

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_V1.1

日期: 2019-06-25

状态: 临时文件

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

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

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 | 2018-07-27 | 陶亮亮 | 初始版本 |
| 1.1 | 2019-06-25 | 王瑞 蒋涛 | <ol style="list-style-type: none"> 更新 AT 命令 AT+QISTATE; 为 AT 命令 AT+QISEND 增加一些设置命令; 更新 AT 命令 AT+QNTTP 参数<server>和<time>的说明; 增加 AT 命令 AT+QIDNSCFG 用来使能 DNS 服务器地址配置; 增加 AT 命令 AT+QICFG="echomode"[,<echo_mode>]用来使能数据回显模式配置。 |

目录

| | |
|---|-----------|
| 文档历史 | 2 |
| 目录 | 3 |
| 表格索引 | 4 |
| 1 引言 | 5 |
| 1.1. 使用 TCP/IP AT 命令的流程..... | 5 |
| 1.2. 数据访问模式说明 | 5 |
| 2 TCP/IP AT 命令详解 | 6 |
| 2.1. AT 命令介绍 | 6 |
| 2.1.1. AT+QIOPEN 打开 Socket 服务 | 6 |
| 2.1.2. AT+QICLOSE 关闭 Socket 服务 | 7 |
| 2.1.3. AT+QISTATE 查询 Socket 服务状态..... | 8 |
| 2.1.4. AT+QISEND 发送十六进制/文本字符串数据..... | 9 |
| 2.1.5. AT+QIRD 读取收到的 TCP/IP 数据 | 11 |
| 2.1.6. AT+QISENDEX 发送十六进制字符串数据 | 12 |
| 2.1.7. AT+QISWTMD 切换数据访问模式 | 13 |
| 2.1.8. AT+QPING Ping 远程服务器..... | 14 |
| 2.1.9. AT+QNTP 通过 NTP 服务器同步本地时间 | 15 |
| 2.1.10. AT+QIDNSGIP 通过域名获取 IP 地址..... | 16 |
| 2.1.11. AT+QIDNSCFG 配置 DNS 服务器地址..... | 17 |
| 2.1.12. AT+QICFG 配置可选参数..... | 18 |
| 2.1.13. AT+QIGETERROR 查询上一个错误代码..... | 19 |
| 2.2. URC 说明 | 20 |
| 2.2.1. 连接断开 URC | 20 |
| 2.2.2. 下行数据 URC | 21 |
| 2.2.3. 下行数据缓存已满 URC..... | 21 |
| 3 错误代码..... | 22 |
| 4 举例 | 24 |
| 4.1. TCP 客户端在非透传 Buffer 模式下工作 | 24 |
| 4.1.1. 创建一个 TCP 客户端连接并进入非透传 Buffer 模式 | 24 |
| 4.1.2. 在非透传 Buffer 模式下发送数据 | 24 |
| 4.1.3. 在非透传 Buffer 模式下从远程服务器上接收数据 | 25 |
| 4.1.4. 断开连接 | 26 |
| 4.2. TCP 客户端在非透传 Push 模式下工作 | 26 |
| 4.2.1. 创建一个 TCP 客户端连接并进入非透传 Push 模式 | 26 |
| 4.2.2. 在非透传 Push 模式下发送数据 | 26 |
| 4.2.3. 在非透传 Push 模式下接收来自远程服务器的数据 | 27 |
| 4.2.4. 断开连接 | 27 |
| 4.3. Ping 远程服务器 | 27 |
| 4.4. 同步本地时间..... | 28 |

| | | |
|------|-----------------|----|
| 4.5. | 配置 DNS 服务器..... | 28 |
| 4.6. | 获取上一个错误代码..... | 28 |
| 5 | 附录 A..... | 29 |

表格索引

| | |
|-------------------|----|
| 表 1: 错误代码列表 | 22 |
| 表 2: 术语缩写 | 29 |

1 引言

Quectel BC26 模块和 BC20 模块内置 TCP/IP 协议栈使得模块可以直接通过 AT 命令访问网络，它可以极大地降低对 PPP 以及外部 TCP/IP 协议栈的依赖，从而降低成本。

BC26&BC20 模块可提供 TCP 客户端、UDP 客户端等 Socket 服务。

该文档适用于以下 Quectel 模块：

- BC26
- BC20

1.1. 使用 TCP/IP AT 命令的流程

通过 TCP/IP AT 命令，模块可以打开/关闭 Socket 服务，可以通过 Socket 服务发送/接收数据。

1.2. 数据访问模式说明

BC26&BC20 模块支持下面两种数据访问模式：

- 非透传 Buffer 模式
- 非透传 Push 模式

当通过 AT+QIOPEN 打开一个 Socket 服务时，可以通过参数<access_mode>来指定数据访问模式。当 Socket 服务成功打开，可以通过 AT+QISWTMD 改变数据访问模式。

1. 非透传 Buffer 模式下，可通过命令 AT+QISEND/AT+QISENDEX 发送数据。当接收到数据时，模块会缓存所接收的数据，并有 URC 上报 “+QIURC: "recv",<connectID>[,<current_rcv_length>]”，模块可以通过命令 AT+QIRD 来读取缓存数据。值得注意的是，在非透传 Buffer 模式下，如果缓存空间仍有数据，模块必须先通过 AT+QIRD 读取完缓存空间里的所有数据，否则收到新数据时无 URC 上报。
2. 非透传 Push 模式下，可通过命令 AT+QISEND/AT+QISENDEX 发送数据。模块新接收的数据会通过上报 URC “+QIURC: "recv",<connectID>,<current_rcv_length><CR><LF><data>” 直接输出。

2 TCP/IP AT 命令详解

2.1. AT 命令介绍

2.1.1. AT+QIOPEN 打开 Socket 服务

该命令用来打开 Socket 服务。可通过参数 <service_type> 来指定服务类型，并通过参数 <access_mode>来指定数据访问模式。执行命令以后，会有 URC 上报 “+QIOPEN: <connectID>,<err>” 说明 Socket 服务是否打开成功。

| AT+QIOPEN 打开 Socket 服务 | |
|--|--|
| 测试命令 AT+QIOPEN=? | 响应 +QIOPEN: (1-3),(0-4),"TCP/UDP","<IP_address>/<domain_name>",<remote_port>,<local_port>,(0-1)[,<0-1>] OK |
| 设置命令 AT+QIOPEN=<contextID>,<connectID>,<service_type>,<IP_address>/<domain_name>,<remote_port>[,<local_port>[,<access_mode>][,<protocol_type>]] | 响应 OK +QIOPEN: <connectID>,<err> 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|----------------|---|
| <contextID> | 整型，场景 ID，范围：1-3 |
| <connectID> | 整型，Socket 服务索引，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，表示远程服务器的 IP 地址，例如：“220.18.23.22”。 |
| <domain_name> | 字符串类型，表示远程服务器的域名地址。 |
| <remote_port> | 表示远程服务器的端口号，范围：1-65535。 |

| | |
|-----------------|--|
| <local_port> | 表示本地端口号，范围：1-65535。若<local_port>值为“0”，会自动分配本地端口号，不为“0”则本地端口号为指定端口。 |
| <access_mode> | 整型，Socket 服务数据访问模式： 0 非透传 Buffer 模式 1 非透传 Push 模式 |
| <protocol_type> | 整型，互联网协议类型： 0 IPv4 1 IPv6 |

备注

1. 目前，仅支持<contextID>=1。
 2. 等待 URC 上报“+QIOPEN: <connectID>,<err>”后再进行其他操作，模块超时时间为 60 秒。
 3. 如果连接失败，需要执行命令 AT+QICLOSE=<connectID>来关闭 Socket。
 4. IP 地址 URC（例如：+IP: 10.18.237.42，表明注网成功）上报以后才能执行该命令。
 5. 创建 TCP 会话后，模块无法进入深度睡眠模式。关闭连接后，模块才能如期进入深度睡眠模式。
 6. UDP 会话创建后，模块会自动备份最新 UDP 配置信息，MCU 可以在唤醒后直接发送/接收数据。

2.1.2. AT+QICLOSE 关闭 Socket 服务

该命令用来关闭指定的 Socket 服务。

| AT+QICLOSE 关闭 Socket 服务 | |
|--------------------------------|--|
| 测试命令 AT+QICLOSE=? | 响应 +QICLOSE: (0-4) OK |
| 设置命令 AT+QICLOSE=<connectID> | 响应 若成功关闭，响应： OK CLOSE OK 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|-------------|------------------------|
| <connectID> | 整型，Socket 服务索引，范围：0-4。 |
|-------------|------------------------|

2.1.3. AT+QISTATE 查询 Socket 服务状态

该命令用来查询 Socket 服务状态。

| AT+QISTATE 查询 Socket 服务状态 | |
|---|---|
| 测试命令 AT+QISTATE=? | 响应 OK |
| 查询命令 AT+QISTATE? | 响应 返回所有现存连接的状态列表，格式如下： [+QISTATE: <connectID>,<service_type>,<IP_address>/<domain_name>,<remote_port>,<local_port>,<socket_state>,<contextID>,<access_mode>] OK 若有任何有关 ME 功能错误，响应： ERROR |
| 设置命令 若参数<query_type>值为“0”，则返回指定场景的连接状态 AT+QISTATE=<query_type>,<context ID> | 响应 返回指定场景里所有现存连接的状态清单，格式如下： [+QISTATE: <connectID>,<service_type>,<IP_address>/<domain_name>,<remote_port>,<local_port>,<socket_state>,<contextID>,<access_mode>] OK 若有任何有关 ME 功能错误，响应： ERROR |
| 设置命令 若参数<query_type>值为“1”，则返回指定 Socket 服务的连接状态 AT+QISTATE=<query_type>,<connectID> | 响应 返回指定 Socket 服务下的连接状态，格式如下： [+QISTATE: <connectID>,<service_type>,<IP_address>/<domain_name>,<remote_port>,<local_port>,<socket_state>,<contextID>,<access_mode>] OK 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

<query_type> 整型，表示查询类型。

| | |
|----------------|------------------------------------|
| | 0 根据<contextID>查询连接状态 |
| | 1 根据<connectID>查询连接状态 |
| <contextID> | 整型，场景 ID，范围：1-3。 |
| <connectID> | 整型，Socket 服务索引，范围：0-4。 |
| <service_type> | 字符串型，表示服务类型。 |
| | "TCP" 客户端创建 TCP 连接 |
| | "UDP" 客户端创建 UDP 连接 |
| <IP_address> | 字符串型，远程客户端 IP 地址。 |
| <domain_name> | 字符串型，远程服务器域名地址。 |
| <remote_port> | 整型，远程服务器的端口号。 |
| <local_port> | 整型，分配的本地端口号。 |
| <socket_state> | 整型，Socket 服务状态。 |
| | 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 |
| <access_mode> | 数据访问模式。 |
| | 0 非透传 Buffer 模式 |
| | 1 非透传 Push 模式 |

备注

1. 目前，仅支持<contextID>=1。
2. 若返回中没有“+QISTATE:”列表，则说明当前没有连接。

2.1.4. AT+QISEND 发送十六进制/文本字符串数据

该命令用来通过指定连接发送十六进制/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 AT+QISEND=? | 响应 +QISEND: (0-4),(1-1024),<data> OK |
| 设置命令 AT+QISEND=<connectID>,<send_length>,<data> | 响应 若数据发送成功，响应： OK SEND OK 否则，响应： OK |

| | |
|---|---|
| | <p>SEND FAIL</p> <p>若有任何有关 ME 功能错误，响应： ERROR</p> |
| <p>设置命令 发送不定长数据 AT+QISEND=<connectID> 待响应 “>” 后，模块进入数据模式，之后输入待发数据。按 “Ctrl+Z” 发送数据，按 “Esc” 取消发送。</p> | <p>响应 若数据发送成功，响应： OK</p> <p>SEND OK</p> <p>否则，响应 OK</p> <p>SEND FAIL</p> <p>若有任何有关 ME 功能错误，响应： ERROR</p> |
| <p>设置命令 发送定长数据 AT+QISEND=<connectID>,<send_length> 待响应 “>” 后，模块进入数据模式，之后输入长度等于<send_length>的待发数据。</p> | <p>响应 若数据发送成功，响应： OK</p> <p>SEND OK</p> <p>否则，响应 OK</p> <p>SEND FAIL</p> <p>若有任何有关 ME 功能错误，响应： ERROR</p> |
| <p>设置命令 查询已确认和已发送但未确认数据的总长度（通过命令 AT+QISEND 和 AT+QISENDEX 发送的数据） AT+QISEND=<connectID>,0</p> | <p>响应 +QISEND: <sent>,<acked>,<nAcked></p> <p>OK</p> <p>若有任何有关 ME 功能错误，响应： ERROR</p> |
| 最大响应时间 | 300 毫秒 |

参数

<connectID> 整型，Socket 服务索引，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，文本格式最大长度 1024 字节，十六进制格式最大长度是 512 字节；在数据模式下，文本格式最大长度是 1460 字节，十六进制格式最大长度是 730 字节。 |
| <data> | 字符串型，表示待发送的文本或者十六进制数据。 |
| <sent> | 整型，数字，表示通过会话已发送数据的总长度，单位：字节 |
| <acked> | 整型，数字，表示远程服务器已经确认的数据总长度，只在 TCP 会话下可用。 |
| <nAcked> | 整型，数字，表示已发送但是还没有被远程服务器确认的数据总长度，只在 TCP 会话下可用。 |

备注

1. “SEND OK” 只表示数据已经到达协议栈。
2. 参数<send_length>的值和<data>的长度必须一致。
3. 如果参数<data>值中包含特殊字符（如：JSON），需要加双引号；目前，参数<data>不支持特殊命令字符，例如分号。
4. MCU 在收到 “SEND OK” 或者 “SEND FAIL” 之后才能继续发送下一条数据。

2.1.5. AT+QIRD 读取收到的 TCP/IP 数据

该命令用来读取从指定连接中接收的 Socket 数据。

在非透传 Buffer 模式下，接收到数据以后模块会缓存数据，然后向外部 MCU 上报 URC “+QIURC: “recv”,<connectID>,<current_rcv_length>” 告知接收到新数据。

| AT+QIRD 读取收到的 TCP/IP 数据 | |
|--|---|
| 测试命令 AT+QIRD=? | 响应 +QIRD: (0-4),(1-512) OK |
| 设置命令 AT+QIRD=<connectID>,<read_length> | 响应 +QIRD: <actual_read_length>,<remaining_length> <data> OK 如果没有数据，响应： +QIRD: 0 OK 若有任何有关 ME 功能错误，响应： ERROR |

| | |
|--------|--------|
| 最大响应时间 | 300 毫秒 |
|--------|--------|

参数

| | |
|----------------------|--------------------------------|
| <connectID> | 整型，Socket 服务索引，范围：0-4。 |
| <read_length> | 整型，需要读取的数据最大长度，范围：1-512，单位：字节。 |
| <actual_read_length> | 整型，表示接收到的数据实际长度，单位：字节。 |
| <remaining_length> | 整型，表示缓存里剩余未读数据的长度，单位：字节。 |
| <data> | 读取的数据。 |

备注

| | |
|----|---|
| 1. | 接收缓存池仍有数据未读取，当模块再次接收到数据，此时不会上报 URC，除非缓存池中所有接收的数据都被读取完成。 |
| 2. | 只有当设置了 AT+QICFG="showlength",1 ，即在非透传 Buffer 模式下显示可选数据长度参数（请参考第 2.1.12 章节），下行提醒上报 URC 中的参数<current_rcv_length>和执行命令 AT+QIRD 后返回的参数<remaining_length>才会提示。 |
| 3. | 剩余数据长度不是缓存池中收到的数据总长度，它只表示在一个节点中保留的当前剩余数据。 |

2.1.6. AT+QISENDEX 发送十六进制字符串数据

该命令用来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 AT+QISENDEX=? | 响应 +QISENDEX: (0-4),(1-512),<hex_string> OK |
| 设置命令 AT+QISENDEX=<connectID>,<send_length>,<hex_string> | 响应 若发送十六进制字符串数据成功，响应： OK SEND OK 否则，响应： OK SEND FAIL 若有任何有关 ME 功能错误，响应： ERROR |

| | |
|--------|--------|
| 最大响应时间 | 300 毫秒 |
|--------|--------|

参数

| | |
|---------------|-------------------------|
| <connectID> | 整型，Socket 服务索引，范围：0-4。 |
| <send_length> | 整型，待发数据长度，最大长度为 512 字节。 |
| <hex_string> | 待发送的十六进制字符串数据。 |

备注

1. “SEND OK” 只表示数据已经到达协议栈。

2. MCU 在收到 “SEND OK” 或者 “SEND FAIL” 之后才能继续发送下一条数据。

2.1.7. AT+QISWTMD 切换数据访问模式

该命令用来切换数据访问模式：非透传 Buffer 模式或者非透传 Push 模式。当开启一个新的 Socket 服务时，模块可以通过命令 AT+QIOPEN 的参数<access_mode>来指定数据访问模式。

| AT+QISWTMD 切换数据访问模式 | |
|--|---|
| 测试命令 AT+QISWTMD=? | 响应 +QISWTMD: (0-4),(0-1) OK |
| 查询命令 AT+QISWTMD? | 响应 OK |
| 设置命令 AT+QISWTMD=<connectID>,<access_mode> | 响应 OK 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|---------------|--|
| <connectID> | 整型，Socket 服务索引，范围：0-4。 |
| <access_mode> | 整型，Socket 服务数据访问模式 0 非透传 Buffer 模式 1 非透传 Push 模式 |

备注

1. 切换数据访问模式以后会立即生效。
2. 参数<access_mode>的配置会自动保存到 NVRAM 中。

2.1.8. AT+QPING Ping 远程服务器

该命令用来测试主机 IP 地址是否可达。

| AT+QPING Ping 远程服务器 | |
|--|---|
| 测试命令 AT+QPING=? | 响应 +QPING: (1-3),"<host>"[, (1-255)][, (1-10)][, (32-200)] OK |
| 设置命令 AT+QPING=<contextID>,<host>[,<time_out>[,<ping_num>[,<ping_size>]]] | 响应 若 Ping 远程服务器成功，响应： OK +QPING: <result>[,<IP_address>,<bytes>,<time>,<ttd>] [...] +QPING: <finresult>[,<sent>,<rcvd>,<lost>,<min>,<max>,<avg>] 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|---------------------------|---|
| <contextID> | 整型，场景 ID，范围：1-3。 |
| <host> | 字符串型，Host 地址，格式为域名或者点分十进制 IP 地址。 |
| <time_out> | 整型，表示每次 Ping 请求后等待响应的最大时长。范围：1-255，单位：秒，默认值：4。 |
| <ping_num> | 整型，表示发送 Ping 请求最大次数，范围：1-10，默认值：4。 |
| <ping_size> | 整型，表示每个 Ping 请求的大小，范围：32-200，默认值：32，单位：字节。 |
| <result> | 整型，表示每次 Ping 请求的结果。 0 表示接收到来自服务器的 Ping 响应 Others 请参考第三章获取指定错误代码信息 |
| <IP_address> | 字符串型，远程服务器的 IP 地址，格式为点分十进制。 |
| <bytes> | 整型，表示每次发送 Ping 请求的长度，单位：字节。 |
| <time> | 整型，表示发送 Ping 请求花费的时间，单位：毫秒。 |

| | |
|-------------|--|
| <ttl> | 整型，Ping 请求的 TTL 值。 |
| <finresult> | 整型，表示 Ping 操作的最终结果。 0 Ping 成功 Others 请参考第三章获取指定错误代码信息 |
| <sent> | 整型，表示所有 Ping 请求发送的字节总长度。 |
| <rcvd> | 整型，表示 Ping 响应中接收到的字节总长度。 |
| <lost> | 整型，表示所有 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，当前不支持 IPv6。

2.1.9. 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）。请参考第三章获取指定错误代码信息。

| AT+QNTTP 通过 NTP 服务器同步本地时间 | |
|--|---|
| 测试命令 AT+QNTTP=? | 响应 +QNTTP: (1-3),"<server>"[,<port>[, (0,1)]] OK |
| 设置命令 AT+QNTTP=<contextID>,<server>[,<p ort>[,<autosettime>]] | 响应 若同步成功，响应： OK +QNTTP: <err>,<time> 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|---------------|------------------------------------|
| <contextID> | 整型，场景 ID，范围：1-3。 |
| <server> | 字符串型，NTP 服务器地址，格式为域名或者点分十进制 IP 地址。 |
| <port> | 整型，表示 NTP 服务器端口号，默认值：123。 |
| <autosettime> | 整型，表示是否自动设置同步本地时间。 0 不设置 |

| | |
|--------|---|
| <time> | 1 设置 |
| | 字符串型，从 NTP 服务器上同步的时间。 |
| | 格式为：“YY/MM/DD, hh:mm:ss±zz”，“zz”的范围是-47~48。 |

备注

1. 目前仅支持<contextID>=1，以及<port>=123。
2. 当<autosettime>设置为 1，RTC 会自动根据同步时间来更新，然后可使用命令 **AT+QCCLK?**来查询更新后的时间。
3. 注网成功后，模块会自动更新 RTC 时间。

2.1.10. AT+QIDNSGIP 通过域名获取 IP 地址

该命令用来转换指定域名为 IP 地址格式。请参考第 3 章获取指定错误代码信息。

| AT+QIDNSGIP 通过域名获取 IP 地址 | |
|---|---|
| 测试命令 AT+QIDNSGIP=? | 响应 +QIDNSGIP: (1-3), "<hostname>" OK |
| 设置命令 AT+QIDNSGIP=<contextID>,<hostname> | 响应 OK +QIURC: "dnsgip",<err>,<IP_count>,<DNS_ttl> [+QIURC: "dnsgip",<hostIPAddr>] [...] 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|--------------|-----------------------------|
| <contextID> | 整型，场景 ID，范围：1-3。 |
| <hostname> | 字符串型，域名。 |
| <IP_count> | 整型，参数<hostname>对应的 IP 地址个数。 |
| <DNS_ttl> | 整型，DNS 的 TTL 值。 |
| <hostIPAddr> | 字符串型，<hostname>的 IP 地址。 |

备注

目前仅支持<contextID>=1

2.1.11. AT+QIDNSCFG 配置 DNS 服务器地址

该命令用来配置主要和次要 DNS 服务器的地址。请参考第 3 章获取指定错误代码信息。

| AT+QIDNSCFG 配置 DNS 服务器地址 | |
|---|--|
| 测试命令 AT+QIDNSCFG=? | 响应 +QIDNSCFG: (1-3),<pridnsaddr>,<secdnsaddr> OK |
| 设置命令 成功配置后，查询主要和次要 DNS 服务器地址 AT+QIDNSCFG=<contextID> | 响应 [+QIDNSCFG: <contextID>,<pridnsaddr>[,<secdnsaddr>]] OK 若有任何有关 ME 功能错误，响应： ERROR |
| 设置命令 配置主要和次要 DNS 服务器地址 AT+QIDNSCFG=<contextID>,<pridnsaddr>[,<secdnsaddr>] | 响应 OK 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|---------------------------|--------------------------|
| <contextID> | 整型，场景 ID，范围：1-3。 |
| <pridnsaddr> | 字符串型，IP 格式的主要 DNS 服务器地址。 |
| <secdnsaddr> | 字符串型，IP 格式的次要 DNS 服务器地址。 |

备注

1. 目前仅支持<contextID>=1。
2. 若网络支持 IPv4，那么只能设置 IPv4 DNS 地址；若网络支持 IPv6，那么只能设置 IPv6 DNS 地址。
3. 只能在返回了 IP 地址 URC（例如：+IP: 10.18.237.42，表示注网成功）后，才能配置 DNS 服务器地址。
4. 配置信息不会保存到 NVRAM，模块重启或者深休眠后需要重新配置。

2.1.12. 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=? | 响应 +QICFG: "dataformat",(0,1),(0,1) +QICFG: "viewmode",(0,1) +QICFG: "showlength",(0,1) +QICFG: "echomode",(0,1) OK |
| 设置命令 设置发送/接收数据格式 AT+QICFG="dataformat" [<send_data_format>,<recv_data_format>] | 响应 +QICFG: "dataformat",<send_data_format>,<recv_data_format> OK 若有任何有关 ME 功能错误，响应： ERROR |
| 设置命令 设置接收数据的输出格式 AT+QICFG="viewmode" [<view_mode>] | 响应 +QICFG: "viewmode",<view_mode> OK 若有任何有关 ME 功能错误，响应： ERROR |
| 设置命令 设置在非透传 Buffer 模式下是否显示可选数据长度参数 ¹⁾ AT+QICFG="showlength" [<show_length_mode>] | 响应 +QICFG: "showlength",<show_length_mode> OK 若有任何有关 ME 功能错误，响应： ERROR |
| 设置命令 设置在数据模式下是否向 UART 回显输入数据 AT+QICFG="echomode" [<echo_mode>] | 响应 +QICFG: "echomode",<echo_mode> OK 若有任何有关 ME 功能错误，响应： ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|---------------------------------|---|
| <send_data_format> | 整型，表示发送数据的格式。 0 文本格式 1 十六进制格式 |
| <recv_data_format> | 整型，表示接收数据的格式。 0 文本格式 1 十六进制格式 |
| <view_mode> | 整型，表示已接收数据的输出格式。 0 已收数据输出格式：data header\r\n\data 1 已收数据输出格式：data header,data |
| <show_length_mode> | 整型，在非透传 Buffer 模式下是否显示可选数据长度参数 ¹⁾ 。 0 非透传 Buffer 模式下不显示 1 非透传 Buffer 模式下显示 |
| <echo_mode> | 整型，是否在数据模式下向 UART 回显输入数据。 0 不向 UART 回显输入数据 1 向 UART 回显输入数据 |

备注

- 1) 可选数据长度参数是：
 - URC “+QIURC: "recv",<connectID>[,<current_recv_length>]” 中的参数<current_recv_length>
 - 命令 AT+QIRD 返回值中的参数<remaining_length>
- 这些配置的参数会立即生效。
- 参数<send_data_format>，<recv_data_format>，<view_mode>和<show_length_mode>的配置会自动保存到 NVRAM。
- 只有在数据模式传输中，参数<echo_mode>才有效，且仅 BC26NxR01A07 以及 BC20NxR01A08 和两者之后的版本才支持该参数。

2.1.13. AT+QIGETERROR 查询上一个错误代码

该命令用来查询参数<err>错误代码，以及上一个 TCP/IP 命令返回<err>错误代码的特定描述。

| AT+QIGETERROR 查询上一个错误代码 | |
|--------------------------------|---|
| 测试命令 AT+QIGETERROR=? | 响应 OK |
| 执行命令 AT+QIGETERROR | 响应： +QIGETERROR: <err>,<errcode_description> OK 若有任何有关 ME 功能错误，响应： |

| | |
|--------|--------|
| | ERROR |
| 最大响应时间 | 300 毫秒 |

参数

| | |
|------------------------------------|---|
| <errcode_description> | 字符串型，表示详细的错误信息描述。请参考第3章获取指定错误代码信息以及对应的描述。 |
|------------------------------------|---|

2.2. URC 说明

TCP/IP AT 命令的 URC 统一以 “<CR><LF>+QIURC: <type>[...]<CR><LF>” 格式上报通知模块，为了简便，本文中 URC 前后的 “<CR><LF>” 均省略。

备注

1. 当模块在 PSM 模式下，不会上报 URC。
2. 当模块在 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. 当模块在连接模式下，会正常上报 URC。

2.2.1. 连接断开 URC

当 TCP Socket 服务被远程关闭或者因为网络异常而断开，模块将上报 URC “+QIURC: "closed",<connectID>”，同时参数<socket_state>（表示 Socket 服务的状态）变为 “Closing” 状态，模块必须执行命令 AT+QICLOSE=<connectID>把<socket_state>改为 “Initial” 状态。

在非透传 Buffer 模式下，模块也可以执行命令 AT+QIRD=< connectID >,<read_length>来读取缓存数据。

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

参数

| | |
|--------------------------|-----------------------|
| <connectID> | 整型，Socket 服务索引，范围：0-4 |
|--------------------------|-----------------------|

2.2.2. 下行数据 URC

在非透传 Buffer 模式或者非透传 Push 模式下，当模块从服务器上接收到数据，模块会上报 URC。

在非透传 Buffer 模式下，URC 的格式为：+QIURC: "recv",<connectID>,<current_rcv_length>]

在非透传 Push 模式下，URC 格式为：+QIURC: "recv",<connectID>,<current_rcv_length><CR><LF><data>

| 下行数据 URC | |
|---|-----------------------|
| +QIURC: "recv",<connectID>,<current_rcv_length>] | 表示非透传 Buffer 模式下有下行数据 |
| +QIURC: "recv",<connectID>,<current_rcv_length><CR><LF><data> | 表示非透传 Push 模式下有下行数据 |

参数

| | |
|----------------------|------------------------|
| <connectID> | 整型，Socket 服务索引，范围：0-4。 |
| <current_rcv_length> | 整型，实际接收到的数据长度。 |
| <data> | 接收到的数据 |

2.2.3. 下行数据缓存已满 URC

在非透传 Buffer 模式下，如果已经没有资源可以分配给下行数据进行缓存，模块会上报如下 URC。

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

参数

| | |
|-------------|------------------------|
| <connectID> | 整型，Socket 服务索引，范围：0-4。 |
|-------------|------------------------|

备注

为非透传 Buffer 模式数据可分配的最大节点数为 20。

3 错误代码

执行 TCP/IP AT 命令后返回的<err>，具体的错误信息可以通过 AT+QIGETERROR 来查询。值得注意的是，AT+QIGETERROR 只返回上一个 TCP/IP AT 命令的错误代码。

表 1：错误代码列表

| <err>错误码 | 错误代码描述 | 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 | 不支持操作 |
| 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 身份被使用 |
| 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 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 | 不允许操作 |
| 573 | APN not configured | 未配置 APN |
| 574 | Port busy | 端口繁忙 |

4 举例

4.1. TCP 客户端在非透传 Buffer 模式下工作

4.1.1. 创建一个 TCP 客户端连接并进入非透传 Buffer 模式

//打开一个 Socket 服务，场景 ID 为 1，Socket 服务索引为 0。

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

OK

+QIOPEN: 0,0 //连接成功，等待 URC 上报后再进行其他操作，模块超时时间为 60 秒。

AT+QISTATE=1,0 //查询<connectID>=0 的连接状态。

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

OK

4.1.2. 在非透传 Buffer 模式下发送数据

AT+QISEND=0,10,1234567890 //发送数据，数据长度是 10 个字节。

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，且长度是 10 个字节。

>

1234567890

OK

SEND OK

```

AT+QISENDEX=0,5,3031323334      //发送十六进制数据。
OK

SEND OK

```

4.1.3. 在非透传 Buffer 模式下从远程服务器上接收数据

```

+QIURC: "recv",0                //索引为 0 的 Socket 服务接收到数据。

AT+QIRD=0,512                    //读取数据，数据长度为 512 字节。
+QIRD: 10
1234567890

OK

AT+QIRD=0,512                    //读取数据，数据长度为 512 字节。
+QIRD: 0                          //缓存中无数据。

OK

AT+QICFG="showlength",1          //使能在非透传 Buffer 模式下显示可选参数
                                  <current_recv_length> 和<remaining_length>
OK

+QIURC: "recv",0,12              //索引为 0 的 Socket 服务接收到了数据，且数据长度为 12 字节。

AT+QIRD=0,10                     //读取数据，读取长度 10 字节。
+QIRD: 10,2                       //已读取 10 字节长度的数据，剩余还有 2 字节未读取。
1234567890

OK

+QIURC: "recv",0,"buff full"     //索引为 0 的 Socket 服务反应缓存已满，模块需要使用命令
                                  AT+QIRD 来读取缓存数据。

AT+QICFG="viewmode",1            //已收到数据的输出格式：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      //断开索引号为 0 的 Socket 服务连接
OK

CLOSE OK
```

4.2. TCP 客户端在非透传 Push 模式下工作

4.2.1. 创建一个 TCP 客户端连接并进入非透传 Push 模式

```
//打开一个 Socket 服务，场景 ID 为 1，Socket 服务索引为 0。
AT+QIOPEN=1,0,"TCP","220.180.239.212",8062,0,1
OK

+QIOPEN: 0,0      //连接成功，等待 URC 上报后再进行其他操作，模块超时时间为 60 秒。

AT+QISTATE=1,0    //查询<connectID>=0 的连接状态。
+QISTATE: 0,"TCP","220.180.239.212",8062,0,2,1,1

OK
```

4.2.2. 在非透传 Push 模式下发送数据

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

SEND OK

AT+QISENDEX=0,5,3132333435                    //发送十六进制数据。
OK

SEND OK

```

4.2.3. 在非透传 Push 模式下接收来自远程服务器的数据

```

+QIURC: "recv",0,5                            //收到来自远程服务器的数据
12345

AT+QICFG="viewmode",1                         //已收到数据的输出格式：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    //断开索引为 0 的 Socket 服务连接
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 NTP 服务器 ntp5.aliyun.com 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. 获取上一个错误代码

```
//打开任意一个 Socket 服务。
```

```
AT+QIOPEN=1,"UDP","220.180.239.212",8063,0,1
ERROR
```

```
AT+QIGETERROR
```

```
+QIGETERROR: 552,invalid parameters
```

```
OK
```

5 附录 A

表 2：术语缩写

| 缩写 | 描述 |
|-------|-----------------------------------|
| eDRX | extended Discontinuous Reception |
| DNS | Domain Name System |
| DRX | Discontinuous Reception |
| IP | Internet Protocol |
| IPv4 | Internet Protocol version 4 |
| IPv6 | Internet Protocol version 6 |
| ME | Mobile Equipment |
| NTP | Network Time Protocol |
| NVRAM | Non-Volatile Random Access Memory |
| PPP | Point to Point Protocol |
| PSM | Power Saving Mode |
| TCP | Transmission Control Protocol |
| UDP | User Datagram Protocol |
| URC | Unsolicited Result Code |
| UTC | Universal Time Coordinated |