

# BC26&BC20 TCP/IP AT 命令手册

**NB-IoT 模块系列**

版本: BC26&BC20\_TCP/IP\_AT 命令手册\_V1.0

日期: 2019-12-05

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 2019，保留一切权利。

**Copyright © Quectel Wireless Solutions Co., Ltd. 2019.**

# 文档历史

## 修订记录

版本	日期	作者	变更表述
1.0	2019-12-05	王瑞/蒋涛/ 刘飞彪	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>
1.1. TCP/IP AT 命令简述 .....	5
1.2. 数据访问模式说明 .....	5
<b>2 TCP/IP AT 命令详解 .....</b>	<b>6</b>
2.1. 定义 .....	6
2.2. AT 命令类型 .....	6
2.3. AT 命令详解 .....	7
2.3.1. AT+QIOPEN 打开 Socket .....	7
2.3.2. AT+QICLOSE 关闭 Socket .....	8
2.3.3. AT+QISTATE 查询 Socket 状态 .....	9
2.3.4. AT+QISEND 发送十六进制/文本字符串数据 .....	10
2.3.5. AT+QIRD 读取 TCP/IP 数据 .....	12
2.3.6. AT+QISENDEX 发送十六进制字符串数据 .....	13
2.3.7. AT+QISWTMD 切换数据访问模式 .....	14
2.3.8. AT+QPING Ping 远程服务器 .....	15
2.3.9. AT+QNTP 通过 NTP 服务器同步本地时间 .....	16
2.3.10. AT+QIDNSGIP 转换域名为 IP 地址 .....	17
2.3.11. AT+QIDNSCFG 配置 DNS 服务器地址 .....	18
2.3.12. AT+QICFG 配置可选参数 .....	19
2.3.13. AT+QIGETERROR 查询上一个错误码 .....	21
2.4. URC 详解 .....	22
2.4.1. 连接断开 URC .....	22
2.4.2. 下行数据已达 URC .....	23
2.4.3. 下行数据缓存已满 URC .....	23
<b>3 错误码 .....</b>	<b>24</b>
<b>4 举例 .....</b>	<b>26</b>
4.1. TCP 客户端在缓存模式下工作 .....	26
4.1.1. 创建一个 TCP 客户端连接并进入缓存模式 .....	26
4.1.2. 在缓存模式下发送数据 .....	26
4.1.3. 在缓存模式下从远程服务器上读取数据 .....	27
4.1.4. 断开连接 .....	28
4.2. TCP 客户端在直吐模式下工作 .....	28
4.2.1. 创建一个 TCP 客户端连接并进入直吐模式 .....	28
4.2.2. 在直吐模式下发送数据 .....	28
4.2.3. 在直吐模式下接收来自远程服务器的数据 .....	29
4.2.4. 断开连接 .....	29

4.3.	Ping 远程服务器 .....	29
4.4.	同步本地时间.....	30
4.5.	配置 DNS 服务器.....	30
4.6.	查询上一个错误码 .....	30
5	附录 A 术语缩写 .....	31

## 表格索引

表 1: AT 命令类型及响应 .....	6
表 2: 错误码列表 .....	24
表 3: 术语缩写 .....	31

# 1 引言

移远通信 BC26 和 BC20 模块均内置 TCP/IP 协议栈，可直接通过 AT 命令访问网络；这大大降低模块对 PPP 以及外部 TCP/IP 协议栈的依赖，从而降低终端设计的成本。

BC26 和 BC20 模块可提供 TCP 客户端、UDP 客户端等套接字（Socket）服务。

## 1.1. TCP/IP AT 命令简述

通过 TCP/IP AT 命令，模块可以打开/关闭 Socket，并通过 Socket 收发数据。

## 1.2. 数据访问模式说明

BC26 和 BC20 模块支持如下两种数据访问模式：

- 缓存模式
- 直吐模式

当通过 **AT+QIOPEN** 打开 Socket 时，可以通过参数 **<access\_mode>** 指定数据访问模式。成功打开 Socket 后，可以通过 **AT+QISWTMD** 切换数据访问模式。两种数据访问模式下，均可通过 **AT+QISEND** 或 **AT+QISENDEX** 发送数据。

1. 缓存模式下，模块接收到数据时，会先缓存所接收的数据并上报 URC **+QIURC: "recv",<connectID>[,<current\_recv\_length>]**，之后可通过 **AT+QIRD** 读取缓存数据。
2. 直吐模式下，模块接收到数据时，会通过上报的 URC **+QIURC: "recv",<connectID>,<current\_recv\_length><CR><LF><data>** 直接输出数据。

### 备注

缓存模式下，在缓存数据被全部读取完之前，模块不会再上报新的 URC 通知 MCU 收到下行数据。

## 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. 定义

- **<CR>**: 回车符;
- **<LF>**: 换行符;
- **<...>**: 参数名称, 实际命令行中不包括尖括号 **<>**; 添加下划线 “\_” 的参数取值为默认值;
- **[...]**: 可选参数, 实际命令行中不包括方括号 **[]**; 若无特别说明, 可选参数省略时将采用默认值。

### 2.2. AT 命令类型

表 1: AT 命令类型及响应

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。

#### 备注

若无特别说明, 本文档中的 AT 命令默认立即生效。



## 2.3. AT 命令详解

### 2.3.1. AT+QIOPEN 打开 Socket

该命令用于打开 Socket 以创建 TCP/UDP 连接；可通过参数<service\_type>指定服务类型，并通过参数<access\_mode>指定数据访问模式。命令执行成功以后，会上报 URC +QIOPEN: <connectID>,<err> 通知 Socket 是否成功打开。

AT+QIOPEN 打开 Socket	
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",&lt;IP_address&gt;/&lt;domain_name&gt;,&lt;remote_port&gt;,&lt;local_port&gt;,(支持的&lt;access_mode&gt;列表)[,(支持的&lt;protocol_type&gt;列表)]</p> <p>OK</p>
设置命令 AT+QIOPEN=<contextID>,<connectID>,<service_type>,<IP_address>/<domain_name>,<remote_port>[,<local_port>[,<access_mode>][,<protocol_type>]]	<p>响应</p> <p>OK</p> <p>+QIOPEN: &lt;connectID&gt;,&lt;err&gt;</p> <p>若有任何错误:</p> <p>ERROR</p>
最大响应时间	300 毫秒
保存机制	<IP_address>/<domain_name>、<remote_port>、<local_port>、<access_mode>和<protocol_type>参数配置不保存至 NVRAM；深睡眠唤醒后仍有效

### 参数

<contextID>	整型。场景 ID；范围：1-3。
<connectID>	整型。Socket ID；范围：0-4。
<service_type>	字符串类型。Socket 服务类型。 "TCP" 客户端创建 TCP 连接 "UDP" 客户端创建 UDP 连接
<IP_address>	字符串类型。远程服务器的 IP 地址，例如：220.18.23.22。最大长度：150 字节。
<domain_name>	字符串类型。远程服务器的域名地址。最大长度：150 字节。
<remote_port>	整型。远程服务器的端口号；范围：1-65535。
<local_port>	整型。本地端口号。 0 自动分配本地端口号 1-65535 指定的本地端口号
<access_mode>	整型。Socket 的数据访问模式。

<protocol_type>	0	缓存模式
	1	直吐模式
	整型。互联网协议类型。	
	0	IPv4
	1	IPv6

## 备注

1. 目前仅支持<contextID>=1。
2. URC **+QIOPEN: <connectID>,<err>**上报的超时时间为 60 秒，请在 URC 上报完成后再进行其他操作。
3. 如果 TCP/UDP 连接创建失败，需执行 **AT+QICLOSE=<connectID>**以关闭当前 Socket。
4. 模块开机后，需等 IP 地址 URC（例如：**+IP: 10.18.237.42**，表明模块注网成功）上报完成后方可执行该命令。
5. TCP 会话创建后，模块将无法进入深休眠（Deep Sleep）模式。TCP 连接关闭后，模块方可如期进入深休眠模式。
6. UDP 会话创建后，模块会自动备份最新 UDP 配置信息，MCU 可以在唤醒后直接收发数据。

## 2.3.2. AT+QICLOSE 关闭 Socket

该命令用于关闭指定的 Socket。

AT+QICLOSE 关闭 Socket	
测试命令 <b>AT+QICLOSE=?</b>	响应 <b>+QICLOSE: (支持的&lt;connectID&gt;范围)</b>  <b>OK</b>
设置命令 <b>AT+QICLOSE=&lt;connectID&gt;</b>	响应 若成功关闭： <b>OK</b>  <b>CLOSE OK</b>  若有任何错误： <b>ERROR</b>
最大响应时间	300 毫秒
保存机制	/

## 参数

<connectID>	整型。Socket ID；范围：0-4。
-------------	----------------------

### 2.3.3. AT+QISTATE 查询 Socket 状态

该命令用于查询 Socket 状态。

AT+QISTATE 查询 Socket 状态	
测试命令 <b>AT+QISTATE=?</b>	响应 <b>OK</b>
查询命令 <b>AT+QISTATE?</b>	响应 返回所有现存连接的状态： <b>[+QISTATE: &lt;connectID&gt;,&lt;service_type&gt;,&lt;IP_address&gt;/&lt;domain_name&gt;,&lt;remote_port&gt;,&lt;local_port&gt;,&lt;socket_state&gt;,&lt;contextID&gt;,&lt;access_mode&gt;]</b> <b>[...]</b>  <b>OK</b>  若有任何错误： <b>ERROR</b>
设置命令 若<query_type>为 0，查询指定场景的连接状态 <b>AT+QISTATE=&lt;query_type&gt;,&lt;context ID&gt;</b>	响应 返回指定场景里所有现存连接的状态： <b>[+QISTATE: &lt;connectID&gt;,&lt;service_type&gt;,&lt;IP_address&gt;/&lt;domain_name&gt;,&lt;remote_port&gt;,&lt;local_port&gt;,&lt;socket_state&gt;,&lt;contextID&gt;,&lt;access_mode&gt;]</b> <b>[...]</b>  <b>OK</b>  若有任何错误： <b>ERROR</b>
设置命令 若<query_type>为 1，查询指定 Socket 的连接状态 <b>AT+QISTATE=&lt;query_type&gt;,&lt;connectID&gt;</b>	响应 返回指定 Socket 的连接状态： <b>[+QISTATE: &lt;connectID&gt;,&lt;service_type&gt;,&lt;IP_address&gt;/&lt;domain_name&gt;,&lt;remote_port&gt;,&lt;local_port&gt;,&lt;socket_state&gt;,&lt;contextID&gt;,&lt;access_mode&gt;]</b>  <b>OK</b>  若有任何错误： <b>ERROR</b>
最大响应时间	300 毫秒
保存机制	/

## 参数

<b>&lt;query_type&gt;</b>	整型。查询类型。 0 查询指定<contextID>的连接状态 1 查询指定<connectID>的连接状态
<b>&lt;contextID&gt;</b>	整型。场景 ID；范围：1-3。
<b>&lt;connectID&gt;</b>	整型。Socket ID；范围：0-4。
<b>&lt;service_type&gt;</b>	字符串类型。Socket 服务类型。 "TCP" 客户端创建 TCP 连接 "UDP" 客户端创建 UDP 连接
<b>&lt;IP_address&gt;</b>	字符串类型。远程客户端的 IP 地址。
<b>&lt;domain_name&gt;</b>	字符串类型。远程服务器的域名地址。
<b>&lt;remote_port&gt;</b>	整型。远程服务器的端口号。
<b>&lt;local_port&gt;</b>	整型。分配的本地端口号。
<b>&lt;socket_state&gt;</b>	整型。Socket 状态。 0 "Initial"：客户端连接尚未建立 1 "Connecting"：客户端正在连接 2 "Connected"：客户端连接已经创建成功 3 "Closing"：客户端连接正在关闭 4 "Remote Closing"：远程服务器正在关闭客户端连接
<b>&lt;access_mode&gt;</b>	整型。数据访问模式。 0 缓存模式 1 直吐模式

## 备注

1. 目前仅支持<contextID>=1。
2. 若响应中无+QISTATE: 列表，则说明当前没有连接。

### 2.3.4. AT+QISEND 发送十六进制/文本字符串数据

该命令用于通过指定连接发送十六进制或文本字符串格式的 Socket 数据。

AT+QISEND 发送十六进制/文本字符串数据	
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),&lt;data&gt;</b>  <b>OK</b>
设置命令 <b>AT+QISEND=&lt;connectID&gt;,&lt;send_length&gt;,&lt;data&gt;</b>	响应 若数据发送成功： <b>OK</b>

	<p><b>SEND OK</b></p> <p>若数据发送失败： <b>OK</b></p> <p><b>SEND FAIL</b></p> <p>若有任何错误： <b>ERROR</b></p>
<p>设置命令</p> <p>数据模式下发送不定长数据</p> <p><b>AT+QISEND=&lt;connectID&gt;</b></p> <p>响应 &gt; 后，模块将进入数据模式；之后可直接输入待发数据，按 <b>Ctrl+Z</b> 发送数据，按 <b>Esc</b> 取消发送。</p>	<p>响应</p> <p>若数据发送成功： <b>OK</b></p> <p><b>SEND OK</b></p> <p>若数据发送失败： <b>OK</b></p> <p><b>SEND FAIL</b></p> <p>若有任何错误： <b>ERROR</b></p>
<p>设置命令</p> <p>数据模式下发送定长数据</p> <p><b>AT+QISEND=&lt;connectID&gt;,&lt;send_length&gt;</b></p> <p>响应 &gt; 后，模块将进入数据模式；之后输入长度等于 <b>&lt;send_length&gt;</b> 的待发数据。</p>	<p>响应</p> <p>若数据发送成功： <b>OK</b></p> <p><b>SEND OK</b></p> <p>若数据发送失败： <b>OK</b></p> <p><b>SEND FAIL</b></p> <p>若有任何错误： <b>ERROR</b></p>
<p>设置命令</p> <p>查询已应答和已发送但未应答数据的总长度（包含通过 <b>AT+QISENDEX</b> 和 <b>AT+QISEND</b> 发送的全部数据）</p> <p><b>AT+QISEND=&lt;connectID&gt;,0</b></p>	<p>响应</p> <p><b>+QISEND: &lt;sent&gt;,&lt;acked&gt;,&lt;nAked&gt;</b></p> <p><b>OK</b></p> <p>若有任何错误： <b>ERROR</b></p>
最大响应时间	300 毫秒

保存机制

/

## 参数

<connectID>	整型。Socket ID；范围：0-4。
<send_length>	整型。待发数据长度，单位：字节。 在非数据模式（普通模式）下，文本格式数据的最大长度 1024 字节，十六进制格式数据的最大长度是 512 字节。 在数据模式下，文本格式数据的最大长度是 1460 字节，十六进制格式数据的最大长度是 730 字节。
<data>	字符串类型。待发送的十六进制或文本格式数据。
<sent>	整型。通过会话已发送数据的总长度；单位：字节。
<acked>	整型。远程服务器已应答数据的总长度；仅 TCP 会话时可用。
<nAcked>	整型。已发送但是还未被远程服务器应答的数据总长度；仅 TCP 会话时可用。

## 备注

1. **SEND OK** 仅表示数据已经发送至协议栈。
2. 参数<send\_length>的值和<data>的实际长度必须一致。
3. <data>为特殊格式（如 JSON）字符时，必须为带双引号的字符串。目前，<data>不支持特殊命令字符，如分号（;）。
4. MCU 应在收到 **SEND OK** 或者 **SEND FAIL** 响应后再继续发送下一条数据。

### 2.3.5. AT+QIRD 读取 TCP/IP 数据

该命令用于读取从指定连接接收的 Socket 数据。

在缓存模式下，模块接收到数据后会先缓存数据，并向外部 MCU 上报 URC **+QIURC: "recv",<connectID>[,<current\_recv\_length>]**以告知接收到新的下行数据。

#### AT+QIRD 读取 TCP/IP 数据

测试命令 <b>AT+QIRD=?</b>	响应 <b>+QIRD: (支持的&lt;connectID&gt;范围),(支持的&lt;read_length&gt;范围)</b>  <b>OK</b>
设置命令 <b>AT+QIRD=&lt;connectID&gt;,&lt;read_length&gt;</b>	响应 若指定连接收到数据： <b>+QIRD: &lt;actual_read_length&gt;[,&lt;remaining_length&gt;] &lt;data&gt;</b>  <b>OK</b>

	若没有收到数据： <b>+QIRD: 0</b>  <b>OK</b>  若有任何错误： <b>ERROR</b>
最大响应时间	300 毫秒
保存机制	/

## 参数

<connectID>	整型。Socket ID；范围：0-4。
<read_length>	整型。待读取数据的最大长度；范围：1-512；单位：字节。
<actual_read_length>	整型。已读取数据的长度；单位：字节。
<remaining_length>	整型。剩余数据的长度；单位：字节。
<data>	已读取的数据。

## 备注

1. 接收缓存池仍有数据未读取时，模块若再接收到下行数据将不会上报新的 URC；只有当缓存池中所有数据均被读取完成后，才会上报新的 URC。
2. 配置 **AT+QICFG="showlength",1**（详情请参考第 2.1.12 章节）后，模块才会在缓存模式下显示该命令返回值中的可选参数 **<remaining\_length>**。
3. 剩余数据长度并非缓存池中已接收数据的总长度，仅表示某一个节点中保存的当前剩余数据数据。

### 2.3.6. AT+QISENDEX 发送十六进制字符串数据

该命令用于通过指定连接发送十六进制字符串 Socket 数据。

AT+QISENDEX 发送十六进制字符串数据	
测试命令 <b>AT+QISENDEX=?</b>	响应 <b>+QISENDEX: ( 支持的 &lt;connectID&gt; 范围 ),( 支持的 &lt;send_length&gt;范围),&lt;hex_string&gt;</b>  <b>OK</b>
设置命令 <b>AT+QISENDEX=&lt;connectID&gt;,&lt;send_length&gt;,&lt;hex_string&gt;</b>	响应 若数据发送成功： <b>OK</b>  <b>SEND OK</b>

	若数据发送失败： <b>OK</b>  <b>SEND FAIL</b>  若有任何错误： <b>ERROR</b>
最大响应时间	300 毫秒
保存机制	/

## 参数

<connectID>	整型。Socket ID；范围：0-4。
<send_length>	整型。待发数据的长度，最大长度为 512 字节。
<hex_string>	字符串类型。待发送数据。

## 备注

1. **SEND OK** 仅表示数据已经发送至协议栈。
2. MCU 应在收到 **SEND OK** 或者 **SEND FAIL** 响应后再继续发送下一条数据。

### 2.3.7. AT+QISWTMD 切换数据访问模式

打开新 Socket 时，模块可通过 **AT+QIOPEN** 的<access\_mode>参数指定数据访问模式；之后可通过此命令切换数据访问模式。

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_mode&gt;列表)</b>  <b>OK</b>
查询命令 <b>AT+QISWTMD?</b>	响应 <b>OK</b>
设置命令 <b>AT+QISWTMD=&lt;connectID&gt;,&lt;access_mode&gt;</b>	响应 <b>OK</b>  若有任何错误： <b>ERROR</b>
最大响应时间	300 毫秒



保存机制	参数配置不保存至 NVRAM；深休眠唤醒后仍有效
------	--------------------------

## 参数

<connectID>	整型。Socket ID；范围：0-4。
<access_mode>	整型。Socket 的数据访问模式。
0	缓存模式
1	直吐模式

## 2.3.8. AT+QPING Ping 远程服务器

该命令用于测试主机设备的 IP 地址是否可达。

AT+QPING Ping 远程服务器	
测试命令 <b>AT+QPING=?</b>	<p>响应</p> <p><b>+QPING:</b> ( 支持的 &lt;contextID&gt; 范围 ), "&lt;host&gt;" [, ( 支持的 &lt;timeout&gt; 范围 ) [, ( 支持的 &lt;ping_num&gt; 列表 ) [, ( 支持的 &lt;ping_size&gt; 列表 ) ] ] ]</p> <p><b>OK</b></p>
设置命令 <b>AT+QPING=&lt;contextID&gt;,"&lt;host&gt;" [,&lt;time_out&gt; [,&lt;ping_num&gt; [,&lt;ping_size&gt; ] ] ]</b>	<p>响应</p> <p>若远程服务器 Ping 成功：</p> <p><b>OK</b></p> <p><b>+QPING:</b> &lt;result&gt; [,&lt;IP_address&gt; ,&lt;bytes&gt; ,&lt;time&gt; ,&lt;ttl&gt; ]</p> <p>[...]</p> <p><b>+QPING:</b> &lt;finresult&gt; [,&lt;sent&gt; ,&lt;rcvd&gt; ,&lt;lost&gt; ,&lt;min&gt; ,&lt;max&gt; ,&lt;avg&gt; ]</p> <p>若有任何错误：</p> <p><b>ERROR</b></p>
最大响应时间	300 毫秒
保存机制	/

## 参数

<contextID>	整型。场景 ID；范围：1-3。
<host>	字符串类型。远程服务器地址，格式为域名或者点分十进制 IP 地址。最大长度：150 字

	节。
<time_out>	整型。每次 Ping 请求后等待响应的最大时长；范围：1-255；默认值：4；单位：秒。
<ping_num>	整型。发送 Ping 请求最大次数；范围：1-10；默认值：4。
<ping_size>	整型。每次 Ping 请求的大小；范围：32-200；默认值：32；单位：字节。
<result>	整型。每次 Ping 请求的结果。 0 接收到远程服务器的 Ping 响应 其他 错误码；请参考第3章获取详细错误码信息
<IP_address>	字符串类型。远程服务器的 IP 地址，格式为点分十进制。
<bytes>	整型。每次发送的 Ping 请求长度；单位：字节。
<time>	整型。发送 Ping 请求花费的时间；单位：毫秒。
<tll>	整型。Ping 请求的 TTL 值。
<finresult>	整型。Ping 操作的最终结果。 0 Ping 成功 其他 错误码；请参考第3章获取详细错误码信息
<sent>	整型。Ping 请求发送的字节总长度。
<rcvd>	整型。Ping 响应中接收到的字节总长度。
<lost>	整型。Ping 请求中丢失的字节总长度。
<min>	整型。最小响应时间；单位：毫秒。
<max>	整型。最大响应时间；单位：毫秒。
<avg>	整型。平均响应时间；单位：毫秒。

## 备注

目前仅支持<contextID>=1、不支持 IPv6。

### 2.3.9. AT+QNTTP 通过 NTP 服务器同步本地时间

该命令通过 NTP 服务器同步本地时间为世界标准时间（UTC）。有关错误码<err>的说明，请参阅第3章。

AT+QNTTP 通过 NTP 服务器同步本地时间	
测试命令 AT+QNTTP=?	响应 +QNTTP: (支持的<contextID>范围),"<server>"[,<port>[(支持的<auto_set_time>列表)]]
设置命令 AT+QNTTP=<contextID>,"<server>"[,<port>[,<auto_set_time>]]	响应 若同步成功： OK  +QNTTP: <err>,<time>

	若有任何错误： <b>ERROR</b>
最大响应时间	300 毫秒
保存机制	/

参数

<contextID>	整型。场景 ID；范围：1-3。
<server>	字符串类型。NTP 服务器地址，格式为域名或者点分十进制 IP 地址。最大长度：150 字节。
<port>	整型。NTP 服务器端口号；默认值：123。
<auto_set_time>	整型。是否自动同步本地时间为 UTC。 0 不自动同步 1 自动同步
<time>	字符串类型。从 NTP 服务器上同步的时间。 格式为："YY/MM/DD,hh:mm:ss±zz"，各字符分别表示年（YY）、月（MM）、日（DD）、时（hh）、分（mm）、秒（ss）、时区（±zz）。±zz 的范围为-47 ~ +48。

备注

1. 目前仅支持<contextID>=1。
  2. 当<autosettime>=1，RTC 会自动更新到同步时间；随后可使用 **AT+QCCLK?** 查询更新后的时间。
  3. 注网成功后，模块会自动更新 RTC 时间。

2.3.10. AT+QIDNSGIP 转换域名为 IP 地址

该命令用于将指定域名转换为 IP 地址。有关错误码<err>的说明，请参阅第3章。

AT+QIDNSGIP 转换域名为 IP 地址	
测试命令 AT+QIDNSGIP=?	响应 +QIDNSGIP: (支持的<contextID>范围),"<hostname> "  OK
设置命令 AT+QIDNSGIP=<contextID>,"<hostname>"	响应 OK  +QIURC: "dnsgip",<err>,<IP_count>,<DNS_ttl> [+QIURC: "dnsgip",<hostIPaddr>] [...]

	若有任何错误： <b>ERROR</b>
最大响应时间	300 毫秒
保存机制	/

## 参数

<contextID>	整型。场景 ID；范围：1-3。
<hostname>	字符串类型。域名。最大长度：150 字节。
<IP_count>	整型。参数<hostname>对应的 IP 地址个数。
<DNS_ttl>	整型。DNS 的 TTL 值。
<hostIPAddr>	字符串类型。<hostname>的 IP 地址。

## 备注

目前仅支持<contextID>=1

### 2.3.11. AT+QIDNSCFG 配置 DNS 服务器地址

该命令用于配置主要和次要 DNS 服务器地址。

AT+QIDNSCFG 配置 DNS 服务器地址	
测试命令 <b>AT+QIDNSCFG=?</b>	响应 <b>+QIDNSCFG: (支持的&lt;contextID&gt;范围),&lt;pridnsaddr&gt;,&lt;secdnsaddr&gt;</b>  <b>OK</b>
设置命令 配置主要和次要 DNS 服务器地址 <b>AT+QIDNSCFG=&lt;contextID&gt;,&lt;pridnsaddr&gt;[,&lt;secdnsaddr&gt;]</b>	响应 <b>OK</b>  若有任何错误： <b>ERROR</b>
设置命令 DNS 服务器地址成功配置后，查询主要和次要 DNS 服务器地址 <b>AT+QIDNSCFG=&lt;contextID&gt;</b>	响应 <b>[+QIDNSCFG: &lt;contextID&gt;,&lt;pridnsaddr&gt;[,&lt;secdnsaddr&gt;]]</b>  <b>OK</b>  若有任何错误：

	ERROR
最大响应时间	300 毫秒
保存机制	<pridnsaddr>和<secdnsaddr>参数配置不保存至 NVRAM； 深休眠唤醒后仍有效

## 参数

<contextID>	整型。场景 ID；范围：1-3。
<pridnsaddr>	字符串类型。主要 DNS 服务器的 IP 地址。最大长度：64 字节。
<secdnsaddr>	字符串类型。次要 DNS 服务器的 IP 地址。最大长度：64 字节。

## 备注

1. 目前仅支持<contextID>=1。
2. 若网络仅支持 IPv4，则只能设置 IPv4 DNS 地址；若网络仅支持 IPv6，则只能设置 IPv6 DNS 地址。
3. 模块开机后，需等 IP 地址 URC（例如：+IP: 10.18.237.42，表明模块注网成功）上报完成后方可配置 DNS 服务器地址。

## 2.3.12. AT+QICFG 配置可选参数

该命令可为 TCP/IP 各项功能配置可选参数。

AT+QICFG 配置可选参数	
测试命令 AT+QICFG=?	<p>响应</p> <p>+QICFG: "dataformat", (支持的 &lt;send_data_format&gt; 列表), (支持的 &lt;recv_data_format&gt; 列表)</p> <p>+QICFG: "viewmode", (支持的 &lt;view_mode&gt; 列表)</p> <p>+QICFG: "showlength", (支持的 &lt;show_length_mode&gt; 列表)</p> <p>+QICFG: "echomode", (支持的 &lt;echo_mode&gt; 列表)</p> <p>OK</p>
设置命令 设置发送/接收数据的格式 AT+QICFG="dataformat", [<send_data_format>, <recv_data_format>]	<p>响应</p> <p>若缺省参数&lt;send_data_format&gt;和&lt;recv_data_format&gt;，则查询当前配置：</p> <p>+QICFG: "dataformat", &lt;send_data_format&gt;, &lt;recv_data_format&gt;</p> <p>OK</p> <p>若不缺省参数&lt;send_data_format&gt;和&lt;recv_data_format&gt;，则设置发送数据格式和接收数据格式：</p>

	<p><b>OK</b></p> <p>若有任何错误:</p> <p><b>ERROR</b></p>
<p>设置命令</p> <p>设置已接收数据的输出格式</p> <p><b>AT+QICFG="viewmode"[,&lt;view_mode&gt;]</b></p>	<p>响应</p> <p>若缺省参数&lt;view_mode&gt;, 则查询当前配置:</p> <p><b>+QICFG: "viewmode",&lt;view_mode&gt;</b></p> <p><b>OK</b></p> <p>若不缺省参数&lt;view_mode&gt;, 则设置已接收数据的输出格式:</p> <p><b>OK</b></p> <p>若有任何错误:</p> <p><b>ERROR</b></p>
<p>设置命令</p> <p>设置在缓存模式下是否显示可选数据长度参数<sup>1)</sup></p> <p><b>AT+QICFG="showlength"[,&lt;show_length_mode&gt;]</b></p>	<p>响应</p> <p>若缺省参数&lt;show_length_mode&gt;, 则查询当前配置:</p> <p><b>+QICFG: "showlength",&lt;show_length_mode&gt;</b></p> <p><b>OK</b></p> <p>若不缺省参数&lt;show_length_mode&gt;, 则设置在缓存模式下是否显示可选数据长度参数:</p> <p><b>OK</b></p> <p>若有任何错误:</p> <p><b>ERROR</b></p>
<p>设置命令</p> <p>设置在数据模式下是否向 UART 回显输入数据</p> <p><b>AT+QICFG="echomode"[,&lt;echo_mode&gt;]</b></p>	<p>响应</p> <p>若缺省参数&lt;echo_mode&gt;, 则查询当前配置:</p> <p><b>+QICFG: "echomode",&lt;echo_mode&gt;</b></p> <p><b>OK</b></p> <p>若不缺省参数&lt;echo_mode&gt;, 则设置回显模式:</p> <p><b>OK</b></p> <p>若有任何错误:</p> <p><b>ERROR</b></p>
最大响应时间	300 毫秒
保存机制	<ol style="list-style-type: none"> <li>1. &lt;send_data_format&gt;、&lt;recv_data_format&gt;、&lt;view_mode&gt;和&lt;show_length_mode&gt;参数配置自动保存至 NVRAM; 深休眠唤醒后仍有效。</li> <li>2. &lt;echo_mode&gt;参数配置不保存至 NVRAM; 深休眠唤醒后</li> </ol>

无效。

## 参数

<b>&lt;send_data_format&gt;</b>	整型。发送数据的格式。 0 文本格式 1 十六进制格式
<b>&lt;recv_data_format&gt;</b>	整型。接收数据的格式。 0 文本格式 1 十六进制格式
<b>&lt;view_mode&gt;</b>	整型。已接收数据的输出格式。 0 已接收数据输出格式: data header\r\n\data 1 已接收数据输出格式: data header,data
<b>&lt;show_length_mode&gt;</b>	整型。在缓存模式下是否显示可选数据长度参数 <sup>1)</sup> 。 0 不显示 1 显示
<b>&lt;echo_mode&gt;</b>	整型。是否在数据模式下向 UART 回显输入数据。 0 不回显 1 回显

## 备注

- 1) 所述可选数据长度参数包含:
  - 下行数据已达 URC +QIURC: "recv",<connectID>[,<current\_recv\_length>]中的参数<current\_recv\_length>;
  - AT+QIRD 命令返回值中的参数<remaining\_length>。
- 参数<echo\_mode>仅在数据模式下才有效,且仅 BC26NxR01A07/BC20NxR01A08 及之后的版本中才支持该参数。

### 2.3.13. AT+QIGETERROR 查询上一个错误码

该命令用于查询上一个 TCP/IP 命令返回的错误码及其相应描述。有关错误码<err>的说明,请参阅第 3 章。

AT+QIGETERROR 查询上一个错误码	
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>

	若有任何错误: <b>ERROR</b>
最大响应时间	300 毫秒

## 参数

<errcode_description>	字符串类型。详细的错误码描述，具体请参考第3章。
-----------------------	--------------------------

## 2.4. URC 详解

TCP/IP AT 命令的 URC 统一以<CR><LF>+QIURC: <type>[...]<CR><LF>的格式上报。本文中 URC 前后的<CR><LF>均将省略。

## 备注

1. 当模块在 PSM 模式下，不会上报 URC。
2. 当模块 Modem 在 DRX 或者 eDRX 模式下，URC 上报会有延迟，延迟时间依据寻呼周期而定。
3. 当模块 Modem 在连接状态下，会正常上报 URC。
4. URC 最大长度为 1400 字节。
5. 使用 UDP 直吐模式接收数据时，模块将丢弃总长度大于 1400 字节的 URC。

### 2.4.1. 连接断开 URC

当 TCP Socket 被远程关闭或者因为网络异常而断开时，模块将上报 URC +QIURC: "closed",<connectID>，同时参数<socket\_state>（表示 Socket 状态）变为"Closing"；此时必须执行 AT+QICLOSE=<connectID>将<socket\_state>设置为"Initial"。

在缓存模式下，模块也可以执行命令 AT+QIRD=<connectID>,<read\_length>以读取缓存数据。

#### 连接断开 URC

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

## 参数

<connectID>	整型。Socket ID；范围：0-4。
-------------	----------------------



## 2.4.2. 下行数据已达 URC

当模块接收到下行数据时，会上报该 URC：

- 在缓存模式下，URC 格式为+QIURC: "recv",<connectID>[,<current\_recv\_length>];
- 在直吐模式下，URC 格式为+QIURC: "recv",<connectID>,<current\_recv\_length><CR><LF><data>。

### 下行数据已达 URC

+QIURC: "recv",<connectID>[,<current_recv_length>]	表示缓存模式下有下行数据
+QIURC: "recv",<connectID>,<current_recv_length><CR><LF><data>	表示直吐模式下有下行数据

### 参数

<connectID>	整型。Socket ID；范围：0-4。
<current_recv_length>	整型。实际接收到的数据长度。
<data>	接收到的数据。

### 备注

配置 AT+QICFG="showlength",1（详情请参考第 2.1.12 章节）后，缓存模式下上报的下行数据已达 URC 才会包含可选参数<current\_recv\_length>。

## 2.4.3. 下行数据缓存已满 URC

在缓存模式下，如果已经没有缓存资源可以分配给下行数据，模块会上报该 URC。

### 下行数据缓存已满 URC

+QIURC: "recv",<connectID>,"buff full"	表示下行数据缓存已满。
----------------------------------------	-------------

### 参数

<connectID>	整型。Socket ID；范围：0-4。
-------------	----------------------

### 备注

缓存模式下，数据可分配的最大节点数为 20。

## 3 错误码

执行 TCP/IP AT 命令后若有返回 **ERROR** 或 **<err>**，则具体的错误码信息可通过 **AT+QIGETERROR** 命令查询。**AT+QIGETERROR** 仅返回最后一个 TCP/IP AT 命令的错误码。

表 2：错误码列表

错误码	错误码描述	含义
0	Operation successful	操作成功
550	Unknown error	未知原因
551	Operation blocked	操作受阻
552	Invalid parameters	无效参数
553	Memory not enough	内存不足
554	Create socket 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	Open PDP context failed	打开 PDP 场景失败
562	Close PDP context failed	关闭 PDP 场景失败
563	Socket identity has been used	Socket ID 被占用
564	DNS busy	DNS 繁忙
565	DNS parse failed	NDS 解析失败

566	Socket connection failed	Socket 连接失败
567	Socket has been closed	Socket 已被关闭
568	Operation busy	操作繁忙
569	Operation timeout	操作超时
570	PDP context broken down	PDP 场景发生故障
571	Cancel send	取消发送
572	Operation not allowed	操作不允许
573	APN not configured	未配置 APN
574	Port busy	端口繁忙

## 4 举例

### 4.1. TCP 客户端在缓存模式下工作

#### 4.1.1. 创建一个 TCP 客户端连接并进入缓存模式

//打开一个 Socket: 场景 ID 为 1、Socket ID 为 0。

**AT+QIOPEN=1,0,"TCP","220.180.239.212",8062,1234,0**

OK

**+QIOPEN: 0,0**

//连接成功，建议等 URC 上报（超时时间：60 秒）后再进行其他操作。

**AT+QISTATE=1,0**

//查询当前 Socket（Socket ID 为 0）的连接状态。

**+QISTATE: 0,"TCP","220.180.239.212",8062,1234,2,1,0**

OK

#### 4.1.2. 在缓存模式下发送数据

**AT+QISEND=0,10,1234567890**

//发送数据，数据长度为 10 字节。

OK

SEND OK

**AT+QISEND=0**

//发送不定长数据。

>

**1234567890<ctrl+Z>**

OK

SEND OK

**AT+QISEND=0,10**

//发送定长数据，数据长度为 10 字节。

>

**1234567890**

OK

SEND OK

```
AT+QISENDEX=0,5,"3031323334" //发送十六进制字符串数据。
OK
SEND OK
```

#### 4.1.3. 在缓存模式下从远程服务器上读取数据

```
+QIURC: "recv",0 //ID 为 0 的 Socket 接收到下行数据。

AT+QIRD=0,512 //读取 512 字节数据。
+QIRD: 10 //实际读取的数据长度为 10 字节。
1234567890 //所读取的数据内容。

OK

AT+QIRD=0,512 //读取数据，数据长度为 512 字节。
+QIRD: 0 //无缓存数据。

OK

AT+QICFG="showlength",1 //使能在缓存模式下显示可选参数<remaining_length>
和<current_recv_length>。

OK

+QIURC: "recv",0,12 //ID 为 0 的 Socket 接收到 12 字节下行数据。

AT+QIRD=0,10 //读取 10 字节数据。
+QIRD: 10,2 //已读取 10 字节数据，剩余还有 2 字节未读取。
1234567890

OK

+QIURC: "recv",0,"buff full" //ID 为 0 的 Socket 上报缓存已满，模块需要使用 AT+QIRD 读取
缓存数据。

AT+QICFG="viewmode",1 //已接收数据的输出格式：data header,data。
OK

AT+QISEND=0,12,"012345678901"
OK
SEND OK
```

```
+QIURC: "recv",0,12

AT+QIRD=0,10
+QIRD: 10,2,0123456789

OK
```

#### 4.1.4. 断开连接

```
AT+QICLOSE=0      //断开 ID 为 0 的 Socket 连接。
OK

CLOSE OK
```

### 4.2. TCP 客户端在直吐模式下工作

#### 4.2.1. 创建一个 TCP 客户端连接并进入直吐模式

```
//打开一个 Socket: 场景 ID 为 1、Socket ID 为 0。
AT+QIOPEN=1,0,"TCP","220.180.239.212",8062,0,1
OK

+QIOPEN: 0,0      //连接成功, 建议等 URC 上报 (超时时间: 60 秒) 后再进行其他操作。

AT+QISTATE=1,0    //查询当前 Socket (Socket ID 为 0) 的连接状态。
+QISTATE: 0,"TCP","220.180.239.212",8062,0,2,1,1

OK
```

#### 4.2.2. 在直吐模式下发送数据

```
AT+QISEND=0,5,12345      //发送数据, 数据长度为 5 字节。
OK

SEND OK

AT+QISEND=0      //发送不定长数据。
>
12345<ctrl+Z>
OK

SEND OK
```

**AT+QISEND=0,5** //发送定长数据，数据长度为 5 字节。

>

12345

OK

SEND OK

**AT+QISENDEX=0,5,"3132333435"** //发送十六进制字符串数据。

OK

SEND OK

#### 4.2.3. 在直吐模式下接收来自远程服务器的数据

**+QIURC: "recv",0,5** //ID 为 0 的 Socket 接收到 5 字节下行数据。

12345

//所接收到的数据。

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

//已接收数据的输出格式：data header,data。

OK

**AT+QISEND=0,12,"012345678901"**

OK

SEND OK

**+QIURC: "recv",0,12,"012345678901"**

#### 4.2.4. 断开连接

**AT+QICLOSE=0** //断开 ID 为 0 的 Socket 连接。

OK

CLOSE OK

### 4.3. Ping 远程服务器

**AT+QPING=1,"iot.quectel.com"**

OK

**+QPING: 0,"47.100.63.174",32,560,88**

```
+QPING: 0,"47.100.63.174",32,220,88
+QPING: 0,"47.100.63.174",32,230,88
+QPING: 0,"47.100.63.174",32,280,88
+QPING: 0,4,4,0,220,560,322
```

#### 4.4. 同步本地时间

```
AT+QNTTP=1,"ntp5.aliyun.com" //使用域名为 ntp5.aliyun.com 的 NTP 服务器同步本地时间。
OK
+QNTTP: 0,"19/06/11,11:08:20:35+32"
```

#### 4.5. 配置 DNS 服务器

```
AT+QIDNSCFG=1,"218.2.2.2","8.8.8.8"
OK
AT+QIDNSCFG=1
+QIDNSCFG: 1,"218.2.2.2","8.8.8.8"
OK
```

#### 4.6. 查询上一个错误码

```
AT+QIOPEN=1,"UDP","220.180.239.212",8063,0,1 //打开任意一个 Socket。
ERROR
AT+QIGETERROR
+QIGETERROR: 552,invalid parameters
OK
```



# 5 附录 A 术语缩写

表 3：术语缩写

术语	英文全称	中文全称
eDRX	extended Discontinuous Reception	扩展不连续接收
DNS	Domain Name System	域名系统
DRX	Discontinuous Reception	非连续接收
ID	Identifier	标识符
IP	Internet Protocol	互联网协议
IPv4	Internet Protocol version 4	互联网协议版本 4
IPv6	Internet Protocol version 6	互联网协议版本 6
MCU	Micro Controller Unit	微控制器
ME	Mobile Equipment	移动设备
NB-IoT	Narrow Band Internet of Things	窄带物联网
NTP	Network Time Protocol	网络时间协议
PPP	Point to Point Protocol	点对点协议
PSM	Power Saving Mode	省电模式
TCP	Transmission Control Protocol	传输控制协议
TTL	Time to Live	生存时间
UDP	User Datagram Protocol	用户数据报协议
URC	Unsolicited Result Code	非请求结果码
UTC	Universal Time Coordinated	世界标准时间