

# LTE Standard MQTT

## 应用指导

**LTE Standard 模块系列**

版本: LTE\_Standard\_MQTT\_应用指导\_V1.0

日期: 2020-04-22

状态: 受控文件

上海移远通信技术股份有限公司始终以为客户提供最及时、最全面的服务为宗旨。如需任何帮助，请随时联系我司上海总部，联系方式如下：

上海移远通信技术股份有限公司

上海市闵行区田林路 1016 号科技绿洲 3 期（B 区）5 号楼 邮编：200233

电话：+86 21 51086236 邮箱：[info@quectel.com](mailto:info@quectel.com)

或联系我司当地办事处，详情请登录：

<http://www.quectel.com/cn/support/sales.htm>

如需技术支持或反馈我司技术文档中的问题，可随时登陆如下网址：

<http://www.quectel.com/cn/support/technical.htm>

或发送邮件至：[support@quectel.com](mailto:support@quectel.com)

## 前言

上海移远通信技术股份有限公司提供该文档内容用以支持其客户的产品设计。客户须按照文档中提供的规范、参数来设计其产品。由于客户操作不当而造成的人身伤害或财产损失，本公司不承担任何责任。在未声明前，上海移远通信技术股份有限公司有权对该文档进行更新。

## 版权申明

本文档版权属于上海移远通信技术股份有限公司，任何人未经我司允许而复制转载该文档将承担法律责任。

版权所有 ©上海移远通信技术股份有限公司 2020，保留一切权利。

**Copyright © Quectel Wireless Solutions Co., Ltd. 2020.**

# 文档历史

## 修订记录

版本	日期	作者	变更表述
1.0	2020-04-22	陈驰/ 邓明/ 李柱	初始版本

## 目录

文档历史 .....	2
目录 .....	3
表格索引 .....	4
<b>1 引言 .....</b>	<b>5</b>
<b>2 MQTT 数据交互.....</b>	<b>6</b>
<b>3 MQTT 相关 AT 命令.....</b>	<b>7</b>
3.1. AT 命令语句.....	7
3.1.1. 定义 .....	7
3.1.2. AT 命令语句 .....	7
3.2. MQTT 相关 AT 命令描述 .....	8
3.2.1. AT+QMTCFG 配置 MQTT 可选参数.....	8
3.2.2. AT+QMTOPEN 打开 MQTT 客户端网络 .....	13
3.2.3. AT+QMTCLOSE 关闭 MQTT 客户端网络 .....	14
3.2.4. AT+QMTCONN 连接客户端到 MQTT 服务器.....	15
3.2.5. AT+QMTDISC 客户端断开与 MQTT 服务器的连接.....	16
3.2.6. AT+QMTSUB 订阅主题 .....	17
3.2.7. AT+QMTUNS 退订主题 .....	18
3.2.8. AT+QMTPUBEX 发布消息 .....	19
3.2.9. AT+QMTRECV 从缓存中读取消息 .....	20
<b>4 MQTT 相关 URC .....</b>	<b>22</b>
4.1. +QMTSTAT MQTT 链路层状态变化的 URC.....	22
4.2. +QMTRECV 通知 Host 读取 MQTT 数据包的 URC .....	23
4.3. +QMTPING 通知 MQTT 保活时间内 Ping 状态的 URC .....	24
<b>5 举例 .....</b>	<b>25</b>
5.1. 无 SSL 的 MQTT 操作举例.....	25
5.2. 含 SSL 的 MQTT 操作举例.....	27
<b>6 附录 A 参考文档和术语缩写.....</b>	<b>30</b>

表格索引

表 1: 适用模块 ..... 5

表 2: AT 命令及响应类型 ..... 7

表 3: MQTT 相关的 URC ..... 22

表 4: URC 中的错误码 ..... 23

表 5: 参考文档 ..... 30

表 6: 术语缩写 ..... 30

# 1 引言

MQTT 是基于代理的发布/订阅模式通讯协议，具有开放、简单、轻量和易于实现的特点。MQTT 最大优点在于，可以以极少的代码和有限的网络带宽，为连接远程设备提供实时可靠的消息服务。

该文档主要介绍如何通过 AT 命令使用移远通信 LTE Standard 模块的 MQTT 功能。

本文档主要适用于以下移远通信模块：

表 1：适用模块

模块系列	模块
EC2x 系列	EC20 R2.1
	EC21 系列
	EC25 系列
	EC20-CN
EG2x-G	EG21-G
	EG25-G
EG9x 系列	EG91 系列
	EG95 系列
EM05 系列	EM05 系列
EP200F-CE	EP200F-CE

## 备注

关于其他 LTE Standard 模块：EC100Y-CN、EG912Y-EU、EC200T 系列、EC200S-CN 的 MQTT 应用指导文档，请参考《Quectel\_EC100Y&EG912Y&EC200x 系列\_MQTT\_应用指导》。

# 2 MQTT 数据交互

本章节主要介绍 MQTT 功能的数据交互机制。

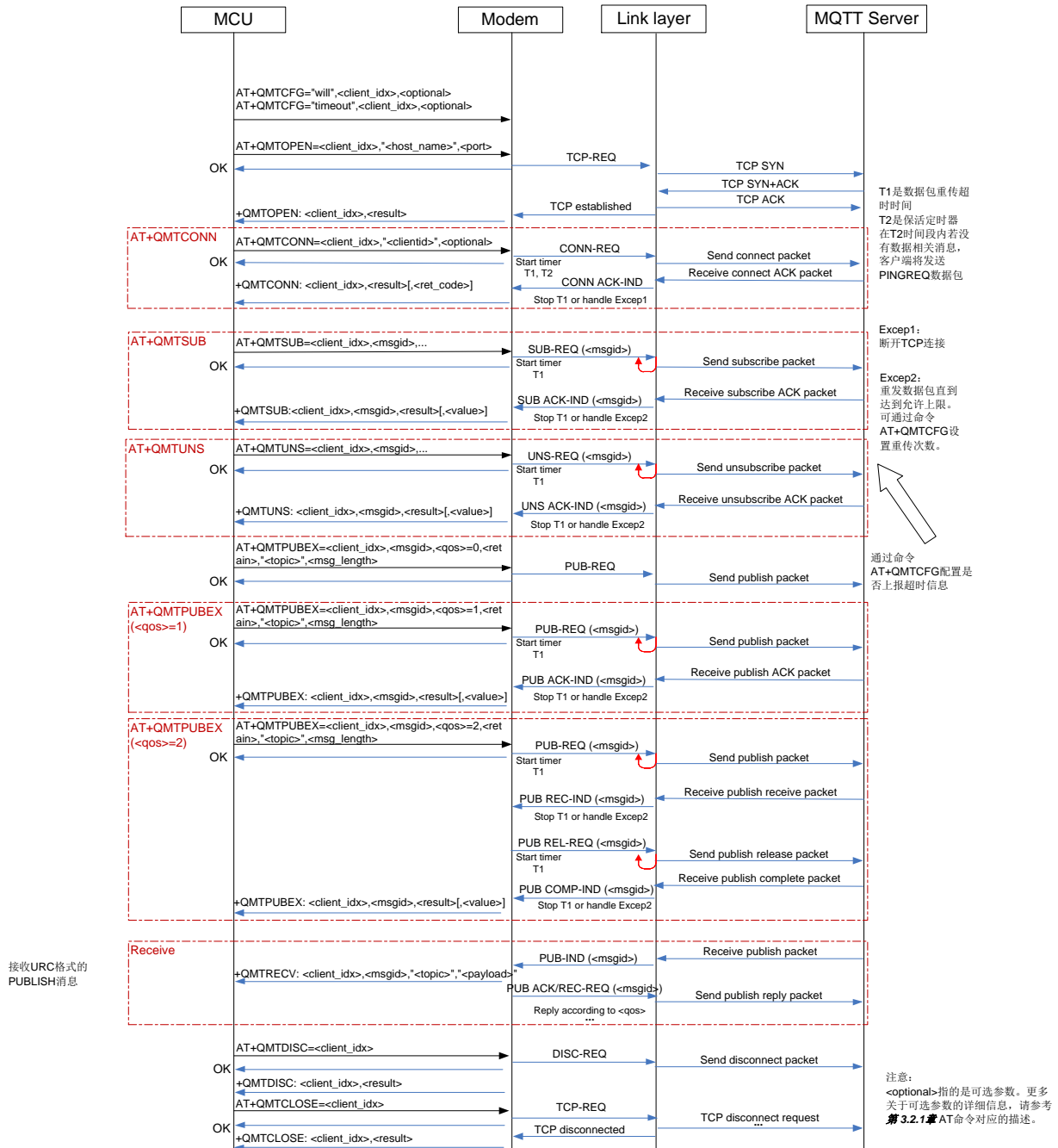


图 1: MQTT 数据交互机制

# 3 MQTT 相关 AT 命令

本章节主要介绍 MQTT 功能的相关 AT 命令。

## 3.1. AT 命令语句

### 3.1.1. 定义

- **<CR>** 回车符。
- **<LF>** 换行符。
- **<...>** 参数名称。实际命令行中不包含尖括号。
- **[...]** 可选参数或 TA 信息响应的可选部分。实际命令行中不包含方括号。若无特别说明，配置命令中的可选参数被省略时，将默认使用其之前已设置的值或其默认值。
- **下划线** 参数的默认设置。

### 3.1.2. AT 命令语句

前缀 **AT** 或 **at** 必须加在每个命令行的开头。输入 **<CR>** 将终止命令行。通常，命令后面跟随形式为 **<CR><LF><response><CR><LF>** 的响应。在本文档中，仅显示响应 **<response>**，省略 **<CR><LF>**。

表 2: AT 命令及响应类型

测试命令	<b>AT+&lt;cmd&gt;=?</b>	返回相应设置命令或内部程序可支持的参数取值列表或范围。
查询命令	<b>AT+&lt;cmd&gt;?</b>	返回相应设置命令的当前参数设置值。
设置命令	<b>AT+&lt;cmd&gt;=&lt;p1&gt;[,&lt;p2&gt;[,&lt;p3&gt;[...]]]</b>	设置用户可自定义的参数值。
执行命令	<b>AT+&lt;cmd&gt;</b>	主动执行内部程序实现的功能集。



## 3.2. MQTT 相关 AT 命令描述

### 3.2.1. AT+QMTCFG 配置 MQTT 可选参数

该命令用于配置 MQTT 的可选参数。

AT+QMTCFG 配置 MQTT 参数	
测试命令 <b>AT+QMTCFG=?</b>	响应 <b>+QMTCFG: "version",</b> (支持的<client_idx>范围),(支持的<vsns>列表) <b>+QMTCFG: "pdpcid",</b> (支持的<client_idx>范围),(支持的<cid>范围) <b>+QMTCFG: "ssl",</b> (支持的<client_idx>范围),(支持的<SSL_enable>列表),(支持的<SSL_ctx_idx>范围) <b>+QMTCFG: "keepalive",</b> (支持的<client_idx>范围),(支持的<keep_alive_time>范围) <b>+QMTCFG: "session",</b> (支持的<client_idx>范围),(支持的<clean_session>列表) <b>+QMTCFG: "timeout",</b> (支持的<client_idx>范围),(支持的<cpkt_timeout>范围),(支持的<retry_times>范围),(支持的<timeout_notice>列表) <b>+QMTCFG: "will",</b> (支持的<client_idx>范围),(支持的<will_fg>列表),(支持的<will_qos>范围),(支持的<will_retain>列表),"willtopic","willmessage" <b>+QMTCFG: "willex",</b> (支持的<client_idx>范围),(支持的<will_fg>列表),(支持的<will_qos>范围),(支持的<will_retain>列表),"willtopic", (支持的<will_len>范围) <b>+QMTCFG: "recv/mode",</b> (支持的<client_idx>范围),(支持的<msg_rcv_mode>列表),(支持的<msg_len_enable>列表) <b>+QMTCFG: "aliauth",</b> (支持的<client_idx>范围),"productkey","devicename","devicesecret" <b>+QMTCFG: "qmtping",</b> (支持的<client_idx>范围),(支持的<qmtping_interval>范围)  OK
设置命令 配置 MQTT 协议版本 <b>AT+QMTCFG="version",&lt;client_idx&gt;[,&lt;vsns&gt;]</b>	响应 若省略<vsns>, 则查询 MQTT 协议版本: <b>+QMTCFG: "version",&lt;vsns&gt;</b>  OK  若指定<vsns>且 MQTT 连接未创建, 配置 MQTT 协议版本: OK

	或者 <b>ERROR</b>
设置命令 配置 MQTT 客户端待使用的 PDP <b>AT+QMTCFG="pdpid",&lt;client_idx&gt;[,&lt;cid&gt;]</b>	响应 若省略<cid>, 则查询当前 MQTT 客户端使用的 PDP: <b>+QMTCFG: "pdpid",&lt;cid&gt;</b>  <b>OK</b>  若指定<cid>且 MQTT 连接未创建, 配置 MQTT 客户端使用的 PDP: <b>OK</b> 或者 <b>ERROR</b>
设置命令 配置 Will 信息 <b>AT+QMTCFG="will",&lt;client_idx&gt;[,&lt;will_fg&gt;[,&lt;will_qos&gt;,&lt;will_retain&gt;,&lt;will_topic&gt;",&lt;will_msg&gt;"]]</b>	响应 若省略<will_fg>、<will_qos>、<will_retain>、"<will_topic>"和"<will_msg>", 则查询当前 Will 配置信息: <b>+QMTCFG: "will",&lt;will_fg&gt;[,&lt;will_qos&gt;,&lt;will_retain&gt;,&lt;will_topic&gt;",&lt;will_msg&gt;"]</b>  <b>OK</b>  若指定<will_fg>、<will_qos>、<will_retain>、"<will_topic>"和"<will_msg>"且 MQTT 连接未创建, 配置 Will 信息: <b>OK</b> 或者 <b>ERROR</b>
设置命令 配置 Will 信息 <b>AT+QMTCFG="willex",&lt;client_idx&gt;[,&lt;will_fg&gt;,&lt;will_qos&gt;,&lt;will_retain&gt;,&lt;will_topic&gt;",&lt;will_len&gt;]</b>	响应 若省略<will_fg>、<will_qos>、<will_retain>、"<will_topic>"和<will_len>, 则查询当前 Will 配置信息: <b>+QMTCFG: "willex",&lt;will_fg&gt;,&lt;will_qos&gt;,&lt;will_retain&gt;,&lt;will_topic&gt;",&lt;will_len&gt;"]</b>  <b>OK</b>  若指定<will_fg>、<will_qos>、<will_retain>、"<will_topic>"和<will_len>, 配置 Will 信息: > 输入 Will 消息, 若数据长度大于<will_len>, 则发送前面长度等于<will_len>的数据。 <b>OK</b> 或者 <b>ERROR</b>
设置命令 配置消息传输超时时间	响应 若省略<pkt_timeout>、<retry_times>和<timeout_notice>, 则

<p>AT+QMTCFG="timeout",&lt;client_idx&gt;[,&lt;pkt_timeout&gt;,&lt;retry_times&gt;,&lt;timeout_notice&gt;]</p>	<p>查询当前设置的消息传输超时时间: +QMTCFG: "timeout",&lt;pkt_timeout&gt;,&lt;retry_times&gt;,&lt;timeout_notice&gt;</p> <p>OK</p> <p>若指定&lt;pkt_timeout&gt;、&lt;retry_times&gt;和&lt;timeout_notice&gt;且 MQTT 连接未创建, 配置消息传输的超时时间: OK 或者 ERROR</p>
<p>设置命令 配置会话类型 AT+QMTCFG="session",&lt;client_idx&gt;[,&lt;clean_session&gt;]</p>	<p>响应 若省略&lt;clean_session&gt;, 则查询当前会话类型: +QMTCFG: "session",&lt;clean_session&gt;</p> <p>OK</p> <p>若指定&lt;clean_session&gt;且 MQTT 连接未创建, 配置会话类型: OK 或者 ERROR</p>
<p>设置命令 配置保活时间 AT+QMTCFG="keepalive",&lt;client_idx&gt;[,&lt;keep_alive_time&gt;]</p>	<p>响应 若省略&lt;keep_alive_time&gt;, 则查询当前保活时间: +QMTCFG: "keepalive",&lt;keep_alive_time&gt;</p> <p>OK</p> <p>若指定&lt;keep_alive_time&gt;且 MQTT 连接未创建, 配置保活时间: OK 或者 ERROR</p>
<p>设置命令 配置 MQTT SSL 模式和 SSL 上下文索引 AT+QMTCFG="ssl",&lt;client_idx&gt;[,&lt;SSL_enable&gt;,&lt;SSL_ctx_idx&gt;]</p>	<p>响应 若省略&lt;SSL_enable&gt;和&lt;SSL_ctx_idx&gt;, 则查询当前 MQTT SSL 模式以及 SSL 上下文索引配置: +QMTCFG: "ssl",&lt;SSL_enable&gt;[,&lt;SSL_ctx_idx&gt;]</p> <p>OK</p> <p>若指定&lt;SSL_enable&gt;和&lt;SSL_ctx_idx&gt;且 MQTT 连接未创建, 配置 MQTT SSL 模式以及 SSL 上下文索引: OK 或者 ERROR</p>

<p>设置命令</p> <p>配置服务器数据的接收模式</p> <p><b>AT+QMTCFG="recv/mode",&lt;client_idx&gt;[,&lt;msg_recv_mode&gt;[,&lt;msg_len_enable&gt;]]</b></p>	<p>响应</p> <p>若省略&lt;msg_recv_mode&gt;和&lt;msg_len_enable&gt;，则查询当前 MQTT 数据接收模式：</p> <p><b>+QMTCFG: "recv/mode",&lt;msg_recv_mode&gt;,&lt;msg_len_enable&gt;</b></p> <p><b>OK</b></p> <p>若指定&lt;msg_recv_mode&gt;和&lt;msg_len_enable&gt;且 MQTT 连接未创建，配置数据接收模式：</p> <p><b>OK</b></p> <p>或者</p> <p><b>ERROR</b></p>
<p>设置命令</p> <p>配置阿里云设备信息</p> <p><b>AT+QMTCFG="aliauth",&lt;client_idx&gt;[,&lt;product_key&gt;",&lt;device_name&gt;",&lt;device_secret&gt;"]</b></p>	<p>响应</p> <p>若省略 "&lt;product_key&gt;" 、 "&lt;device_name&gt;" 和 "&lt;device_secret&gt;"，则查询设备信息，响应：</p> <p><b>+QMTCFG: "aliauth",&lt;product_key&gt;",&lt;device_name&gt;",&lt;device_secret&gt;"</b></p> <p><b>OK</b></p> <p>若指定 "&lt;product_key&gt;" 、 "&lt;device_name&gt;" 和 "&lt;device_secret&gt;"且 MQTT 连接未创建，配置 Alibaba 设备信息：</p> <p><b>OK</b></p> <p>或者</p> <p><b>ERROR</b></p>
<p>设置命令</p> <p>设置 MQTT 心跳间隔</p> <p><b>AT+QMTCFG="qmtping",&lt;client_idx&gt;[,&lt;qmtping_interval&gt;]</b></p>	<p>响应</p> <p>若省略&lt;qmtping_interval&gt;，则查询当前配置：</p> <p><b>+QMTCFG: "qmtping",&lt;qmtping_interval&gt;</b></p> <p><b>OK</b></p> <p>若指定&lt;qmtping_interval&gt;且 MQTT 连接未建立，配置 MQTT 心跳间隔：</p> <p><b>OK</b></p> <p>或者</p> <p><b>ERROR</b></p>
<p>最大响应时间</p>	<p>300 毫秒</p>
<p>特性说明</p>	<p>该命令立即生效；</p> <p>参数配置不保存。</p>

## 参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<vsn>	整型。MQTT 协议版本。 3 MQTT 协议 V3.1 4 MQTT 协议 V3.1.1
<cid>	整型。MQTT 客户端待使用 PDP。范围：1~16；默认值：1。
<will_fg>	整型。配置 Will flag。 0 无需配置 Will flag 1 需要配置 Will flag
<will_gos>	整型。发送消息时的 QoS 级别。 0 最多发送一次 1 至少发送一次 2 只发送一次
<will_retain>	整型。Will retain 标记只适用于 PUBLISH 消息。 0 当客户端发布一条 PUBLISH 消息到服务器，消息成功发送到当前订阅者后，服务器不保留该消息 1 当客户端发布一条 PUBLISH 消息到服务器，消息成功发送到当前订阅者后，服务器保留该消息
<will_topic>	字符串型。Will topic 名称。范围：1~256。
<will_msg>	字符串型。当客户端意外断开以后，该参数定义需要发布到 Will Topic 的消息内容。范围：0~256。
<will_len>	整型。Will 消息的长度。范围：0~256。
<pkt_timeout>	整型。数据包传输超时时间。范围：1~60；默认值：5；单位：秒。
<retry_times>	整型。数据包传输超时后重发次数。范围：0~10；默认值：3。
<timeout_notice>	整型。传输数据包时是否上报超时消息。 0 不上报 1 上报
<clean_session>	整型。配置会话类型。 0 客户端断开连接后，服务器必须存储该客户端的订阅消息 1 服务器必须删除之前保留的客户端任何消息，此连接状态为“clean”。
<keep_alive_time>	整型。保活时间。范围：0~3600；默认值：120；单位：秒。该参数定义从客户端接收消息的最大间隔时间，在 1.5 倍的设置时间内，若服务器未从客户端收到消息，则默认客户端发送了 DISCONNECT 消息，因此服务器会断开客户端连接。 0 表示不断开客户端连接
<SSL_enable>	整型。配置 MQTT SSL 模式。 0 使用普通 TCP 连接 1 使用 SSL TCP 安全连接
<SSL_ctx_idx>	整型。SSL 上下文索引。范围：0~5。
<msg_rcv_mode>	整型。配置 MQTT 消息接收模式。 0 从服务器接收的 MQTT 消息以 URC 的形式上报 1 从服务器接收的 MQTT 消息不以 URC 的形式上报
<msg_len_enable>	整型。配置 URC 中是否包含从服务器接收的 MQTT 消息长度。

	0 不包含
	1 包含
<product_key>	字符串型。阿里云发布的产品密钥。
<device_name>	字符串型。阿里云发布的设备名称。
<device_secret>	字符串型。阿里云发布的设备校验凭证。
<qmtping_interval>	整型。心跳间隔。范围：5~60；默认值：5；单位：秒。

## 备注

1. 若<will\_fg>=1，那么必须指定<will\_qos>，<will\_retain>，"<will\_topic>"和"<will\_msg>"，否则以上参数会被省略。
2. 只有当服务器支持存储会话操作时，<clean\_session>=0 才有效。
3. 若 MQTT 连接配置为 SSL 模式，必须指定<SSL\_ctx\_idx>。另外，客户需要使用 AT+QSSLCFG 配置 SSL 版本、加密套件、安全等级、CA 证书、客户端证书和客户端密钥，并忽略 RTC 时间，这些在 MQTT SSL 握手过程中都会使用到。
4. 请确保数据在传输过程中不能超时。
5. 命令 AT+QMTCFG="aliauth" 只用于阿里云，若已经配置，可以省略 AT+QMTCONN 中的 "<username>"和"<password>"。

## 3.2.2. AT+QMTOPEN 打开 MQTT 客户端网络

该命令用于为 MQTT 客户端打开网络。

AT+QMTOPEN 打开 MQTT 客户端网络	
测试命令 AT+QMTOPEN=?	响应 +QMTOPEN: (支持的<client_idx>范围),"<host_name>",(支持的<port>范围)  OK
查询命令 AT+QMTOPEN?	响应 [+QMTOPEN: <client_idx>,"<host_name>",<port>]  OK 或者 ERROR
设置命令 AT+QMTOPEN=<client_idx>,"<host_name>",<port>	响应 OK  +QMTOPEN: <client_idx>,<result> 或者 ERROR
最大响应时间	120 秒，受网络状态影响

特性说明	/
------	---

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<host_name>	字符串型。服务器地址，可以是 IP 地址或者域名。最大长度：100 字节。
<port>	整型。服务器端口。范围：1~65535。
<result>	整型。命令执行结果。 -1 打开网络失败 0 打开网络成功 1 参数错误 2 MQTT 标识符被占用 3 激活 PDP 失败 4 域名解析失败 5 网络断开导致错误

3.2.3. AT+QMTCCLOSE 关闭 MQTT 客户端网络

该命令用于关闭 MQTT 客户端网络。

AT+QMTCCLOSE 关闭 MQTT 客户端网络	
测试命令 AT+QMTCCLOSE=?	响应 +QMTCCLOSE: (支持的<client_idx>范围)  OK
设置命令 AT+QMTCCLOSE=<client_idx>	响应 OK  +QMTCCLOSE: <client_idx>,<result> 或者 ERROR
最大响应时间	30 秒
特性说明	/

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<result>	整型。命令执行结果。 -1 关闭网络失败 0 关闭网络成功

3.2.4. AT+QMTCONN 连接客户端到 MQTT 服务器

当客户端请求连接 MQTT 服务器时使用该命令。当客户端创建和服务端之间的 TCP/IP Socket 连接时，必须使用 CONNECT 流创建一个协议层会话。

AT+QMTCONN 连接客户端到 MQTT 服务器	
测试命令 AT+QMTCONN=?	响应 +QMTCONN: ( 支持的 <client_idx> 范围),"clientid","username","password"  OK
查询命令 AT+QMTCONN?	响应 [+QMTCONN: <client_idx>,<state>]  OK 或者 ERROR
设置命令 AT+QMTCONN=<client_idx>,"<clientid>","<username>","<password>"]	响应 OK  +QMTCONN: <client_idx>,<result>[,<ret_code>] 或者 ERROR
最大响应时间	参数<pkt_timeout>值（默认 5 秒），受网络状态影响
特性说明	/

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<clientid>	字符串型。客户端标识符。
<username>	字符串型。客户端用户名，可用于授权。
<password>	字符串型。客户端用户名对应的密码，可用于授权。
<result>	整型。命令执行结果。 0 数据包发送成功且接收到服务器的 ACK 1 数据包重传 2 数据包发送失败
<state>	整型。MQTT 连接状态。 1 MQTT 初始化 2 MQTT 正在连接 3 MQTT 已经连接成功



<ret_code>	4	MQTT 正在断开连接
	整型。连接状态返回码。	
	0	接受连接
	1	拒绝连接：不接受的协议版本
	2	拒绝连接：标识符被拒绝
	3	拒绝连接：服务器不可用
<pkt_timeout>	4	拒绝连接：错误的用户名或密码
	5	拒绝连接：未授权
	整型。数据包传输超时时间。范围：1~60；默认值：5；单位：秒。可通过 AT +QMTCFG="timeout",<client_idx>[,<pkt_timeout>,<retry_times,<timeout_notice>]配置超时时间。	

备注

若某一个客户端 ID 已经连接服务器端，另一个客户端使用相同 ID 重新连接，服务器必须断开原来的客户端连接，而后才能进行新客户端的 CONNECT 流。

3.2.5. AT+QMTDISC 客户端断开与 MQTT 服务器的连接

当客户端请求断开与 MQTT 服务器的连接时使用该命令。客户端发送 DISCONNECT 消息到服务器表示即将与服务器断开 TCP/IP 连接。

AT+QMTDISC MQTT 服务器断开与客户端连接	
测试命令 AT+QMTDISC=?	响应 +QMTDISC: (支持的<client_idx>范围)  OK
设置命令 AT+QMTDISC=<client_idx>	响应 OK  +QMTDISC: <client_idx>,<result> 或者 ERROR
最大响应时间	30 秒
特性说明	/

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<result>	整型。命令执行结果。

- 1 断开连接失败
- 0 断开连接成功

3.2.6. AT+QMTSUB 订阅主题

该命令用于订阅一个或多个主题。客户端发送 SUBSCRIBE 消息向服务器订阅一个或多个主题，当订阅主题发布消息时，服务器会把这些消息作为 PUBLISH 消息传输到客户端。

AT+QMTSUB 订阅主题	
测试命令 AT+QMTSUB=?	响应 +QMTSUB: ( 支持的 <client_idx> 范围 ),<msgid>,list of ["topic",qos]  OK
设置命令 AT+QMTSUB=<client_idx>,<msgid>,"<topic1>",<qos1>[,"<topic2>",<qos2>...]	响应 OK  +QMTSUB: <client_idx>,<msgid>,<result>[,<value>] 或者 ERROR
最大响应时间	<pkt_timeout> x <retry_times>（默认 15 秒），受网络状态影响
特性说明	/

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<msgid>	整型。数据包标识符。范围：1~65535。
<topic>	字符串型。客户端订阅或者退订的主题。
<qos>	整型。客户端发布消息的 QoS 等级。 0 最多发送一次 1 至少发送一次 2 只发送一次
<result>	整型。命令执行结果。 0 数据包发送成功且接收到服务器的 ACK 1 数据包重传 2 数据包发送失败
<value>	整型。 若<result>=0，则<value>作为已确认 QoS 等级的矢量； 若<result>=1，则表示数据包重传次数； 若<result>=2，则不显示。
<pkt_timeout>	整型。数据包传输超时时间。范围：1~60；默认值：5；单位：秒。可通过 AT+QMTC

	FG="timeout",<client_idx>[,<pkt_timeout>,<retry_times>,<timeout_notice>]进行配置超时时间。
<retry_times>	整型，数据包传输失败后重试次数，范围：0~10，默认值：3。

备注

只有在固定报头中的 QoS 位显示 QoS 等级是 1 或者 2 的消息中才显示参数<msgid>，在特定通信方向中的一组飞行消息中<msgid>必须唯一。一般来说该参数值根据消息数逐一增加，但实际情况无此要求。

3.2.7. AT+QMTUNS 退订主题

该命令用于退订一个或者多个主题。客户端向服务器发送 UNSUBSCRIBE 消息退订确定的主题。

AT+QMTUNS 退订主题	
测试命令 AT+QMTUNS=?	响应 +QMTUNS: ( 支持的 <client_idx> 范围 ),<msgid>,list of ["topic"]  OK
设置命令 AT+QMTUNS=<client_idx>,<msgid>, ,"<topic1>"["<topic2>"...]	响应 OK  +QMTUNS: <client_idx>,<msgid>,<result>[,<value>] 或者 ERROR
最大响应时间	<pkt_timeout> x <retry_times>（默认 15 秒），受网络状态影响
特性说明	/

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<msgid>	整型。数据包标识符。范围：1~65535。
<topic>	字符串型。客户端想要订阅或者退订的主题。
<result>	整型。命令执行结果。 0 数据包发送成功且接收到服务器的 ACK 1 数据包重传 2 数据包发送失败
<value>	整型。 若<result>=0，则<value>作为已确认 QoS 等级的矢量； 若<result>=1，则表示数据包重传次数；

	若<result>=2，则不显示。
<pkt_timeout>	数据包传输超时时间。范围：1~60；默认值：5；单位：秒。可通过 AT+QMTCFG="timeout",<client_idx>,<pkt_timeout>,<retry_times>,<timeout_notice>].进行配置超时时间。
<retry_times>	整型。数据包传输失败后重试次数。范围：0~10；默认值：3。

3.2.8. AT+QMTPUBEX 发布消息

客户端通过该命令可以发布定长消息到服务器，再由服务器分配到感兴趣的订阅者。每一个 PUBLISH 消息都关联一个主题名称，若一个客户端订阅了一个或者多个主题，当订阅主题发布消息时，服务器会把 这些消息作为 PUBLISH 消息传输到客户端。

AT+QMTPUBEX 发布消息	
测试命令 AT+QMTPUBEX=?	响应  +QMTPUBEX: (支持的<client_idx>范围),<msgid>,(支持的<qos>范围),(支持的<retain>列表),"topic","length"  OK
设置命令 AT+QMTPUBEX=<client_idx>,<msgid>,<qos>,<retain>,"<topic>",<msg_length>	响应 > 输入需要发送的数据，若数据实际长度大于<msg_length>，多出来的字节会被删除。 OK  +QMTPUBEX: <client_idx>,<msgid>,<result>[,<value>] 或者 ERROR
最大响应时间	<pkt_timeout> × <retry_times>（默认 15 秒），受网络状态影响
特性说明	/

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<msgid>	整型。数据包消息标识符。范围：0~65535；只有当<qos>=0 时；该参数值为 0。
<qos>	整型。客户端想要发布消息时的 QoS 等级。 0 最多发送一次 1 最少发送一次 2 只发送一次
<retain>	整型。消息发送到当前订阅者后，服务器是否保存该消息。 0 不保存 1 保存

<topic>	字符串型。待发布主题。
<msg_length>	整型。待发布消息数据长度。
<result>	整型。命令执行结果。 0 数据包发送成功且接收到服务器的 ACK（当<qos>=0 时发布了数据，则无需 ACK） 1 数据包重传 2 数据包发送失败
<value>	整型。 若<result>=1，则表示数据包重传次数； 若<result>=0 或者 2，则不显示此参数。
<pkt_timeout>	整型。数据传输超时时间。范围：1~60；默认值：5；单位：秒。可通过 <b>AT+QMTCFG="timeout",&lt;client_idx&gt;,&lt;pkt_timeout&gt;,&lt;retry_times&gt;,&lt;timeout_notice&gt;]</b> 进行配置超时时间。
<retry_times>	整型，数据包传输失败后重试次数，范围：0~10，默认值：3。

## 备注

1. 若该命令执行成功并响应 **OK**，客户端可以继续发布新的数据包。传输数据包的最大数量不能大于滑动窗口的大小（滑动窗口的大小为 5），否则会返回 **ERROR**。
2. 该命令执行后，客户端可发送数据即负载信息。每次输入数据的最大长度为 1500 字节。
3. 发布者可以发布 PUBLISH 消息到服务器，服务器也可以发布 PUBLISH 消息到订阅者。当服务器发布消息到订阅者时，会返回 URC 通知 host 读取 MQTT 服务器发送的数据：**+QMTRECV:<client\_idx>,<msgid>,<topic>[,<payload\_length>],<payload>**”，关于 URC 描述的详细信息，请参考第 4.2 章。

### 3.2.9. AT+QMTRECV 从缓存中读取消息

该命令用于从存储缓存中读取消息，服务器上报消息时都会存储到这个缓存中。

AT+QMTRECV 从缓存中读取消息	
测试命令 <b>AT+QMTRECV=?</b>	响应 <b>OK</b>
查询命令 <b>AT+QMTRECV?</b>	响应 <b>+QMTRECV: &lt;client_idx&gt;,&lt;store_status_0&gt;,&lt;store_status_1&gt;,&lt;store_status_2&gt;,&lt;store_status_3&gt;,&lt;store_status_4&gt;</b>  <b>OK</b>  若无 MQTT 连接： <b>OK</b>
设置命令 <b>AT+QMTRECV=&lt;client_idx&gt;,&lt;recv_i</b>	响应 <b>[(+QMTRECV: &lt;client_idx&gt;,&lt;msgid&gt;,&lt;topic&gt;[,&lt;payload</b>

d>]	<div><div>_len&gt;], "&lt;payload&gt;"列表)]</div><div>OK</div><div>若无 MQTT 连接:</div><div>ERROR</div></div>
特性说明	/

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<store_status>	整型。表示缓存中是否存有消息，缓存中最多可存储 5 条信息，因此最多同时上报 5 条 URC。 0 缓存中没有数据 1 缓存中有数据
<recv_id>	整型。表示每条接收数据的序号。范围：0~4。
<msgid>	整型。数据包消息标识符。范围：0~65535，只有当<gos>=0 时，此参数可为 0。
<topic>	字符串型。待发布的主题。
<payload_len>	整型。负载信息的长度。
<payload>	字符串型。主题名相关的负载信息。

# 4 MQTT 相关 URC

本章节主要描述 MQTT 相关的 URC。

表 3: MQTT 相关的 URC

序号	URC 格式	描述
[1]	<b>+QMTSTAT:</b> <client_idx>,<err_code>	当 MQTT 链路层状态改变，客户端会断开 MQTT 连接并上报 URC。
[2]	<b>+QMTRECV:</b> <client_idx>,<msgid>,"<topic> "[,<payload_len>], "<payload>"	当客户端接收到 MQTT 服务器的数据包会上报 URC。
[3]	<b>+QMTRECV:</b> <client_idx>,<recv_id>	当从 MQTT 服务器接收的消息存储到缓存时上报 URC。
[4]	<b>+QMTPING:</b> <client_idx>,<result>	当 MQTT 链层状态变化时，客户端会关闭 MQTT 连接并上报此 URC。

## 4.1. +QMTSTAT MQTT 链路层状态变化的 URC

此 URC 以**+QMTSTAT:**开头，当 MQTT 链路层状态发生变化就会上报该 URC。

<b>+QMTSTAT MQTT 链路层状态变化的 URC</b>	
<b>+QMTSTAT:</b> <client_idx>,<err_code>	当 MQTT 链路层状态改变，客户端会断开 MQTT 连接并上报 URC。

### 参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<err_code>	整型。错误代码。详细信息请参考表 4。

表 4：URC 中的错误码

<err_code>	描述	解决方法
1	连接被服务器断开或者重置	执行命令 <b>AT+QMTOPEN</b> 重建 MQTT 连接。
2	发送 PINGREQ 包超时或者失败	首先反激活 PDP，然后再激活 PDP 并重建 MQTT 连接。
3	发送 CONNECT 包超时或者失败	1. 查看输入的用户名和密码是否正确。 2. 确保客户端 ID 未被占用。 3. 重建 MQTT 连接，并尝试再次发送 CONNECT 包到服务器。
4	接收 CONNACK 包超时或者失败	1. 查看输入的用户名和密码是否正确。 2. 确保客户端 ID 未被占用。 3. 重建 MQTT 连接，并尝试再次发送 CONNECT 包到服务器。
5	客户端向服务器发送 DISCONNECT 包，但是服务器主动断开 MQTT 连接	正常流程。
6	因为发送数据包总是失败，客户端主动断开 MQTT 连接	1. 确保数据正确。 2. 可能因为网络拥堵或者其他错误，尝试重建 MQTT 连接。
7	链路不工作或者服务器不可用	确保当前链路或者服务器可用。
8	客户端主动断开 MQTT 连接	尝试重建连接。
9~255	留作将来使用	

4.2. +QMTRECV 通知 Host 读取 MQTT 数据包的 URC

此 URC 以+QMTRECV:开头，主要用于通知 Host 读取 MQTT 服务器发送的数据包。

+QMTRECV 通知 Host 读取 MQTT 数据包的 URC	
+QMTRECV: <client_idx>,<msgid>,"<topic>"[,<payload_len>],"<payload>"	通知 Host 读取 MQTT 服务器发送的数据包。
+QMTRECV: <client_idx>,<recv_id>	当从 MQTT 服务器接收的消息存储到缓存时上报 URC。

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<msgid>	整型。数据包的消息标识符。



<topic>	字符串型。从 MQTT 服务器接收的主题。
<payload_len>	整型。负载信息的长度。
<payload>	字符串型。主题名称相关的负载。
<recv_id>	整型。表示每条接收数据的序号。范围：0~4。

4.3. +QMTPING 通知 MQTT 保活时间内 Ping 状态的 URC

此 URC 以+QMTPING:开头，若服务器未在 1.5 倍保活时间内接收到客户端的消息，则相当于客户端发送了 DISCONNECT 消息，服务器会断开与客户端的连接，此时会上报该 URC。

+QMTPING 通知 MQTT 保活时间内 Ping 状态的 URC	
+QMTPING: <client_idx>,<result>	当 MQTT 链层状态变化时，客户端会关闭 MQTT 连接并上报此 URC。

参数

<client_idx>	整型。MQTT 客户端标识符。范围：0~5。
<result>	整型。Ping 状态的结果。 1 失败

# 5 举例

本章节主要提供举例用于解释如何使用 MQTT 相关命令。

## 5.1. 无 SSL 的 MQTT 操作举例

```
//配置接收模式
AT+QMTCFG="recv/mode",0,0,1
OK

//配置阿里云设备信息
AT+QMTCFG="aliauth",0,"oyjtmPI5a5j","MQTT_TEST","wN9Y6pZSIly7Exa5qVzcmigEGO4kAazZ"
OK
AT+QMTOPEN=?
+QMTOPEN: (0-5),"host_name",(1-65535)

OK

//MQTT 客户端打开网络
AT+QMTOPEN=0,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883
OK

+QMTOPEN: 0,0 //MQTT 客户端成功打开网络
AT+QMTOPEN?
+QMTOPEN: 0,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883

OK
AT+QMTCONN=?
+QMTCONN: (0-5),"clientid","username","password"

OK

//客户端连接 MQTT 服务器
//若已连接阿里云，可使用命令 AT+QMTCFG="aliauth" 提前配置设备信息，之后可缺省参数
"<username>" 和 "<password>"。
AT+QMTCONN=0,"clientExample"
OK
```

```

+QMTCONN: 0,0,0                                //客户端成功连接到 MQTT 服务器
AT+QMTSUB=?
+QMTSUB: (0-5),<msgid>, list of ["topic",qos]

OK

//订阅主题
AT+QMTSUB=0,1,"topic/example",2
OK

+QMTSUB: 0,1,0,2
AT+QMTSUB=0,1,"topic/pub",0
OK

+QMTSUB: 0,1,0,0

//如果客户端订阅某个主题，而其他设备将同一主题发布到服务器，模块会上报如下信息：
+QMTRECV: 0,0, "topic/example",36,"This is the payload related to topic"

//退订主题
AT+QMTUNS=0,2,"topic/example"
OK

+QMTUNS: 0,2,0
AT+QMTPUBEX=?
+QMTPUBEX: (0-5),<msgid>,(0-2),(0,1),"topic","length"

OK

//响应>后，输入并发送 This is test data, hello MQTT.，数据最大长度是 1500 字节，多出的部分会被忽略。
AT+QMTPUBEX=0,0,0,0,"topic/pub",30
>This is test data, hello MQTT.
OK

+QMTPUBEX: 0,0,0

//若客户端订阅了名为"topic/pub"的主题，且其他设备向服务器发布了相同主题，模块会上报如下信息：
+QMTRECV: 0,0,"topic/pub",30,This is test data, hello MQTT.

//断开客户端在 MQTT 服务器的连接
AT+QMTDISC=0
OK
    
```

+QMTDISC: 0,0

//连接断开成功

## 5.2. 含 SSL 的 MQTT 操作举例

//配置接收模式

**AT+QMTCFG="recv/mode",0,0,1**

OK

//配置 MQTT 会话为 SSL 模式

**AT+QMTCFG="ssl",0,1,2**

OK

//若 SSL 授权方式为 server authentication，则存储 CA 证书到 RAM。

**AT+QFUPL="RAM:cacert.pem",1758,100**

CONNECT

<Input the cacert.pem data, the size is 1758 bytes>

+QFUPL: 1758,384a

OK

//若 SSL 授权方式为 server authentication，则存储 CC 证书到 RAM。

**AT+QFUPL="RAM:client.pem",1220,100**

CONNECT

<Input the client.pem data, the size is 1220 bytes>

+QFUPL: 1220,2d53

OK

//若 SSL 授权方式为 server authentication，则存储 CK 证书到 RAM。

**AT+QFUPL="RAM:user\_key.pem",1679,100**

CONNECT

<Input the user\_key.pem data, the size is 1679 bytes>

+QFUPL: 1679,335f

OK

//配置 CA 证书

**AT+QSSLCFG="cacert",2,"RAM:cacert.pem"**

OK

//配置 CC 证书

**AT+QSSLCFG="clientcert",2,"RAM:client.pem"**

OK

//配置 CK 证书

**AT+QSSLCFG="clientkey",2,"RAM:user\_key.pem"**

OK

//配置 SSL 参数

**AT+QSSLCFG="seclevel",2,2** //SSL 授权方式: server authentication

OK

**AT+QSSLCFG="sslversion",2,4** //SSL 授权版本

OK

**AT+QSSLCFG="ciphersuite",2,0xFFFF** //加密套件

OK

**AT+QSSLCFG="ignorelocaltime",2,1** //忽略授权时间

OK

//开始 MQTT SSL 连接

**AT+QMTOPEN=0,"a1zgnxur10j8ux.iot.us-east-1.amazonaws.com",8883**

OK

**+QMTOPEN: 0,0**

//连接 MQTT 服务器

**AT+QMTCONN=0,"M26\_0206"**

OK

**+QMTCONN: 0,0,0**

//订阅主题

**AT+QMTSUB=0,1,"\$aws/things/M26\_0206/shadow/update/accepted",1**

OK

**+QMTSUB: 0,1,0,1**

//发布消息

**AT+QMTPUBEX=0,1,1,0,"\$aws/things/M26\_0206/shadow/update/accepted",32**

**>This is publish data from client**

OK

**+QMTPUBEX: 0,1,0**

//若客户端订阅了名为"\$aws/things/M26\_0206/shadow/update/accepted"的主题，且其他设备发布了相同主题的消息到服务器，模块会上报如下信息：

**+QMTRECV: 0,1,"\$aws/things/M26\_0206/shadow/update/accepted",32,"This is publish data from**

client"

//客户端断开 MQTT 服务器

**AT+QMTDISC=0**

OK

+QMTDISC: 0,0

## 6 附录 A 参考文档和术语缩写

表 5: 参考文档

序号	文档名称	说明
[1]	MQTT V3.1 Protocol Specification	MQTT 协议规格说明 3.1 版本
[2]	MQTT V3.1.1 Protocol Specification	MQTT 协议规格说明 3.1.1 版本

表 6: 术语缩写

缩写	英文全称	中文全称
ACK	Acknowledgement	确认消息
MQTT	Message Queuing Telemetry Transport	消息队列遥测传输
QoS	Quality of Service	服务质量
RAM	Random Access Memory	随机存取存储器
SSL	Secure Sockets Layer	安全套接层协议
TCP	Transmission Control Protocol	传输控制协议
URC	Unsolicited Result Code	非请求结果码