmi_wwan, cdc_mbim, ECM, RNDIS, and cdc_ncm are Linux built-in drivers. Linux version 3.4 and above support qmi_wwan; Linux version 3.18 and above support cdc_mbim; Linux version 2.6 and above support the ECM/RNDIS/cdc_ncm.
3. qmi_wwan_q is the driver developed by Quectel based on qmi_wwan.
4. pcie_mhi and pcie_mhi_mbim use the same driver source codes. If *mhi_mbim_enabled* is set to *1*, the module uses MBIM; and if *mhi_mbim_enabled* is set to *0*, the module uses QMI.
5. "√" indicates supported and "-" indicates unsupported or NA.

2 Toolkit

QConnectManager kit contains reference logs, project files, source codes and other files. The files are shown in the following table:

Table 2: File List

Directory	Description
log	This directory stores the use logs of the QConnectManager. When connection with APN fails, you can compare logs to preliminarily troubleshoot the cause of the error.
Makefile	Project file and source codes, based on which Linux users can compile QConnectManager and generate executable programs.
.c and *.h* file	
Notice	Copyright notice.
ReleaseNote.txt	Release note.

3 Tool Parameter

QConnectManager supports setting multiple parameters. See ***Table 3*** for the details of each parameter.

Table 3: Parameter Description

No.	Parameter	Optional/ Compulsory	Description
1	*-s apn [user password auth]*	Optional	set *apn/user/password/auth* when establishing a connection, which must be obtained from the contracted operator. *auth* can be set to: *0* No authentication *1* PAP authentication *2* CHAP authentication
2	*-f logfile*	Optional	Save the tool logs to log file.
3	*-u usbmonlog*	Optional	Save usbmon log to the file specified by *-u*. Before using *-u usbmonlog*, make sure that your Linux OS has transplanted usbmon. usbmon is the USB monitor, which is used to monitor the data transmission on the USB bus. When you try connect over QMI/MBIM/AT command but the module does not respond, you can find the reason for that in this log.
4	*-v*	Optional	Print binary QMI and MBIM messages. When you try to connect over QMI/MBIM but the module does not respond or the connection fails, you can find the reason for that in this log.
5	*-4*	Optional	Request IPv4 connection.
6	*-6*	Optional	Whether to request IPv6 connection. *-4* Request only IPv4 connection *-6* Request only IPv6 connection *-4 -6* Request both IPv4 and IPv6 connection simultaneously Without *-4 -6* Request IPv4 connection by default
7	*-i iface*	Optional	When there are multiple Quectel modules in your product, the relevant module can be set with "*-i* network interface name".

8	*-n pdn*	Optional	Specify which PDN to use to establish a connection. Default: 1. ● When trying to establish a multi-APN connection over QMI/MBIM, you must set *-n pdn* to specify which PDN to use. ● *-n pdn* can be set if needed in case of a single connection. ● *n* must be set to 3 when using Verizon network.
9	*-k pdn*	Optional	This parameter disconnects the specified PDN configured with the *-n pdn* in case of a multi-APN connection over QMI/MBIM.
10	*-m iface-idx*	Optional	Default: *-n X*, which indicates binding this APN connection to the Xth network interface. In case of a multi-APN connection over QMI/MBIM, the APN connection is bound to the network interface specified by *-n* by default; Besides, in case of a multi-APN connection over QMI, the connection can also be bound to another network interface through this parameter.
11	*-b*	Optional	Enable bridge connection mode. This parameter only supports establishing a connection over QMI.
12	*-p pincode*	Optional	This parameter is used to verify (U)SIM PIN. It is used when the (U)SIM sets a PIN.
13	*-p proxy server*	Optional	Connect to proxy client program (QConnectManager will not directly use the QMI/MBIM device node), such as quectel-qmi-proxy, quectel-mbim-proxy, qmi-proxy, and mbim-proxy. Multiple QConnectManager examples must be run in case of a multi-APN connection over QMI/MBIM. But qmi_wwan_q/cdc_mbim/pcie_mhi restricts that only one program can read and write the QMI/MBIM device node at a time. Therefore, when running multiple QConnectManager examples, you must use a QMI/MBIM proxy client program to connect all QConnectManager examples to this proxy client program to read and write QMI/MBIM device nodes. QConnetManager automatically connects to quectel-qmi-proxy or quectel-mbim-proxy in case of a multi-APN connection over QMI/MBIM and no proxy client program need specified by this parameter.

4 Use Guidance

This chapter introduces how to establish a connection via QConnectManager in Linux OS.

4.1. Tool Preparation

Execute the following command to compile the tool source code and generate the executable program:

```
make CROSS_COMPILE=<User's Cross Compiler>
```

Compile three executable programs: quectel-CM, quectel-qmi-proxy, quectel-mbim-proxy. quectel-CM is the data call program; quectel-qmi-proxy and quectel-mbim-proxy are proxy client programs needed in case of a multi-APN connection.

NOTE

Both qmi-proxy and mbim-proxy are open-resource drivers and you can install them as needed.

4.2. Basic Use

Execute the following command to run quectel-CM:

```
./quectel-CM &
```

quectel-CM must keep running and cannot terminate. "&" at the end of the command is used to keep the program running in the background.

If you need to disconnect, run the following command to directly exit quectel-CM.

```
Killall quectel-CM
```

4.3. Single Connection

If only one PDN connection is needed, see ***Chapter 4.2*** for details.

The connection log of each driver:

- *log/cdc_mbim.txt*
- *log/ecm_ncm_rndis.txt*
- *log/gobinet.txt*
- *log/pcie_mhi_mbim.txt*
- *log/pcie_mhi_qmap=1.txt*
- *log/qmi_wwan_q.txt*
- *log/qmi_wwan.txt*

4.4. Aggregate Connection

Traditional connection carries only one IP network packet in a URB, while aggregate connection can carry multiple IP network packets in a URB. Aggregate connection can greatly decrease the number of USB transmissions thus reducing the CPU load of the system and improving the data throughput.

- Aggregate connection is recommended when using the LTE-A and 5G modules.
- Aggregate connection is recommended if the CPU frequency of the host is low and the throughput test fails to reach the theoretical rate when using the standard LTE modules.

After enabling aggregate connection according to the following table, you can try to establish a connection by following the steps in ***Chapter 4.2***.

Table 4: Aggregate Connection

Driver	Enablement
qmi_wwan_q	Set *qmap_mode* to *1*. *1* Enable *0* Disable Greater than *1* Multi-APN connection Default: *0*. When using the LTE-A and 5G modules, although *qmap_mode* is *0*, aggregate connection is automatically enabled. Log: *log/qmi_wwan_q_qmap=1.txt*
GobiNet	Set *qmap_mode* to *1*.

	1 Enable *0* Disable Greater than *1* Multi-APN connection Default: *0*. Log: *log/gobinet_qmap=1.txt*
pcie_mhi	Set *qmap_mode* to *1*. *1* Enable *0* Disable Greater than *1* Multi-APN connection Default: *1*. Log: *log/pcie_mhi_qmap=1.txt*
pcie_mhi_mbim	Set *qmap_mode* to *1*. *1* Enable *0* Disable Greater than *1* Multi-APN connection Default: *1*. Log: *log/pcie_mhi_mbim.txt*
cdc_mbim	No setting is needed and aggregate connection is automatically enabled. Log: *log/cdc_mbim.txt*
cdc_ncm	No setting is needed and aggregate connection is automatically enabled. Log: *log/ecm_ncm_rndis.txt*
RNDIS	No setting is needed and aggregate connection is automatically enabled. Log: *log/ecm_ncm_rndis.txt*

4.5. Multi-APN Connection

Multi-APN connection refers to creating multiple virtual network interfaces based on a physical one for multiple PDNs.

The steps for establishing a multi-APN connection using different drivers are as follows:

Table 5: Multi-APN Connection

Driver	Enablement
qmi_wwan_q	1. Set *qmap_mode* to the number of needed network interfaces, such as *4*. Maximum: *7*. 2. Run proxy client program **quectel-qmi-proxy -d /dev/cdc-wdm0** in the background. 3. Call quectel-CM to establish a connection. *-n* indicates specifying which PDN to connect to.

	Log: *log/qmi_wwan_q_qmap=4.txt*
GobiNet	1. Set *qmap_mode* to the number of needed network interfaces, such as *4*. Maximum: *7*. 2. Call quectel-CM to establish a connection. *-n* indicates specifying which PDN to connect to. Log: *log/gobinet_qmap=4.txt*
pcie_mhi	1. Set *qmap_mode* to the number of needed network interfaces, such as *4*. Maximum: *7*. 2. Run proxy client program **quectel-qmi-proxy -d /dev/mhi_QMI0** in the background. 3. Call quectel-CM to establish a connection. *-n* indicates specifying which PDN to connect to. Log: *log/pcie_mhi_qmap=4.txt*
pcie_mhi_mbim	1. Set *qmap_mode* to the number of needed network interfaces, such as *4*. Maximum: *7*. 2. Run proxy client program **quectel-mbim-proxy -d /dev/mhi_MBIM** in the background. 4. Run quectel-CM to establish a connection. *-n* indicates specifying which PDN to connect to. Log: *log/pcie_mhi_mbim_qmap=4.txt*
cdc_mbim	1. Execute the following command to create VLAN interface. *X* indicates which PDN. `# ip link add link wwan0 name wwan0.X type vlan id X` 3. Run proxy client program **quectel-mbim-proxy -d /dev/cdc-wdm0** in the background. 2. Call quectel-CM to establish a connection. *-n* indicates specifying which PDN to connect to. The value of *-n* is the same as *X* when creating the VLAN network interface. Log: *log/cdc_mbim_vlan.txt*

4.6. Bridge Connection

In traditional router application mode, PCs and mobile phones are connected to the LAN interface of the router through Wi-Fi or network cables. The router uses the module network interface as the WAN interface. The IP that the PC gets is a LAN IP starting with 192, and the router uses NAT to enable the PC to connect to the Internet through a WAN interface. In this application mode, port forwarding must be configured on the router if you need to use your PC as a TCP/UDP server.

Another way is to join the LAN interface and WAN interface into a bridge, so that LAN devices can communicate directly with the WAN interface.

Then, the LAN PC can directly obtain the IP address of the operator form WAN interface. To achieve the bridge connection, network interfaces to support ARP functionality. When ECM, RNIDS and cdc_ncm drivers are used, the generated network interface supports ARP functionality and can be directly added to the bridge. If you use GobiNet, pcie_mhi, qmi_wwan_q, and pcie_mhi_mbim drivers, the created network interface does not support ARP functionality. If you want to use bridge functionality, follow these steps:

1. Enable the macro **QUECTEL_BRIDGE_MODE** in the driver source codes.
   Source path: *Quectel_Linux_PCIE_MHI_Driver_V1.3.1.zip*/*pcie_mhi*/*devices*/*mhi_netdev_quectel.c*.
2. See ***Chapter 4.4*** and ***Chapter 4.5*** to enable the aggregate/multiple setting of the driver.
3. Run proxy client program quectel-qmi-proxy or quectel-mbim-proxy in the background based on different drivers when establishing a multi-APN connection.
4. Create a bridge and add the relevant network interface into the bridge.
   Only one LAN interface and one network interface of Quectel module can be added to a bridge; And the router can no longer call any DHCP program to get the IP address from the module.
5. Call quectel-CM to establish a connection. *-b* is required for establishing the connection, indicating that the bridge functionality is required.

The following log files show the steps for establishing a bridge connection by using the GobiNet, pcie_mhi, qmi_wwan_q, and pcie_mhi_mbim drivers:

- *log/gobinet_bridge.txt*
- *log/gobinet_qmap=1_bridge.txt*
- *log/gobinet_qmap=4_bridge.txt*
- *log/pcie_mhi_qmap=1_bridge.txt*
- *log/pcie_mhi_qmap=4_bridge.txt*
- *log/qmi_wwan_q_bridge.txt*
- *log/qmi_wwan_q_qmap=1_bridge.txt*
- *log/qmi_wwan_q_qmap=4_bridge.txt*
- *log/pcie_mhi_mbim_qmap=1_bridge.txt*
- *log/pcie_mhi_mbim_qmap=4_bridge.txt*

5 FAQ

1. Q: Why can't I access the Internet when I connect to PDN successfully?
   A: (1) Execute **ifconfig** to examine whether the network interface has an IP address.
   (2) Execute **ip ro show** to examine the routing table settings to ensure that there is only one default route and that it uses the module's network interface.

2. Q: Why can I Ping IP, but cannot Ping domain name?
   A: Execute **cat /etc/resolv.conf** to examine whether there is DNS and the DNS is obtained by establishing a connection via quectel-CM.

3. Q: Why does the udhcpc show that the IP address has been obtained, but the network interface does not have an IP address?
   A: After the udhcpc successfully obtains the IP address and DNS, it calls the *default.script* file, which is responsible for setting the IP address of the network interface and updating the DNS to */etc/resolv.conf*.

   You can execute the following command to see where the script is stored. Make sure that the script file exists and has the execute permission:

```
# busybox udhcpc -h
-s PROG          Run PROG at DHCP events (default /etc/udhcpc/default.script)
```

   If it does not already exist, you can copy the *default.script* in the quectel-CM source code directory to *default/etc/udhcpc/*. Note that the script should have execute permission.

4. Q: What logs should be provided when a connection fails?
   A: (1) First, execute the relevant AT commands to examine whether the module is successful in network registration: Execute **AT+COPS?** to get operator information and then execute **AT+CGREG?** (3G), **AT+CEREG?** (4G) or **AT+C5GREG?** (5G) to obtain the module network registration status.
   (2) Run quectel-CM with the parameter *-v -u usbmon_logfile*; Then provide the quectel-CM running log along with *usbmon_logfile*.
   (3) Use QLog to grab the log of the module. See ***document [1]*** for more information about QLog.

5. Q: How do I access the Internet through OpenWrt?
   A: In addition to running the quectel-CM on OpenWrt to establish a connection, you need to set the module network interface to the WAN interface. You can execute the following command to view the network interface configuration:

```
# uci show network.wan.ifname
network.wan.ifname='usb0'
```

Then execute **ifstatus wan** to view the WAN interface status.

6. Q: How do I select a PDN channel in case of a multi-APN connection?
   A: Some PDN channels have been used by internal applications in the module. You are not advised to use these PDN channels for establishing a connection. Generally speaking, the PDN channel name used internally contains characters such as "ims" and "SOS". You can execute the following command to view the PDN channel name.

```
# busybox microcom /dev/ttyUSB2
AT+CGDCONT?
+CGDCONT: 1,"IP","cmnet","0.0.0.0",0,0,0,0
+CGDCONT: 2,"IPV4V6","ims","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,0
+CGDCONT: 3,"IPV4V6","SOS","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,1
OK
```

You can select an unused PDN channel or a PDN channel that is not configured with APN to establish a connection.

For PDN channels which are not configured with APN, APN must be set with **AT+CGDCONT** or **AT+QICSGP** before establishing a connection using quectel-CM, otherwise the connection will fail.

NOTE

See the AT Command Manual and TCP(IP) Application Note of the corresponding module for more information about AT commands mentioned in this document.

6 Appendix References

Table 6: Related Document

Document Name
[1] Quectel_QLog_Linux&Android_User_Guide

Table 7: Terms and Abbreviations

Abbreviation	Description
APN	Access Point Name
ARP	Address Resolution Protocol
CDC	Communications Device Class
CHAP	Challenge-Handshake Authentication Protocol
CPU	Central Processing Unit
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Server
ECM	Ethernet Control Model
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
LAN	Local Area Network
LTE	Long-Term Evolution
MBIM	Mobile Broadband Interface Model
MHI	Modem Host Interface

NAT	Network Address Translation
NCM	Network Control Model
OS	Operation System
PAP	Password Authentication Protocol
PCIe	Peripheral Component Interconnect express
PDN	Public Data Network
PIN	Personal Identification Number
QMI	Qualcomm Message Interface
RNDIS	Remote Network Driver Interface Specification
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
URB	USB Request Block
USB	Universal Serial Bus
(U)SIM	(Universal) Subscriber Identity Module
VLAN	Virtual Local Area Network
WAN	Wide Area Network
WWAN	Wireless Wide Area Network