

# RG50xQ&RM5xxQ 系列

## TCP/IP 应用指导

**5G 模块系列**

版本：1.0

日期：2022-05-05

状态：受控文件



上海移远通信技术股份有限公司（以下简称“移远通信”）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 5108 6236 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，请随时登陆网址：

<http://www.quectel.com/cn/support/technical.htm> 或发送邮件至：[support@quectel.com](mailto:support@quectel.com)。

## 前言

移远通信提供该文档内容以支持客户的产品设计。客户须按照文档中提供的规范、参数来设计产品。同时，您理解并同意，移远通信提供的参考设计仅作为示例。您同意在设计您目标产品时使用您独立的分析、评估和判断。在使用本文档所指导的任何硬软件或服务之前，请仔细阅读本声明。您在此承认并同意，尽管移远通信采取了商业范围内的合理努力来提供尽可能好的体验，但本文档和其所涉及服务是在“可用”基础上提供给您的。移远通信可在未事先通知的情况下，自行决定随时增加、修改或重述本文档。

## 使用和披露限制

### 许可协议

除非移远通信特别授权，否则我司所提供硬软件、材料和文档的接收方须对接收的内容保密，不得将其用于除本项目的实施与开展以外的任何其他目的。

### 版权声明

移远通信产品和本协议项下的第三方产品可能包含受移远通信或第三方材料、硬软件和文档版权保护的相关资料。除非事先得到书面同意，否则您不得获取、使用、向第三方披露我司所提供的文档和信息，或对此类受版权保护的资料进行复制、转载、抄袭、出版、展示、翻译、分发、合并、修改，或创造其衍生作品。移远通信或第三方对受版权保护的资料拥有专有权，不授予或转让任何专利、版权、商标或服务商标权的许可。为避免歧义，除了正常的非独家、免版税的产品使用许可，任何形式的购买都不可被视为授予许可。对于任何违反保密义务、未经授权使用或以其他非法形式恶意使用所述文档和信息的违法侵权行为，移远通信有权追究法律责任。

### 商标

除另行规定，本文档中的任何内容均不授予在广告、宣传或其他方面使用移远通信或第三方的任何商标、商号及名称，或其缩略语，或其仿冒品的权利。

### 第三方权利

您理解本文档可能涉及一个或多个属于第三方的硬软件和文档（“第三方材料”）。您对此类第三方材料的使用应受本文档的所有限制和义务约束。

移远通信针对第三方材料不做任何明示或暗示的保证或陈述，包括但不限于任何暗示或法定的适销性或特定用途的适用性、平静受益权、系统集成、信息准确性以及与许可技术或被许可人使用许可技术相关的不侵犯任何第三方知识产权的保证。本协议中的任何内容都不构成移远通信对任何移远通信产品或任何其他硬件、设备、工具、信息或产品的开发、增强、修改、分销、营销、销售、提供销售或以其他方式维持生产的陈述或保证。此外，移远通信免除因交易过程、使用或贸易而产生的任何和所有保证。

## 隐私声明

为实现移远通信产品功能，特定设备数据将会上传至移远通信或第三方服务器（包括运营商、芯片供应商或您指定的服务器）。移远通信严格遵守相关法律法规，仅为实现产品功能之目的或在适用法律允许的情况下保留、使用、披露或以其他方式处理相关数据。当您与第三方进行数据交互前，请自行了解其隐私保护和数据安全政策。

## 免责声明

- 1) 移远通信不承担任何因未能遵守有关操作或设计规范而造成损害的责任。
- 2) 移远通信不承担因本文档中的任何因不准确、遗漏、或使用本文档中的信息而产生的任何责任。
- 3) 移远通信尽力确保开发中功能的完整性、准确性、及时性，但不排除上述功能错误或遗漏的可能。除非另有协议规定，否则移远通信对开发中功能的使用不做任何暗示或法定的保证。在适用法律允许的最大范围内，移远通信不对任何因使用开发中功能而遭受的损害承担责任，无论此类损害是否可以预见。
- 4) 移远通信对第三方网站及第三方资源的信息、内容、广告、商业报价、产品、服务和材料的可访问性、安全性、准确性、可用性、合法性和完整性不承担任何法律责任。

版权所有 ©上海移远通信技术股份有限公司 2022，保留一切权利。

**Copyright © Quectel Wireless Solutions Co., Ltd. 2022.**

# 文档历史

## 修订记录

版本	日期	作者	变更表述
-	2022-03-31	Ozzy ANG/ Demon YANG	文档创建
1.0	2022-05-05	Ozzy ANG/ Demon YANG	受控版本

## 目录

文档历史 .....	3
目录 .....	4
表格索引 .....	6
<b>1 引言 .....</b>	<b>7</b>
1.1. 适用模块 .....	7
1.2. TCP/IP AT 命令使用流程 .....	7
1.3. 数据访问模式说明 .....	9
<b>2 TCP/IP AT 命令详解 .....</b>	<b>10</b>
2.1. AT 命令说明 .....	10
2.1.1. 定义 .....	10
2.1.2. AT 命令语句 .....	10
2.2. AT 示例声明 .....	11
2.3. AT 命令详解 .....	11
2.3.1. AT+QICSGP 配置 TCP/IP 上下文参数 .....	11
2.3.2. AT+QIACT 激活 PDP 上下文 .....	12
2.3.3. AT+QIDEACT 去激活 PDP 上下文 .....	13
2.3.4. AT+QIOPEN 打开 Socket 服务 .....	14
2.3.5. AT+QICLOSE 关闭 Socket 服务 .....	15
2.3.6. AT+QISTATE 查询 Socket 服务状态 .....	16
2.3.7. AT+QISEND 发送数据 .....	18
2.3.8. AT+QIRD 读取接收的 TCP/IP 数据 .....	20
2.3.9. AT+QISENDEX 发送 16 进制字符串数据 .....	22
2.3.10. AT+QISWTMD 切换数据访问模式 .....	22
2.3.11. AT+QPING 进行远程服务器 Ping 操作 .....	23
2.3.12. AT+QISDE 控制是否回显 AT+QISEND 要发送的数据 .....	24
2.3.13. AT+QIGETERROR 查询上一个 AT 命令错误代码 .....	25
2.4. URC 说明 .....	26
2.4.1. +QIURC: "closed" 连接断开通知 .....	26
2.4.2. +QIURC: "recv" 数据接收通知 .....	26
2.4.3. +QIURC: "incoming full" 客户端连接已满通知 .....	27
2.4.4. +QIURC: "incoming" 客户端连接通知 .....	27
2.4.5. +QIURC: "pdpdeact" PDP 去激活通知 .....	28
<b>3 示例 .....</b>	<b>29</b>
3.1. 上下文配置和激活 .....	29
3.1.1. 上下文配置 .....	29
3.1.2. 上下文激活 .....	29
3.1.3. 上下文去激活 .....	29
3.2. TCP 客户端在缓存模式下工作 .....	29
3.2.1. 创建 TCP 客户端连接并进入缓存模式 .....	29
3.2.2. 缓存模式下发送数据 .....	30

3.2.3.	缓存模式下从远程服务器接收数据 .....	30
3.2.4.	断开连接 .....	31
3.3.	TCP 客户端在透传模式下工作 .....	31
3.3.1.	创建 TCP 客户端连接并进入透传模式 .....	31
3.3.2.	透传模式下发送数据 .....	31
3.3.3.	透传模式接收远程器端数据 .....	31
3.3.4.	断开 TCP 客户端连接 .....	32
3.4.	TCP 客户端在直吐模式下工作 .....	32
3.4.1.	创建 TCP 客户端连接并进入直吐模式 .....	32
3.4.2.	直吐模式下发送数据 .....	32
3.4.3.	直吐模式下接收远程服务器端数据 .....	33
3.4.4.	断开 TCP 用户端连接 .....	33
3.5.	TCP 服务器在缓存模式下工作 .....	33
3.5.1.	启动 TCP 服务器 .....	33
3.5.2.	接受客户端的连接请求 .....	33
3.5.3.	接收的客户端数据 .....	33
3.5.4.	断开 TCP 服务器连接 .....	34
3.6.	UDP 服务示例 .....	34
3.6.1.	打开 UDP 服务 .....	34
3.6.2.	向服务器发送 UDP 数据 .....	34
3.6.3.	接收远程端数据 .....	35
3.6.4.	关闭 UDP 服务 .....	35
3.7.	PING.....	35
3.8.	获取上一个错误代码示例 .....	36
4	错误代码.....	37
5	附录 参考文档及术语缩写 .....	39

## 表格索引

表 1: 适用模块 .....	7
表 2: AT 命令类型 .....	10
表 3: 错误代码列表.....	37
表 4: 参考文档.....	39
表 5: 术语缩写 .....	39

# 1 引言

移远通信 5G RG50xQ 系列和 RM5xxQ 系列模块内置 TCP/IP 协议栈，Host 可以直接通过 AT 命令访问网络；这在很大程度上可降低对 PPP 和外部 TCP/IP 协议栈的依赖性，从而降低终端设计的成本。

RG50xQ 系列和 RM5xxQ 系列模块可提供 TCP 客户端、UDP 客户端、TCP 服务器、UDP 服务器等 Socket 服务。

本文档主要介绍如何通过 AT 命令使用移远通信 RG50xQ 系列和 RM5xxQ 系列模块的 TCP/IP 功能。

## 1.1. 适用模块

表 1：适用模块

模块系列	模块型号
RG50xQ	RG500Q 系列
	RG501Q-EU
	RG502Q 系列
RM5xxQ	RM500Q 系列
	RM502Q-AE
	RM510Q-GL
	RM505Q-AE

## 1.2. TCP/IP AT 命令使用流程

通过 TCP/IP AT 命令，Host 可以配置 PDP 上下文、激活/去激活 PDP 上下文、打开/关闭 Socket 服务，并通过 Socket 服务发送/接收数据。TCP/IP AT 命令的使用流程如下图所示：

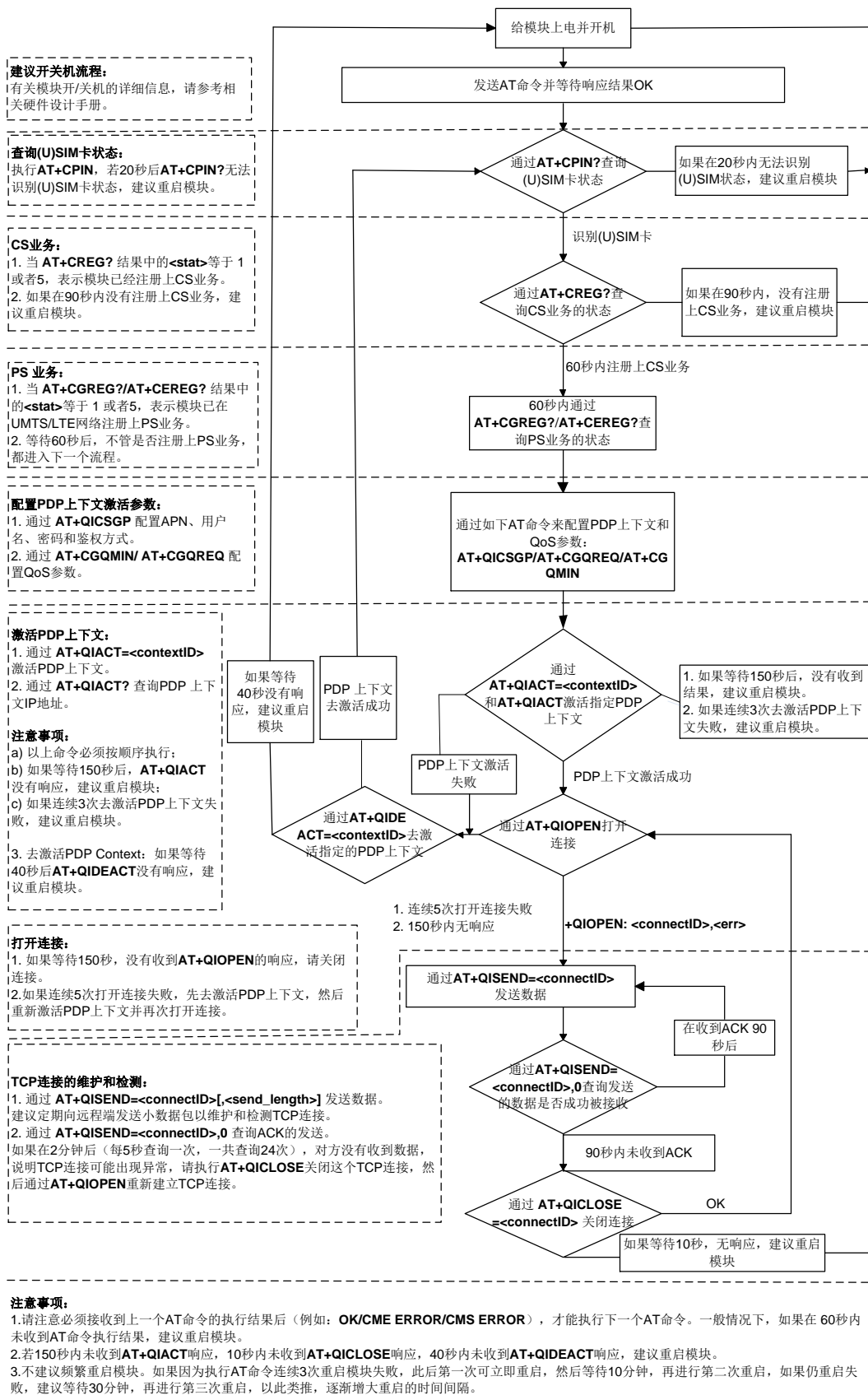


图 1: TCP/IP AT 命令使用流程图

### 1.3. 数据访问模式说明

RG50xQ 系列和 RM5xxQ 系列模块支持以下 3 种数据访问模式：

- 缓存模式
- 直吐模式
- 透传模式

当使用 **AT+QIOPEN** 打开 Socket 服务时，可以通过 **<access\_mode>** 来指定 Socket 的数据访问模式；当 Socket 服务成功打开后，可以通过 **AT+QISWTMD** 切换数据访问模式。

1. 缓存模式下，可以通过 **AT+QISEND** 发送数据。从网络接收数据时，模块会缓存所接收的数据，并直接上报 **URC +QIURC: "recv",<connectID>**，之后用户可以通过 **AT+QIRD** 来读取缓存数据。
2. 直吐模式下，可以通过 **AT+QISEND** 发送数据。从网络接收数据时，数据会以如下格式直接输出到 COM 口：**+QIURC: "recv",<connectID>, <currentrecvlength><CR><LF><data>或 +QIURC: "recv",<connectID>,<currentrecvlength>,<remoteIP>,<remote\_port><CR><LF><data>**。
3. 透传模式下，相对应的串口（比如 UART 口、USB Modem 口等）会进入独占模式，通过 COM 口接收的数据会直接发送到网络端，从网络接收到的数据会从 COM 口直接输出。此时串口不可进行其他操作，如执行 AT 命令等。

#### ● 退出透传模式

用户可以通过 **+++** 方式退出透传模式，为了防止 **+++** 被当成数据发送，实际操作时必须遵循以下步骤：

- 1) 输入 **+++** 前后 1 秒内不能输入其它任何数据；
- 2) 在 1 秒内输入 **+++**，等待模块返回 **OK**；得到 **OK** 响应后，模块会退出透传模式。

#### ● 切换到透传模式

- 1) 通过执行 **AT+QISWTMD**：执行此命令时，指定 **<access\_mode>** 为 2，若返回 **CONNECT**，则表示成功切换到透传模式；
- 2) 通过执行 **ATO**：退出透传模式后，可通过 **ATO** 再切换回至透传模式；若返回 **CONNECT**，则表示成功切换到透传模式。若模块之前未曾进入过透传模式，执行 **ATO** 则会返回 **NO CARRIER**。

### 备注

1. 在缓存模式下，如果缓存不为空，直到通过 **AT+QIRD** 读取缓存里面所有接收的数据后，模块才会上报一个新的 URC。
2. 在透传模式下，不可执行 AT 命令。若因网络错误或者其他原因导致 Socket 连接断开，模块会上报 **NO CARRIER**，并退出透传模式，在这种情况下可以执行 **AT+QICLOSE** 来关闭 Socket 服务。

## 2 TCP/IP AT 命令详解

本章节主要描述和 TCP/IP 相关的 AT 命令。

### 2.1. AT 命令说明

#### 2.1.1. 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。

#### 2.1.2. 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。在本文档中表现命令和响应的表格中，省略了 **<CR><LF>**，仅显示命令和响应。

表 2：AT 命令类型

AT 命令类型	语句	描述
测试命令	<b>AT+&lt;cmd&gt;=?</b>	测试是否存在相应命令，并返回有关其参数的类型、值或范围的信息。
查询命令	<b>AT+&lt;cmd&gt;?</b>	查询相应命令的当前参数值。
设置命令	<b>AT+&lt;cmd&gt;=&lt;p1&gt;[,&lt;p2&gt;[,&lt;p3&gt;[...]]]</b>	设置用户可定义的参数值。
执行命令	<b>AT+&lt;cmd&gt;</b>	返回特定的参数信息或执行特定的操作。

## 2.2. AT 示例声明

本文中的示例仅为方便用户了解 AT 命令的使用方法，不构成移远通信对终端流程设计的建议或意见，也不代表模块应被设置成相应示例中的状态。某些 AT 命令存在多个示例，这些示例之间不存在承接关系或连续性。

## 2.3. AT 命令详解

### 2.3.1. AT+QICSGP 配置 TCP/IP 上下文参数

该命令用于配置<APN>、<username>和<password>以及其他 TCP/IP 上下文参数。QoS 参数配置需通过 AT+CGQMIN 和 AT+CGQREQ 命令；有关所述 AT 命令的详情，请参考文档 [1]。

AT+QICSGP 配置 TCP/IP 上下文参数	
测试命令 <b>AT+QICSGP=?</b>	响应 <b>+QICSGP: (支持的&lt;contextID&gt;范围),(支持的&lt;context_type&gt;范围),&lt;APN&gt;,&lt;username&gt;,&lt;password&gt;,(支持的&lt;authentication&gt;范围)</b>  <b>OK</b>
设置命令 查询指定上下文配置 <b>AT+QICSGP=&lt;contextID&gt;</b>	响应 <b>+QICSGP: &lt;context_type&gt;,&lt;APN&gt;,&lt;username&gt;,&lt;password&gt;,&lt;authentication&gt;</b>  <b>OK</b>
设置命令 配置指定上下文 <b>AT+QICSGP=&lt;contextID&gt;[,&lt;context_type&gt;,&lt;APN&gt;[,&lt;username&gt;,&lt;password&gt;[,&lt;authentication&gt;]]]</b>	响应 若省略可选参数，则查询当前指定上下文配置： <b>+QICSGP: &lt;context_type&gt;,&lt;APN&gt;,&lt;username&gt;,&lt;password&gt;,&lt;authentication&gt;</b>  <b>OK</b>  若指定可选参数，则配置指定上下文： <b>OK</b>  如有错误： <b>ERROR</b>
最大响应时间	/
特性说明	该命令立即生效； 参数配置自动保存。

## 参数

<contextID>	整型。PDP 上下文 ID。范围：1~16。
<context_type>	整型。协议类型。
	1 IPv4
	2 IPv6
	3 IPv4v6
<APN>	字符串类型。接入点名称。
<username>	字符串类型。用户名。最大长度：127 字节。
<password>	字符串类型。密码。最大长度：127 字节。
<authentication>	整型。APN 鉴权方式。
	0 None
	1 PAP
	2 CHAP
	3 PAP 或 CHAP

### 2.3.2. AT+QIACT 激活 PDP 上下文

在用 **AT+QIACT** 激活 PDP 上下文前，需要使用 **AT+QICSGP** 配置上下文。上下文激活后，可以通过 **AT+QIACT?** 查询 IP 地址。

<contextID>的范围为 1~16，模块最多仅可同时激活 3 路 PDP 上下文。受网络状态影响，执行 **AT+QIACT** 后，等待返回结果 **OK** 或者 **ERROR** 的最大时间为 150 秒，在结果尚未返回之前，不允许执行任何 AT 命令。

AT+QIACT 激活 PDP 上下文	
测试命令 <b>AT+QIACT=?</b>	响应 <b>+QIACT: (支持的&lt;contextID&gt;范围)</b>  <b>OK</b>
查询命令 <b>AT+QIACT?</b>	响应 返回当前所有被激活的上下文与 IP 地址： <b>+QIACT: 1,&lt;context_state&gt;,&lt;context_type&gt;[,&lt;IP_addresses&gt;]</b> <b>[...]</b> <b>+QIACT: 16,&lt;context_state&gt;,&lt;context_type&gt;[,&lt;IP_addresses&gt;]]</b>  <b>OK</b>
设置命令 激活指定 PDP 上下文 <b>AT+QIACT=&lt;contextID&gt;</b>	响应 <b>OK</b>  如有错误：

	ERROR
最大响应时间	受网络状态影响，最大响应时间 150 秒。
特性说明	/

## 参数

<contextID>	整型。PDP 上下文 ID。范围：1~16。
<context_state>	整型。PDP 上下文状态。 0 去激活 1 激活
<context_type>	整型。协议类型。 1 IPv4 2 IPv6 3 IPv4v6
<IP_address>	字符串类型。PDP 上下文激活后的本地 IP 地址。

### 2.3.3. AT+QIDEACT 去激活 PDP 上下文

该命令用于去激活特定上下文，断开在这个上下文内建立的所有 TCP/IP 连接。受网络状态影响，执行 **AT+QIDEACT** 命令后，等待返回结果 **OK** 或者 **ERROR** 的最大时间为 40 秒。在结果尚未返回之前，不允许执行任何 AT 命令。

AT+QIDEACT 去激活 PDP 上下文	
测试命令 <b>AT+QIDEACT=?</b>	响应 <b>+QIDEACT: (支持的&lt;contextID&gt;范围)</b>  <b>OK</b>
设置命令 <b>AT+QIDEACT=&lt;contextID&gt;</b>	响应 <b>OK</b>  如有错误： <b>ERROR</b>
最大响应时间	受网络状态影响，最大响应时间 40 秒。
特性说明	/

## 参数

<contextID>	整型。PDP 上下文 ID。范围：1~16。
-------------	------------------------

### 2.3.4. AT+QIOPEN 打开 Socket 服务

该命令可用来打开 Socket 服务。服务类型可通过<service\_type>指定，数据访问模式（缓存模式、直吐模式和透传模式）可通过<access\_mode>配置，URC +QIOPEN 会显示 Socket 服务是否成功打开。

1. 如果<service\_type>是"TCP LISTENER"，那么模块作为 TCP 服务器使用。接受一个新的 TCP 连接后，模块会自动指定一个<connectID>并上报 URC +QIURC: "incoming",<connectID>,<serverID>,<remoteIP>,<remote\_port>，其中<connectID>范围为 0~11，所述新连接的请求类型为"TCP INCOMING"，<access\_mode>值和"TCP LISTENER"的该参数值一样。
2. 如果<service\_type>是"UDP SERVICE"，则可通过<local\_port>向远程 IP 收发 UDP 数据。
  - 发送数据：执行 AT+QISEND=<connectID>,<send\_length>,<remoteIP>,<remote\_port>。
  - 在直吐模式下接收数据：模块会上报 URC +QIURC: "recv",<connectID>,<currentrecvlength>,<remoteIP>,<remote\_port><CR><LF><data>。
  - 在缓存模式下接收数据：模块会上报 URC +QIURC: "recv",<connectID>，然后用户可通过 AT+QIRD=<connectID>读取数据。
3. 等待+QIOPEN: <connectID>,<err>输出，建议等待 150 秒，如果 150 秒内未接收到 URC，则需通过 AT+QICLOSE 来关闭 Socket。

#### AT+QIOPEN 打开 Socket 服务

测试命令 AT+QIOPEN=?	<p>响应</p> <p>+QIOPEN: (支持的&lt;contextID&gt;范围),(支持的&lt;connectID&gt;范围),"TCP/UDP/TCP LISTENER/UDP SERVICE",&lt;IP_address&gt;/&lt;domain_name&gt;,&lt;remote_port&gt;,&lt;local_port&gt;,(支持的&lt;access_mode&gt;范围)</p> <p>OK</p>
<p>设置命令</p> <p>AT+QIOPEN=&lt;contextID&gt;,&lt;connectID&gt;,&lt;service_type&gt;,&lt;IP_address&gt;/&lt;domain_name&gt;,&lt;remote_port&gt;[,&lt;local_port&gt;[,&lt;access_mode&gt;]]</p>	<p>响应</p> <p>如果数据访问模式是透传模式（&lt;access_mode&gt;=2）并且成功打开服务： CONNECT</p> <p>如有错误： ERROR 可以通过 AT+QIGETERROR 查询错误描述。</p> <p>如果数据访问模式是缓存模式（&lt;access_mode&gt;=0）或者直吐模式（&lt;access_mode&gt;=1）： OK</p> <p>+QIOPEN: &lt;connectID&gt;,&lt;err&gt;</p>

	当服务成功打开，<err>为 0，其他情况下<err>不为 0。
最大响应时间	受网络状态影响，最大响应时间 150 秒。
特性说明	/

## 参数

<contextID>	整型。PDP 上下文 ID。范围：1~16。
<connectID>	整型。Socket 连接号。范围：0~11。
<service_type>	字符串类型。Socket 服务类型。 "TCP" 客户端建立 TCP 连接 "UDP" 客户端建立 UDP 连接 "TCP LISTENER" 建立 TCP 服务器监听 TCP 连接 "UDP SERVICE" 建立 UDP 服务
<IP_address>	字符串类型。 如果<service_type>是"TCP"或者"UDP"，则为远程服务器的 IP 地址，例如 220.180.239.212。 如果<service_type>是"TCP LISTENER"或者"UDP SERVICE"，请输入 127.0.0.1。
<domain_name>	字符串类型，远程服务器的域名地址。
<remote_port>	整型。远程服务器端口。范围：0~65535。仅当<service_type>为"TCP"或者"UDP"时才有效。
<local_port>	整型。本地端口。范围：0~65535。 如果<service_type>是"TCP LISTENER"或者"UDP SERVICE"，该参数必须指定。 如果<service_type>是"TCP"或者"UDP"，且<local_port>是 0，那么将会自动分配本地端口；如为其他值，则指定本地端口。
<access_mode>	整型。Socket 服务的数据访问模式。 0 缓存模式 1 直吐模式 2 透传模式
<err>	错误码。详情请参考第 4 章。

### 2.3.5. AT+QICLOSE 关闭 Socket 服务

该命令可用来关闭特定的 Socket 服务。受网络状态影响，执行 AT+QICLOSE 后，等待返回结果 OK 或者 ERROR 的最大时间为 10 秒（此为默认值，可以通过<timeout>来修改）。在结果尚未返回之前，不允许执行任何 AT 命令。

#### AT+QICLOSE 关闭 Socket 服务

测试命令 AT+QICLOSE=?	响应 +QICLOSE: (支持的<connectID>范围),(支持的<timeout>范围)
----------------------	---

	OK
设置命令 AT+QICLOSE=<connectID>[,<timeout>]	响应 关闭成功: OK  关闭失败: ERROR
最大响应时间	默认 10 秒，取决于<timeout>取值
特性说明	/

## 参数

<connectID>	整型。Socket 连接号。范围：0~11。
<timeout>	整型。输出响应结果的超时时间值。若未在<timeout>时间范围内接收到其他客户端的 FIN ACK，模块会强制关闭 Socket。范围：0~65535；默认值：10；单位：秒。

### 2.3.6. AT+QISTATE 查询 Socket 服务状态

该命令用于查询 Socket 服务状态。若<query\_type>为 0，则返回在特定上下文下现存所有 Socket 服务的状态；若<query\_type>为 1，则返回特定 Socket 服务的状态。

AT+QISTATE 查询 Socket 服务状态	
测试命令 AT+QISTATE=?	响应 OK
查询/执行命令 AT+QISTATE? 或 AT+QISTATE	响应 返回现存所有连接状态: +QISTATE: <connectID>,<service_type>,<IP_address>,<remote_port>,<local_port>,<socket_state>,<contextID>,<serverID>,<access_mode>,<AT_port> [...]  OK
设置命令 若<query_type>为 0，查询特定上下文中的连接状态 AT+QISTATE=<query_type>,<contextID>	响应 返回特定上下文中现存所有连接的状态: +QISTATE: <connectID>,<service_type>,<IP_address>,<remote_port>,<local_port>,<socket_state>,<contextID>,<serverID>,<access_mode>,<AT_port> [...]

	OK
设置命令 若<query_type>为 1，查询特定 Socket 服务连接状态 <b>AT+QISTATE=&lt;query_type&gt;,&lt;connectID&gt;</b>	响应 <b>+QISTATE: &lt;connectID&gt;,&lt;service_type&gt;,&lt;IP_address&gt;,&lt;remote_port&gt;,&lt;local_port&gt;,&lt;socket_state&gt;,&lt;contextID&gt;,&lt;serverID&gt;,&lt;access_mode&gt;,&lt;AT_port&gt;</b> OK
最大响应时间	/
特性说明	/

## 参数

<b>&lt;query_type&gt;</b>	整型。查询类型。 0 查询特定上下文中所有 Socket 服务的连接状态 1 查询特定 Socket 服务连接状态
<b>&lt;contextID&gt;</b>	整型。PDP 上下文 ID。范围：1~16。
<b>&lt;connectID&gt;</b>	整型。Socket 连接号。范围：0~11。
<b>&lt;service_type&gt;</b>	字符串类型。Socket 服务类型。 "TCP" 客户端建立 TCP 连接 "UDP" 客户端建立 UDP 连接 "TCP LISTENER" 建立 TCP 服务器监听 TCP 连接 "TCP INCOMING" 建立 TCP 服务器接受的 TCP 连接 "UDP SERVICE" 建立 UDP 服务
<b>&lt;IP_address&gt;</b>	字符串类型。IP 地址。 若<service_type>="TCP"或"UDP"，则指远程服务器 IP 地址 若<service_type>="TCP LISTENER"或"UDP SERVICE"，则指本地 IP 地址 若<service_type>="TCP INCOMING"，则指远程客户端 IP 地址
<b>&lt;remote_port&gt;</b>	整型。远程服务器端口。范围：0~65535。 若<service_type>="TCP"或"UDP"，则指远程服务器端口 若<service_type>="TCP LISTENER"或"UDP SERVICE"，则<remote_port>为 0，端口无效 若<service_type>="TCP INCOMING"，则指远程客户端端口
<b>&lt;local_port&gt;</b>	整型。本地端口号。范围：0~65535。 若<local_port>为 0，本地端口自动分配。
<b>&lt;socket_state&gt;</b>	整型。Socket 服务状态。 0 "Initial"：尚未建立连接 1 "Opening"：客户端正在连接或者服务器正尝试监听 2 "Connected"：客户端连接已建立 3 "Listening"：服务器正在监听 4 "Closing"：连接断开
<b>&lt;serverID&gt;</b>	整型。仅当<service_type>为"TCP INCOMING"时才有效。<serverID>表示接受了这

	个 TCP 连接的服务器，这个参数值和该服务器"TCP LISTENER"的<connectID>值一致。
<access_mode>	整型。数据访问模式。
	0 缓存模式
	1 直吐模式
	2 透传模式
<AT_port>	字符串类型。Socket 服务的 COM 口。
	"usbmodem" USB Modem port
	"usbat" USB AT port
	"uart1" UART port 1
	"cmux1" MUX port 1
	"cmux2" MUX port 2
	"cmux3" MUX port 3
	"cmux4" MUX port 4

### 2.3.7. AT+QISEND 发送数据

若特定 Socket 服务的数据访问模式是缓存模式（<access\_mode>=0）或者直吐模式（<access\_mode>=1），可通过 **AT+QISEND** 发送数据。若成功发送数据到模块，则返回 **SEND OK**；否则返回 **SEND FAIL** 或 **ERROR**。

- **SEND FAIL** 表示发送缓存已满，可以尝试重新发送数据；
- **ERROR** 表示在发送数据过程中遇到错误，请延迟一段时间后再次发送数据，发送数据的最大字节长度为 1460 字节；
- **SEND OK** 并不表示数据已成功发送至服务器，可通过 **AT+QISEND=<connectID>,0** 查询数据是否已经发送至服务器。

AT+QISEND 发送数据	
测试命令 <b>AT+QISEND=?</b>	响应 <b>+QISEND: ( 支持的 &lt;connectID&gt; 范围 ),( 支持的 &lt;send_length&gt;范围)</b>  <b>OK</b>
设置命令 若<service_type>="TCP"、"UDP"或者 "TCP INCOMING", 发送不定长数据 <b>AT+QISEND=&lt;connectID&gt;</b>	响应 <b>&gt;</b> 响应 <b>&gt;</b> 后，输入需要发送的数据，按 <b>CTRL + Z</b> 发送数据，按 <b>Esc</b> 取消发送。  若连接已经建立，且发送成功： <b>SEND OK</b>  若连接已经建立，但是发送缓存已满： <b>SEND FAIL</b>

	<p>若连接未建立，异常断开或者参数不正确： <b>ERROR</b></p>
<p>设置命令 若 <b>&lt;service_type&gt;="TCP"、"UDP" 或 "TCP INCOMING"</b>，发送定长数据 <b>AT+QISEND=&lt;connectID&gt;,&lt;send_length&gt;</b></p>	<p>响应 <b>&gt;</b> 响应 <b>&gt;</b> 后，输入长度等于<b>&lt;send_length&gt;</b>的数据。</p> <p>若连接已经建立，且发送成功： <b>SEND OK</b></p> <p>若连接已经建立，但是发送缓存已满： <b>SEND FAIL</b></p> <p>若连接未建立，异常断开或者参数不正确： <b>ERROR</b></p>
<p>设置命令 若<b>&lt;service_type&gt;="UDP SERVICE"</b> <b>AT+QISEND=&lt;connectID&gt;,&lt;send_length&gt;,&lt;remoteIP&gt;,&lt;remote_port&gt;</b></p>	<p>响应 该命令可用来发送固定字节长度数据到特定的远程 IP 地址和远程端口，<b>&lt;service_type&gt;</b>必须是"UDP SERVICE"。 <b>&gt;</b> 响应 <b>&gt;</b> 后，输入长度等于<b>&lt;send_length&gt;</b>的数据。</p> <p>若连接已经建立且发送成功： <b>SEND OK</b></p> <p>若连接已经建立，但是发送缓存已满： <b>SEND FAIL</b></p> <p>若连接未建立，异常断开或者参数不正确： <b>ERROR</b></p>
<p>设置命令 当<b>&lt;send_length&gt;=0</b>，可查询发送数据 <b>AT+QISEND=&lt;connectID&gt;,0</b></p>	<p>响应 若指定连接存在： <b>+QISEND: &lt;total_send_length&gt;,&lt;ackedbytes&gt;,&lt;unacked bytes&gt;</b></p> <p><b>OK</b></p> <p>如有错误： <b>ERROR</b></p>
最大响应时间	/
特性说明	/

## 参数

<connectID>	整型。Socket 连接号。范围：0~11。
<send_length>	整型。发送数据字节长度。范围：1~1460；单位：字节。
<remoteIP>	字符串类型，远程 IP 地址（必须是点分十进制格式）。仅当<service_type>为"UDP SERVICE"时有效。
<remote_port>	整型。远程端口。范围：0~65535。仅当<service_type>为"UDP SERVICE"时有效。
<total_send_length>	整型。发送数据总长度。单位：字节。
<ackedbytes>	整型。收到数据总长度。单位：字节。
<unackedbytes>	整型。未收到数据总长度。单位：字节。

### 2.3.8. AT+QIRD 读取接收的 TCP/IP 数据

在缓存模式下，模块收到数据后会先缓存数据并上报+QIURC: "recv",<connectID>，之后可通过 AT+QIRD 读取所述数据。

缓存仍有数据时，模块若再次接收到数据，将不会上报 URC，直到缓存中所有数据均被读取，才会上报新的 URC。

AT+QIRD 读取接收的 TCP/IP 数据	
测试命令 AT+QIRD=?	<p>响应</p> <p>+QIRD: (支持的&lt;connectID&gt;范围),(支持的&lt;read_length&gt;范围)</p> <p>OK</p>
设置命令 当<service_type>为 "TCP"/"UDP"/"TCP INCOMING" AT+QIRD=<connectID>[,<read_length>]	<p>响应</p> <p>若特定连接收到数据：</p> <p>+QIRD: &lt;read_actual_length&gt; &lt;data&gt;</p> <p>OK</p> <p>若无数据：</p> <p>+QIRD: 0</p> <p>OK</p> <p>若连接不存在：</p> <p>ERROR</p>
设置命令 当<service_type>为"UDP SERVICE" AT+QIRD=<connectID>	<p>响应</p> <p>若数据存在：</p> <p>+QIRD: &lt;read_actual_length&gt;,&lt;remoteIP&gt;,&lt;remote_port&gt;</p>

	<p><b>&lt;data&gt;</b></p> <p><b>OK</b></p> <p>若无数据： <b>+QIRD: 0</b></p> <p><b>OK</b></p> <p>若连接不存在： <b>ERROR</b></p>
设置命令 当<read_length>为 0，查询读取的数据 <b>AT+QIRD=&lt;connectID&gt;,0</b>	<p>响应</p> <p>若指定连接存在： <b>+QIRD: &lt;total_receive_length&gt;,&lt;have_read_length&gt;,&lt;unread_length&gt;</b></p> <p><b>OK</b></p> <p>如有错误： <b>ERROR</b></p>
最大响应时间	/
特性说明	/

## 参数

<b>&lt;connectID&gt;</b>	整型。Socket 连接号。范围：0~11。
<b>&lt;read_length&gt;</b>	整型。待读取数据的最大长度。范围：0~1500；单位：字节。
<b>&lt;read_actual_length&gt;</b>	整型。实际读取数据的长度。单位：字节。
<b>&lt;remoteIP&gt;</b>	字符串类型。远程 IP 地址。仅当<service_type>为"UDP SERVICE"时有效。
<b>&lt;remote_port&gt;</b>	整型。远程端口。范围：0~65535。仅当<service_type>为"UDP SERVICE"时有效。
<b>&lt;data&gt;</b>	字符串类型。读取的数据。
<b>&lt;total_receive_length&gt;</b>	整型。接收数据的总长度。单位：字节。
<b>&lt;have_read_length&gt;</b>	整型。已读取数据的长度。单位：字节。
<b>&lt;unread_length&gt;</b>	整型。未读取数据的长度。单位：字节。

### 2.3.9. AT+QISENDEX 发送 16 进制字符串数据

该命令用于发送 16 进制字符串数据，不适用于"UDP SERVICE"和"TCP LISTENER"两种 Socket 服务类型。

AT+QISENDEX 发送 16 进制字符串数据	
测试命令 <b>AT+QISENDEX=?</b>	响应 <b>+QISENDEX: (支持的&lt;connectID&gt;范围),&lt;hex_string&gt;</b>  <b>OK</b>
设置命令 <b>AT+QISENDEX=&lt;connectID&gt;,&lt;hex_string&gt;</b>	响应 若数据发送成功: <b>SEND OK</b>  若发送缓存已满: <b>SEND FAIL</b>  若连接不存在: <b>ERROR</b>
最大响应时间	/
特性说明	/

#### 参数

<b>&lt;connectID&gt;</b>	整型。Socket 连接号。范围：0~11。
<b>&lt;hex_string&gt;</b>	字符串类型。16 进制字符串数据。最大长度：512 字节。

### 2.3.10. AT+QISWTMD 切换数据访问模式

该命令用于切换数据访问模式：缓存模式、直吐模式和透传模式。建立 Socket 服务时，可以通过 **AT+QIOPEN** 的<access\_mode>来指定数据访问模式，Socket 服务打开以后，可以通过 **AT+QISWTMD** 来切换模式。

AT+QISWTMD 切换数据访问模式	
测试命令 <b>AT+QISWTMD=?</b>	响应 <b>+QISWTMD: (支持的&lt;connectID&gt;范围),(支持的&lt;access_m ode&gt;范围)</b>  <b>OK</b>
设置命令	响应

AT+QISWTMD=<connectID>,<access_mode>	<p>若&lt;access_mode&gt;=0 或 1 且数据访问模式切换成功: <b>OK</b></p> <p>若&lt;access_mode&gt;=2 且数据访问模式切换成功, 模块会进入透传模式: <b>CONNECT</b></p> <p>如有错误: <b>ERROR</b></p>
最大响应时间	/
特性说明	该命令立即生效; 参数配置不保存。

## 参数

<connectID>	整型。Socket 连接号。范围: 0~11。
<access_mode>	整型。连接数据访问模式。
0	缓存模式
1	直吐模式
2	透传模式

### 2.3.11. AT+QPING 进行远程服务器 Ping 操作

该命令用于检测 Host 网络协议可达性。在执行 Ping 操作命令前, Host 应首先通过 **AT+QIACT** 激活<contextID>对应的上下文 ID, 在<timeout>期间内返回结果, <timeout>默认值是 4 秒。

#### AT+QPING 进行远程服务器 Ping 操作

测试命令 AT+QPING=?	<p>响应</p> <p><b>+QPING: (支持的&lt;contextID&gt;范围),&lt;host&gt;,(支持的&lt;timeout&gt;范围),(支持的&lt;pingnum&gt;范围)</b></p> <p><b>OK</b></p>
设置命令 AT+QPING=<contextID>,<host>[,<timeout>[,<pingnum>]]	<p>响应</p> <p>若远程服务器Ping操作成功: <b>OK</b></p> <p><b>+QPING: &lt;result&gt;[,&lt;IP_address&gt;,&lt;bytes&gt;,&lt;time&gt;,&lt;ttl&gt;]</b></p> <p><b>[...]</b></p> <p><b>+QPING: &lt;finresult&gt;[,&lt;sent&gt;,&lt;rcvd&gt;,&lt;lost&gt;,&lt;min&gt;,&lt;max&gt;,&lt;maxrtt&gt;]</b></p>

	<b>&lt;avg&gt;]</b>  如有错误： <b>ERROR</b>
最大响应时间	/
特性说明	该命令立即生效； 参数配置不保存。

## 参数

<b>&lt;contextID&gt;</b>	整型。PDP 上下文 ID。范围：1~16。
<b>&lt;host&gt;</b>	字符串类型。Host 地址，格式为域名或点分十进制的 IP 地址。
<b>&lt;timeout&gt;</b>	整型。设置每个 Ping 请求回应的等待最大时间。范围：1~255；默认值：4；单位：秒。
<b>&lt;pingnum&gt;</b>	整型。设置 Ping 请求最大次数。范围：1~10；默认值：4。
<b>&lt;result&gt;</b>	整型。每个 Ping 请求结果。 0 从服务器接收到 Ping 响应，其后紧跟,<IP_address>,<bytes>,<time>,<ttd>。 其他值 错误码。请参考第 4 章。
<b>&lt;IP_address&gt;</b>	字符串类型。远程服务器 IP 地址。格式为点分十进制 IP。
<b>&lt;bytes&gt;</b>	整型。实际发送的 Ping 请求字节长度。单位：字节。
<b>&lt;time&gt;</b>	整型。发送 Ping 请求后的等待响应时间。单位：毫秒。
<b>&lt;ttd&gt;</b>	整型。Ping 请求回应数据包的 TTL 值。
<b>&lt;finresult&gt;</b>	整型。该命令执行后的最终结果。 0 正常结束，成功激活上下文并找到 Host，此情况下其后紧跟,<sent>,<rcvd>,<lost>,<min>,<max>,<avg> 其他值 错误码。请参考第 4 章。
<b>&lt;sent&gt;</b>	整型。Ping 请求的发送次数。
<b>&lt;rcvd&gt;</b>	整型。得到响应的 Ping 请求个数。
<b>&lt;lost&gt;</b>	整型。超时的 Ping 请求个数。
<b>&lt;min&gt;</b>	整型。最小响应时间。单位：毫秒。
<b>&lt;max&gt;</b>	整型。最大响应时间。单位：毫秒。
<b>&lt;avg&gt;</b>	整型。平均响应时间。单位：毫秒。

### 2.3.12. AT+QISDE 控制是否回显 AT+QISEND 要发送的数据

该命令用于控制是否回显 AT+QISEND 要发送的数据。

#### AT+QISDE 控制是否回显 AT+QISEND 要发送的数据

测试命令	响应
<b>AT+QISDE=?</b>	<b>+QISDE: (支持的&lt;echo&gt;列表)</b>
	<b>OK</b>

查询命令 <b>AT+QISDE?</b>	响应 <b>+QISDE: &lt;echo&gt;</b>  <b>OK</b>
设置命令 <b>AT+QISDE=&lt;echo&gt;</b>	响应 <b>OK</b>  如有错误: <b>ERROR</b>
最大响应时间	/
特性说明	该命令立即生效; 参数配置不保存。

## 参数

<b>&lt;echo&gt;</b>	整型。是否回显 <b>AT+QISEND</b> 要发送的数据。 0 不回显 1 回显
---------------------	---

### 2.3.13. AT+QIGETERROR 查询上一个 AT 命令错误代码

若执行 TCP/IP 命令后返回 **ERROR**，错误详情可通过 **AT+QIGETERROR** 查询。需注意 **AT+QIGETERROR** 只返回上一个 TCP/IP AT 命令的错误代码。

<b>AT+QIGETERROR 查询上一个 AT 命令错误代码</b>	
测试命令 <b>AT+QIGETERROR=?</b>	响应: <b>OK</b>
执行命令 <b>AT+QIGETERROR</b>	响应: <b>+QIGETERROR: &lt;err&gt;,&lt;errcode_description&gt;</b>  <b>OK</b>
最大响应时间	/
特性说明	/

## 参数

<b>&lt;err&gt;</b>	错误码。详情请参考第 4 章。
<b>&lt;errcode_description&gt;</b>	字符串类型。指示详细错误信息。请参考第 4 章。

## 2.4. URC 说明

TCP/IP AT 命令的 URC 统一以**+QIURC:**的形式上报 Host，涉及数据接收、连接断开和接入等。在 URC 的前后均有<CR><LF>，本文里均不显示。

### 2.4.1. +QIURC: "closed" 连接断开通知

当 TCP Socket 服务连接被远程客户端断开或者因为网络异常导致断开，模块将上报 URC，同时该 Socket 将处于"Closing"状态（<socket\_state>=4），调用 **AT+QICLOSE=<connectID>**后 Socket 服务可恢复成"Initial"状态。

#### +QIURC: "closed" 连接断开通知

+QIURC: "closed",<connectID>	Socket 服务连接被断开
------------------------------	----------------

#### 参数

<connectID>	整型。Socket 连接号。范围：0~11。
-------------	------------------------

### 2.4.2. +QIURC: "recv" 数据接收通知

在缓存模式或者直吐模式下，模块接收到数据后会上报 URC 通知 Host。

- 缓存模式下：URC 格式为**+QIURC: "recv",<connectID>**；URC 上报后，Host 可通过 **AT+QIRD** 读取数据。请注意，如果缓存不为空且模块再次接收数据的情况下，只有当 Host 通过 **AT+QIRD** 读取所有接收的数据后，模块才会上报新的 URC。
- 直吐模式下：接收的数据会直接从 COM 端输出。

#### +QIURC: "recv" 数据接收通知

+QIURC: "recv",<connectID>	在缓存模式下，模块接收到数据会上报该 URC，Host 可通过 <b>AT+QIRD</b> 获取数据。
+QIURC: "recv",<connectID>,<currentrecvlength><CR><LF><data>	在直吐模式下，当<service_type>为"TCP"、"UDP"或"TCP INCOMING"时，模块接收到数据会上报该 URC。
+QIURC: "recv",<connectID>,<currentrecvlength>,<remoteIP>,<remote_port><CR><LF><data>	在直吐模式下，当<service_type>为"UDP SERVICE"时，模块接收到数据会上报该 URC。

## 参数

<connectID>	整型。Socket 连接号。范围：0~11。
<currentrecvlength>	整型。实际接收数据的长度。
<remoteIP>	字符串类型。远程 IP 地址。
<remote_port>	整型。远程端口。范围：0~65535。
<data>	字符串类型。接收的数据。

### 2.4.3. +QIURC: "incoming full" 客户端连接已满通知

如果客户端连接已达限额，或者已经没有 Socket 系统资源可供分配，有新的客户端连接请求时模块会上报 URC：+QIURC: "incoming full",<serverID>。

#### +QIURC: "incoming full" 客户端连接已满通知

+QIURC: "incoming full",<serverID>	客户端连接请求已满
------------------------------------	-----------

## 参数

<serverID>	整型。接受客户端连接<connectID>的服务器。服务器的<service_type>是"TCP LISTENER"，监听 Socket ID 是<serverID>。
------------	---

### 2.4.4. +QIURC: "incoming" 客户端连接通知

如果<service\_type>为"TCP LISTENER"，当一个远程客户端连接到这个服务器时，Host 会给新连接自动分配一个空闲的<connectID>，其中<connectID>范围是 0~11。此时模块会上报该 URC，新连接的<service\_type>是"TCP INCOMING"，<access\_mode>的值和"TCP LISTENER"的参数值相同。

#### +QIURC: "incoming" 客户端连接通知

+QIURC: "incoming",<connectID>,<serverID>,<remoteIP>,<remote_port>	<serverID>接受了一个新的连接请求时，分配给新连接的<connectID>以及新连接的<remoteIP>和<remote_port>将通过该 URC 上报。
--	---

## 参数

<connectID>	整型。模块自动为客户端连接指定分配的 Socket 服务。范围：0~11。
<serverID>	整型。接受客户端连接<connectID>的服务器。服务器的<service_type>是"TCP LISTENER"，监听 Socket ID 是<serverID>。
<remoteIP>	字符串类型。客户端连接<connectID>的远程 IP 地址。
<remote_port>	整型。客户端连接<connectID>的远程端口。范围：0~65535。

### 2.4.5. +QIURC: "pdpdeact" PDP 去激活通知

PDP 可以被网络去激活。PDP 被去激活以后，模块会上报该 URC 通知 Host，Host 需执行 **AT+QIDEACT** 去激活上下文 ID 并重置所有连接。

#### +QIURC: "pdpdeact" PDP 去激活通知

+QIURC: "pdpdeact",<contextID>	<contextID>对应的 PDP 上下文 ID 被去激活
--------------------------------	--------------------------------

#### 参数

<contextID>	整型。PDP 上下文 ID。范围：1~16。
-------------	------------------------

## 3 示例

### 3.1. 上下文配置和激活

#### 3.1.1. 上下文配置

```
AT+QICSGP=1                                     //查询上下文 1 配置。
+QICSGP: 1,"","","",0

AT+QICSGP=1,1,"UNINET","",1                     //配置上下文 1，APN 配置为"UNINET"（中国联通）。
OK
```

#### 3.1.2. 上下文激活

```
AT+QIACT=1                                       //激活上下文 1，受网络状态影响，最大响应时间为 150 秒。
OK                                              //激活成功。
AT+QIACT?                                       //查询上下文状态。
+QIACT: 1,1,1,"10.7.157.1"

OK
```

#### 3.1.3. 上下文去激活

```
AT+QIDEACT=1                                    //去激活上下文 1。
OK                                              //上下文去激活成功，受网络状态影响，最大响应时间为 40 秒。
```

### 3.2. TCP 客户端在缓存模式下工作

#### 3.2.1. 创建 TCP 客户端连接并进入缓存模式

```
AT+QIOPEN=1,0,"TCP","220.180.239.212",8009,0,0 //上下文是 1，<connectID>为 0。在执行
AT+QIOPEN 之前，Host 需要使用 AT+QIACT
激活上下文。
```

OK

+QIOPEN: 0,0

//TCP 客户端连接成功，等待 URC: **+QIOPE**  
**N: <connectID>,<err>**响应，建议等待 150  
秒。若 150 秒内未收到 URC 响应，Host 可以  
使用 **AT+QICLOSE** 断开 Socket。

**AT+QISTATE=1,0**

//查询<connectID>为 0 时 socket 服务连接状态。

+QISTATE: 0,"TCP","220.180.239.201",8009,65514,2,1,0,0,"usbmodem"

OK

### 3.2.2. 缓存模式下发送数据

**AT+QISEND=0**

//发送可变长度数据。

> test1<ctrl+Z>

SEND OK

//**SEND OK** 不能代表数据已经成功发送到服务器。Host 可以通过 **AT+QISEND=0,0** 来查询数据是否到达服务器。

**AT+QISEND=0,4**

//发送固定长度数据，字节长度为 4 字节。

> test

SEND OK

**AT+QISEND=0,0**

//查询发送数据长度。

+QISEND: 9,9,0

OK

**AT+QISENDEX=0,"3132333435"**

//发送 16 进制字符串数据。

SEND OK

**AT+QISEND=0,0**

//查询发送数据长度，确认数据长度和未确认数据长度。

+QISEND: 14,14,0

OK

### 3.2.3. 缓存模式下从远程服务器接收数据

+QIURC: "recv",0

// <connectID>为 0 接收的数据。

**AT+QIRD=0,1500**

//读取数据，字节长度为 1500 字节。

+QIRD: 5

//实际接收数据长度为 5 字节。

test1

OK

**AT+QICFG="recvind",1**

OK

```

+QIURC: "recv",0,5           //<connectID>为 0，接收的数据字节长度为 5 字节。
AT+QIRD=0,1500              //读取数据，字节长度为 1500 字节。
+QIRD: 5                     //实际接收数据长度为 5 字节。
test1

OK
AT+QIRD=0,1500
+QIRD: 0                     //缓存里无数据。

OK
AT+QIRD=0,0                  //查询接收数据总长度，包含已读和未读数据。
+QIRD: 10,10,0

OK

```

### 3.2.4. 断开连接

```

AT+QICLOSE=0                 //断开<connectID>为 0 的连接。受网络状态影响，最大响应时间是 10 秒。
OK

```

## 3.3. TCP 客户端在透传模式下工作

### 3.3.1. 创建 TCP 客户端连接并进入透传模式

```

AT+QIOPEN=1,0,"TCP","220.180.239.212",8009,0,2 //上下文为 1，<connectID> 为 0。执行
AT+QIOPEN 之前，Host 需使用 AT+QIACT
来激活上下文。
CONNECT                                     //连接成功。等待 URC 响应结果 CONNECT 建
议等待 150 秒。若 150 秒内 URC 无响应，Host
可通过 AT+QICLOSE 断开 Socket 连接。

```

### 3.3.2. 透传模式下发送数据

```

<All data got from COM port will be sent to internet directly>

```

### 3.3.3. 透传模式接收远程器端数据

```

Test 1                               //所有从网络接收的数据都会通过 COM 端口直接输出。

```

### 3.3.4. 断开 TCP 客户端连接

```
AT+QICLOSE=0 //使用+++退出透传模式后,Host 可以通过 AT+QICLOSE 断开 TCP 连接。
               受网络状态影响, 最大响应时间是 10 秒。

OK
```

## 3.4. TCP 客户端在直吐模式下工作

### 3.4.1. 创建 TCP 客户端连接并进入直吐模式

```
AT+QIOPEN=1,0,"TCP","220.180.239.212",8009,0,1 //上下文为 1, <connectID> 为 0。执行
                                                    AT+QIOPEN 之前, Host 需使用 AT+QIACT 来
                                                    激活上下文。

OK

+QIOPEN: 0,0 //连接成功, 等待 URC: +QIOPEN: <connectID>, <err> 响应, 建议等待 150 秒, 若 150 秒内未
              响应, Host 可通过 AT+QICLOSE 断开 Socket。

AT+QISTATE=1,0 //查询<connectID>的连接状态是否为 0。
+QISTATE: 0,"TCP","220.180.239.201",8009,65344,2,1,0,1,"usbmodem"

OK
```

### 3.4.2. 直吐模式下发送数据

```
AT+QISEND=0 //发送长度可变数据。
> test1<ctrl+Z>
SEND OK //SEND OK 不能代表数据已经成功发送至服务器, Host
         可通过 AT+QISEND=0,0 查询数据是否到达服务器。

AT+QISEND=0,5 //发送固定长度数据, 长度为 5 字节。
> test2
SEND OK

AT+QISEND=0,0 //查询发送数据长度, 已确认数据长度和未确认数据长度。
+QISEND: 10,10,0

OK
```

### 3.4.3. 直吐模式下接收远程服务器端数据

```
+QIURC: "recv",0,4           //接收远程服务器端数据。
test
```

### 3.4.4. 断开 TCP 用户端连接

```
AT+QICLOSE=0                 //断开<connectID>为 0 的连接。受网络状态影响，最
                             大响应时间为 10 秒。
OK
```

## 3.5. TCP 服务器在缓存模式下工作

### 3.5.1. 启动 TCP 服务器

```
AT+QIOPEN=1,1,"TCP LISTENER","127.0.0.1",0,2020,0 //上下文为 1，<connectID>为 1。执行
AT+QIOPEN 之前，Host 需通过
AT+QIACT 来激活上下文。
OK

+QIOPEN: 1,0                 //TCP 服务器启动成功。
AT+QISTATE=0,1               //查询<connectID>连接状态是否为 1。
+QISTATE: 1,"TCP LISTENER","127.0.0.1",0,2020,3,1,1,0,"usbmodem"
OK
```

### 3.5.2. 接受客户端的连接请求

```
+QIURC: "incoming",11,1,"172.31.242.222",54091 //接受一个 TCP 连接，<service_type>为"TCP
incoming"，<connectID>为 11。
```

### 3.5.3. 接收的客户端数据

```
+QIURC: "recv",11           //从远程客户端连接接收的数据。
AT+QIRD=11,1500             //读取客户端连接接收的数据。
+QIRD: 4                     //实际数据长度为 4 字节。
test
```

```

OK
AT+QIRD=11,1500
+QIRD: 0 //缓存无数据。

OK
AT+QIRD=11,0 //查询接收数据总长度，包含已读和未读数据。
+QIRD: 4,4,0

OK

```

### 3.5.4. 断开 TCP 服务器连接

```

AT+QICLOSE=11 //断开客户端连接，受网络状态影响最大响应时间 10 秒。
OK
AT+QICLOSE=1 //关闭 TCP 服务器监听。
OK

```

## 3.6. UDP 服务示例

### 3.6.1. 打开 UDP 服务

```

AT+QIOPEN=1,2,"UDP SERVICE","127.0.0.1",0,3030,0 //打开 UDP 服务，<connectID> 为 2，
                                                    <contextID> 为 1。执行 AT+QIOPEN 之前，
                                                    Host 需通过 AT+QIACT 来激活上下文。

OK

+QIOPEN: 2,0 //成功打开 UDP 服务。
AT+QISTATE=0,1 //查询<contextID>的连接状态是否为 1。
+QISTATE: 2,"UDP SERVICE","127.0.0.1",0,3030,2,1,2,0,"usbmodem"

OK

```

### 3.6.2. 向服务器发送 UDP 数据

```

AT+QISEND=2,10,"10.7.89.10",6969 //向远程端发送字节长度为 10 字节的数据，远程
                                     端 IP 地址是 10.7.89.10，端口是 6969。

>1234567890
SEND OK

```

### 3.6.3. 接收远程端数据

<b>+QIURC: "recv",2</b>	//接收远程端数据。
<b>AT+QIRD=2</b>	//读取 UDP 数据。可输出一整个 UDP 数据包，无需指定读取长度。
<b>+QIRD: 4,"10.7.76.34",7687</b>	//数据长度为 4，远程 IP 地址为 10.7.76.34，远程端口是 7687。
<b>AAAA</b>	
<b>OK</b>	
<b>AT+QIRD=2</b>	//读取数据。
<b>+QIRD: 0</b>	//缓存无数据。
<b>OK</b>	
<b>AT+QISEND=2,10,"10.7.76.34",7687</b>	//向远程端发送数据，远程端 IP 地址为 10.7.76.34，远程端口是 7687。
<b>&gt;1234567890</b>	
<b>SEND OK</b>	

### 3.6.4. 关闭 UDP 服务

<b>AT+QICLOSE=2</b>	//关闭服务。
<b>OK</b>	

## 3.7. PING

<b>AT+QPING=1,"www.baidu.com"</b>	//在上下文 1 中 Ping <a href="http://www.baidu.com">www.baidu.com</a> ，在 Ping 目标 IP 地址前，Host 需通过 <b>AT+QIACT</b> 来激活上下文。
<b>OK</b>	
<b>+QPING: 0,"61.135.169.125",32,192,255</b>	
<b>+QPING: 0,"61.135.169.125",32,240,255</b>	
<b>+QPING: 0,"61.135.169.125",32,241,255</b>	
<b>+QPING: 0,"61.135.169.125",32,479,255</b>	
<b>+QPING: 0,4,4,0,192,479,288</b>	

### 3.8. 获取上一个错误代码示例

```
AT+QIOPEN=1,"TCP","220.180.239.212",8009,0,1 //开启Socket服务，省略了<connectID>。  
ERROR  
AT+QIGETERROR  
+QIGETERROR: 552, invalid parameters  
  
OK
```

## 4 错误代码

当 TCP/IP 的 AT 命令返回 **ERROR** 时，具体的错误类型可以通过 **AT+QIGETERROR** 来查询。需注意 **AT+QIGETERROR** 查询的是上一个 TCP/IP AT 命令的错误代码。

表 3：错误代码列表

<err>	<errcode_description>	中文含义
0	Operation success	操作成功
550	Unknown error	未知错误
551	Operation blocked	操作受阻
552	Invalid parameters	无效参数
553	Memory allocation failed	内存不足
554	Socket creation failed	创建Socket失败
555	Operation not supported	操作不支持
556	Socket bind failed	Socket绑定失败
557	Socket listen failed	Socket监听失败
558	Socket write failed	Socket写入失败
559	Socket read failed	Socket读取失败
560	Socket accept failed	Socket接受失败
561	PDP context activate failed	打开PDP上下文失败
562	PDP context deactivate failed	关闭PDP上下文失败
563	Socket identity has been used	Socket标识被占用
564	DNS busy	DNS繁忙
565	DNS parse failed	DNS解析失败

566	Socket connect failed	Socket连接失败
567	Socket has connection reset	Socket连接重置
568	Operation busy	系统忙碌
569	Operation timeout	操作超时
570	PDP context deactivated	PDP上下文去激活
571	Cancel sending	取消发送
572	Operation not allowed	操作不允许
573	Port busy	端口繁忙
574	PDP has been activated	PDP已激活

## 5 附录 参考文档及术语缩写

表 4：参考文档

文档名称
[1] Quectel_RG50xQ&RM5xxQ 系列_AT 命令手册

表 5：术语缩写

缩写	英文全称	中文全称
3GPP	3rd Generation Partnership Project	第三代合作伙伴计划
ACK	Acknowledge	确认字符
APN	Access Point Name	接入点名称
ASCII	American Standard Code for Information Interchange	美国信息交换标准代码
CHAP	Challenge Handshake Authentication Protocol	挑战握手认证协议
CS	Circuit Switching	电路交换
DNS	Domain Name System	域名系统
FIN	Finish	完成
ID	Identify	身份证标识号
IP	Internet Protocol	互联网协议
NV	Non-Volatile	非易失
PAP	Password Authentication Protocol	密码认证协议
PDP	Packet Data Protocol	分组数据协议
PPP	Point-to-Point Protocol	点对点协议

PS	Packet Switching	分组交换
QoS	Quality of Service	服务质量
TCP	Transmission Control Protocol	传输控制协议
UART	Universal Asynchronous Receiver& Transmitter	通用异步收发器
UDP	User Datagram Protocol	用户数据包协议
URC	Unsolicited Result Code	非请求结果码
USB	Universal Serial Bus	通用串行总线
(U)SIM	(Universal) Subscriber Identity Module	（通用）用户身份模块