

# EC200x&EG912Y&EC600S 系列

## HTTP(S)应用指导

**LTE Standard 模块系列**

版本：1.0

日期：2020-11-30

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。因未能遵守有关操作或设计规范而造成的损害，上海移远通信技术股份有限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。

## 免责声明

上海移远通信技术股份有限公司尽力确保开发中功能的完整性、准确性、及时性或效用，但不排除上述功能错误或遗漏的可能。除非其他有效协议另有规定，否则上海移远通信技术股份有限公司对开发中功能的使用不做任何暗示或明示的保证。在适用法律允许的最大范围内，上海移远通信技术股份有限公司不对任何因使用开发中功能而遭受的损失或损害承担责任，无论此类损失或损害是否可以预见。

## 保密义务

除非上海移远通信技术股份有限公司特别授权，否则我司所提供文档和信息的接收方须对接收的文档和信息保密，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。未经上海移远通信技术股份有限公司书面同意，不得获取、使用或向第三方泄露我司所提供的文档和信息。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，上海移远通信技术股份有限公司有权追究法律责任。

## 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 |
|-----|------------|-----|------|
| -   | 2020-10-30 | 刘康康 | 文档创建 |
| 1.0 | 2020-11-30 | 刘康康 | 受控版本 |

## 目录

|   |           |
|---|-----------|
| 文档历史.....   | 2         |
| 目录.....   | 3         |
| 表格索引.....   | 5         |
| <b>1 引言.....</b>  | <b>6</b>  |
| 1.1. 适用模块.....  | 6         |
| 1.2. HTTP(S)命令使用流程.....                                     | 6         |
| 1.3. HTTP(S)请求头信息说明.....                                    | 7         |
| 1.3.1. 自定义 HTTP(S)请求头信息.....                                | 7         |
| 1.3.2. 输出 HTTP(S)响应头信息.....                                 | 8         |
| 1.4. 数据模式说明.....  | 8         |
| <b>2 HTTP(S) AT 命令详解.....</b>                               | <b>9</b>  |
| 2.1. AT 命令语句.....   | 9         |
| 2.1.1. 定义.....  | 9         |
| 2.1.2. AT 命令语句.....   | 9         |
| 2.2. AT 命令详解.....   | 9         |
| 2.2.1. AT+QHTTPCFG 配置 HTTP(S)服务器参数.....                     | 9         |
| 2.2.2. AT+QHTTPURL 设置 HTTP(S)服务器 URL.....                   | 13        |
| 2.2.3. AT+QHTTPGET 发送 GET 请求到 HTTP(S)服务器.....               | 14        |
| 2.2.4. AT+QHTTPGETEX 发送范围 GET 请求到 HTTP(S)服务器.....           | 15        |
| 2.2.5. AT+QHTTPPOST 通过 UART/USB 发送 POST 请求到 HTTP(S)服务器..... | 16        |
| 2.2.6. AT+QHTTPPOSTFILE 通过文件发送 POST 请求到 HTTP(S)服务器.....     | 18        |
| 2.2.7. AT+QHTTPREAD 通过 UART/USB 读取 HTTP(S)服务器响应信息.....      | 19        |
| 2.2.8. AT+QHTTPREADFILE 通过文件读取 HTTP(S)服务器响应信息.....          | 20        |
| 2.2.9. AT+QHTTPSTOP 取消 HTTP(S)请求.....                       | 21        |
| <b>3 举例.....</b>  | <b>22</b> |
| 3.1. 访问 HTTP 服务器.....                                       | 22        |
| 3.1.1. 发送 HTTP GET 请求并读取响应信息.....                           | 22        |
| 3.1.2. 发送 HTTP POST 请求并读取响应信息.....                          | 23        |
| 3.1.2.1. 通过 UART/USB 获取 POST 请求体.....                       | 23        |
| 3.1.2.2. 从文件系统获取 POST 请求体.....                              | 24        |
| 3.2. 访问 HTTPS 服务器.....                                      | 25        |
| 3.2.1. 发送 HTTPS GET 请求并读取响应信息.....                          | 25        |
| 3.2.2. 发送 HTTPS POST 请求并读取响应信息.....                         | 27        |
| 3.2.2.1. 从 UART/USB 获取 POST 请求体.....                        | 27        |
| 3.2.2.2. 从文件系统获取 POST 请求体.....                              | 28        |
| <b>4 常见问题处理.....</b>  | <b>30</b> |
| 4.1. HTTP(S) AT 命令执行失败.....                                 | 30        |
| 4.2. PDP 激活失败.....  | 30        |
| 4.3. DNS 解析失败.....  | 30        |

---

|      |                      |    |
|------|----------------------|----|
| 4.4. | 数据模式进入失败.....        | 31 |
| 4.5. | GET/POST 请求发送失败..... | 31 |
| 4.6. | 响应信息读取失败.....        | 31 |
| 5    | 错误代码.....            | 32 |
| 6    | HTTP(S)响应错误代码 .....  | 34 |
| 7    | 附录 参考文档和术语缩写.....    | 35 |

## 表格索引

|                          |    |
|--------------------------|----|
| 表 1: 适用模块 .....          | 6  |
| 表 2: AT 命令及响应类型 .....    | 9  |
| 表 3: 错误代码列表 .....        | 32 |
| 表 4: HTTP(S)响应代码列表 ..... | 34 |
| 表 5: 参考文档 .....          | 35 |
| 表 6: 术语缩写 .....          | 35 |

# 1 引言

EC200x 系列、EG912Y 系列和 EC600S-CN 模块为 HTTP(S)服务器提供 HTTP(S)应用程序。

HTTP（HyperText Transfer Protocol，超文本传输协议）是一种用于分布式、协作式和超媒体信息系统的應用层协议。

HTTPS（HyperText Transfer Protocol Secure，超文本传输安全协议）是一种通过计算机网络进行安全通信的传输协议。HTTPS 经由 HTTP 进行通信，但利用 SSL/TLS 来加密数据包。HTTPS 开发的主要目的，是提供对网站服务器的身份认证，保护交换资料的隐私与完整性。

本文档主要介绍了与 HTTP(S)相关的 AT 命令。

## 1.1. 适用模块

表 1：适用模块

| 模块系列      | 模块        |
|-----------|-----------|
| EC200x 系列 | EC200T 系列 |
|           | EC200S-CN |
| EG912Y 系列 | EG912Y 系列 |
| EC600S    | EC600S-CN |

## 1.2. HTTP(S)命令使用流程

通过 EC200x 系列、EG912Y 系列和 EC600S-CN 模块 TCP/IP AT 命令，可配置 PDP 上下文，激活/去激活上下文以及查询 PDP 上下文状态。通过 EC200x 系列、EG912Y 系列和 EC600S-CN 模块的 HTTP(S) AT 命令，可以发送 HTTP(S) GET/POST 请求到 HTTP(S)服务器，并读取来自 HTTP(S)服务器的响应结果。大致流程如下：

第一步，使用 **AT+QICSGP** 配置 PDP 上下文的<APN>、<username>、<password>和其他参数，详细信息请参考文档[1]。

第二步，通过 **AT+QIACT** 激活 PDP 上下文后，可使用 **AT+QIACT?** 查询已分配的 IP 地址，参见文档[1]。

第三步，通过 **AT+QHTTPCFG** 配置 PDP 和 SSL 上下文 ID。

第四步，通过 **AT+QSSLCFG** 配置 SSL 上下文参数，参见文档[2]。

第五步，通过 **AT+QHTTTPURL** 设置 HTTP(S) URL。

第六步，发送 HTTP(S)请求。**AT+QHTTTPGET** 用于发送 HTTP(S) GET 请求，**AT+QHTTTPGETEX** 用于发送 HTTP(S)范围 GET 请求。**AT+QHTTTPPOST** 或 **AT+QHTTTPPOSTFILE** 用于发送 HTTP(S) POST 请求。

第七步，通过 **AT+QHTTTPREAD** 或 **AT+QHTTTPREADFILE** 读取 HTTP(S)响应信息。

第八步，通过 **AT+QIDEACT** 去激活 PDP 上下文，参见文档[1]。

## 1.3. HTTP(S)请求头信息说明

### 1.3.1. 自定义 HTTP(S)请求头信息

模块自动填补 HTTP(S)请求头信息，用户可通过 **AT+QHTTPCFG** 将<request\_header>配置为 1 自定义 HTTP(S)请求头信息，但需遵循以下标准：

- 遵循 HTTP(S)请求头信息语句规范。
- HTTP(S)请求行中的 URI 值和 **Host:**请求头信息必须与 **AT+QHTTTPURL** 配置的 URL 一致。
- HTTP(S)请求头信息必须以<CR><LF>结尾。

以下为标准 HTTP(S) POST 请求头信息示例：

```
POST /processorder.php HTTP/1.1<CR><LF>
Host: 220.180.239.212:8011<CR><LF>
Accept: /*<CR><LF>
User-Agent: QUECTEL_MODULE<CR><LF>
Connection: Keep-Alive<CR><LF>
Content-Type: application/x-www-form-urlencoded<CR><LF>
Content-Length: 48<CR><LF>
<CR><LF>
Message=1111&Appleqty=2222&Orangeqty=3333&find=1
```

### 1.3.2. 输出 HTTP(S)响应头信息

模块不自动输出 HTTP(S)响应头信息，可通过 **AT+QHTTPCFG** 将 `<response_header>` 配置为 1 获取 HTTP(S)响应头信息，然后执行 **AT+QHTTTPREAD** 或 **AT+QHTTTPREADFILE**，HTTP(S)响应头信息将以 HTTP(S)响应体形式输出。

## 1.4. 数据模式说明

EC200x 系列、EG912Y 系列和 EC600S-CN 模块的 COM 口有两种工作模式，一是 AT 命令模式，另一种是数据模式。AT 命令模式下，通过 COM 口输入的数据被认为是 AT 命令；数据模式下，则被认为是数据。

用户可以通过 **+++** 或者 **DTR**（需先设置 **AT&D1**）两种方式退出数据模式，为了防止 **+++** 被当成数据发送，实际操作时必须遵循以下标准：

- **+++** 输入前 1 秒内不能输入其它任何数据。
- 必须在 1 秒内输入 **+++**，并且不能输入其它任何数据。
- **+++** 输入后 1 秒内不能输入其它任何数据。

通过执行 **AT+QHTTTPURL**、**AT+QHTTTPPOST** 和 **AT+QHTTTPREAD** 可使 COM 口进入数据模式，但在这些命令响应之前，若通过 **+++** 或拉高 **DTR** 退出数据模式，会中断这些命令的执行。在这种情况下，无法通过执行 **ATO** 命令使 COM 口重新进入数据模式。

## 2 HTTP(S) AT 命令详解

### 2.1. AT 命令语句

#### 2.1.1. 定义

- **<CR>** 回车符。
- **<LF>** 换行符。
- **<...>** 参数名称。实际命令中不包含尖括号。
- **[...]** 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，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。             |

### 2.2. AT 命令详解

#### 2.2.1. AT+QHTTPCFG 配置 HTTP(S)服务器参数

该命令用于配置 HTTP(S)服务器参数，包括配置 PDP 上下文 ID，自定义 HTTP(S)请求头信息，输出 HTTP(S)响应头信息和查询 SSL 设置。若执行设置命令时只保留一个参数，表示查询当前配置。

**AT+QHTTPCFG 配置 HTTP(S)服务器参数**

|  |  |
|--|--|
| <p>测试命令<br/>AT+QHTTPCFG=?</p>  | <p>响应<br/>+QHTTPCFG: "contextid", (支持的&lt;contextID&gt;范围)<br/>+QHTTPCFG: "requestheader", (支持的&lt;request_header&gt;列表)<br/>+QHTTPCFG: "responseheader", (支持的&lt;response_header&gt;列表)<br/>+QHTTPCFG: "sslctxid", (支持的&lt;sslctxID&gt;范围)<br/>+QHTTPCFG: "contenttype", (支持的&lt;content_type&gt;范围)<br/>+QHTTPCFG: "rspout/auto", (支持的&lt;auto_outrsp&gt;列表)<br/>+QHTTPCFG: "closed/ind", (支持的&lt;closedind&gt;列表)<br/><br/>OK</p> |
| <p>设置命令<br/>AT+QHTTPCFG="contextid" [, &lt;contextID&gt;]</p>            | <p>响应<br/>若省略可选参数，查询当前配置：<br/>+QHTTPCFG: "contextid", &lt;contextID&gt;<br/><br/>OK<br/><br/>若指定可选参数：<br/>OK<br/>或者<br/>+CME ERROR: &lt;err&gt;</p>  |
| <p>设置命令<br/>AT+QHTTPCFG="requestheader" [, &lt;request_header&gt;]</p>   | <p>响应<br/>若省略可选参数，查询当前配置：<br/>+QHTTPCFG: "requestheader", &lt;request_header&gt;<br/><br/>OK<br/><br/>若指定可选参数：<br/>OK<br/>或者<br/>+CME ERROR: &lt;err&gt;</p>   |
| <p>设置命令<br/>AT+QHTTPCFG="responseheader" [, &lt;response_header&gt;]</p> | <p>响应<br/>若省略可选参数，查询当前配置：<br/>+QHTTPCFG: "responseheader", &lt;response_header&gt;<br/><br/>OK<br/><br/>若指定可选参数：<br/>OK<br/>或者</p>   |

|   |   |
|---|---|
|   | <p><b>+CME ERROR: &lt;err&gt;</b></p>   |
| <p>设置命令<br/><b>AT+QHTTPCFG="sslctxid"[,&lt;sslctxid&gt;]</b></p>        | <p>响应<br/>若省略可选参数，查询当前配置：<br/><b>+QHTTPCFG: "sslctxid",&lt;sslctxid&gt;</b></p> <p><b>OK</b></p> <p>若指定可选参数：<br/><b>OK</b><br/>或者<br/><b>+CME ERROR: &lt;err&gt;</b></p>        |
| <p>设置命令<br/><b>AT+QHTTPCFG="contenttype"[,&lt;content_type&gt;]</b></p> | <p>响应<br/>若省略可选参数，查询当前配置：<br/><b>+QHTTPCFG: "contenttype",&lt;content_type&gt;</b></p> <p><b>OK</b></p> <p>若指定可选参数：<br/><b>OK</b><br/>或者<br/><b>+CME ERROR: &lt;err&gt;</b></p> |
| <p>设置命令<br/><b>AT+QHTTPCFG="rspout/auto"[,&lt;auto_outrsp&gt;]</b></p>  | <p>响应<br/>若省略可选参数，查询当前配置：<br/><b>+QHTTPCFG: "rspout/auto",&lt;auto_outrsp&gt;</b></p> <p><b>OK</b></p> <p>若指定可选参数：<br/><b>OK</b><br/>或者<br/><b>+CME ERROR: &lt;err&gt;</b></p>  |
| <p>设置命令<br/><b>AT+QHTTPCFG="closed/ind"[,&lt;closedind&gt;]</b></p>     | <p>响应<br/>若省略可选参数，查询当前配置：<br/><b>+QHTTPCFG: "closed/ind",&lt;closedind&gt;</b></p> <p><b>OK</b></p> <p>若指定可选参数：<br/><b>OK</b><br/>或者<br/><b>+CME ERROR: &lt;err&gt;</b></p>     |
| <p>查询命令</p>   | <p>响应</p>   |

|              |   |
|--------------|---|
| AT+QHTTPCFG? | <p>+QHTTPCFG: "contextid",&lt;contextID&gt;<br/>                 +QHTTPCFG: "requestheader",&lt;request_header&gt;<br/>                 +QHTTPCFG: "responseheader",&lt;response_header&gt;<br/>                 +QHTTPCFG: "sslctxid",&lt;sslctxID&gt;<br/>                 +QHTTPCFG: "contenttype",&lt;content_type&gt;<br/>                 +QHTTPCFG: "rspout/auto",&lt;auto_outrsp&gt;<br/>                 +QHTTPCFG: "closed/ind",&lt;closedind&gt;</p> <p>OK</p> |
| 特性说明         | 该命令立即生效，参数配置不保存   |

### 参数

|                   |  |
|-------------------|--|
| <contextID>       | 整型。PDP 上下文 ID。范围：1~15。默认值：1。   |
| <request_header>  | <p>整型。禁用或启用自定义 HTTP(S)请求头信息。</p> <p><u>0</u> 禁用<br/>1 启用</p>   |
| <response_header> | <p>整型。禁用或启用输出 HTTP(S)响应头信息。</p> <p><u>0</u> 禁用<br/>1 启用</p>  |
| <sslctxID>        | 整型。HTTP(S)的 SSL 上下文 ID。范围：0~5。默认值：1。通过 <b>AT+QSSLCFG</b> 配置 SSL 参数，参考 <a href="#">文档[2]</a> 。  |
| <content_type>    | <p>整型。HTTP(S)体的数据类型。</p> <p><u>0</u> application/x-www-form-urlencoded<br/>1 text/plain<br/>2 application/octet-stream<br/>3 multipart/form-data</p> |
| <auto_outrsp>     | <p>整型。禁用或启用自动输出 HTTP(S)响应头信息。若启用自动输出 HTTP(S)响应头信息，<b>AT+QHTTTPREAD</b> 和 <b>AT+QHTTTPREADFILE</b> 会执行失败。</p> <p><u>0</u> 禁用<br/>1 启用</p>             |
| <closedind>       | <p>整型。禁用或启用上报 HTTP(S)会话关闭标识。</p> <p><u>0</u> 禁用<br/>1 启用</p>   |
| <err>             | 整型。错误代码。参考 <a href="#">第5章</a> 。   |

### 2.2.2. AT+QHTTPURL 设置 HTTP(S)服务器 URL

HTTP(S)服务器的 URL 必须以 http://或 https://开头，表示访问 HTTP 或 HTTPS 服务器。

| AT+QHTTPURL 设置 HTTP(S)服务器 URL                                   |  |
|---|--|
| 测试命令<br><b>AT+QHTTPURL=?</b>                                    | <p>响应</p> <p><b>+QHTTPURL:</b> (支持的 &lt;URL_length&gt; 范围),(支持的 &lt;timeout&gt;范围)</p> <p><b>OK</b></p>  |
| 设置命令<br><b>AT+QHTTPURL=&lt;URL_length&gt;[,&lt;timeout&gt;]</b> | <p>响应</p> <p>若参数格式正确，且不发送 HTTP(S) GET/POST 请求：<br/><b>CONNECT</b></p> <p>TA 切换到透传模式，即可输入 URL。当输入数据的总大小达到&lt;URL_length&gt;，TA 将切换回命令模式并上报以下结果：<br/><b>OK</b></p> <p>若输入时间达到&lt;timeout&gt;，但接收的 URL 长度小于&lt;URL_length&gt;，TA 将切换回命令模式并上报以下结果：<br/><b>+CME ERROR: &lt;err&gt;</b></p> <p>若参数格式不正确或发生其他错误：<br/><b>+CME ERROR: &lt;err&gt;</b></p> |
| 查询命令<br><b>AT+QHTTPURL?</b>                                     | <p>响应</p> <p><b>[+QHTTPURL: &lt;URL&gt;]</b></p> <p><b>OK</b></p>  |
| 最大响应时间  | 取决于<timeout>   |
| 特性说明  | 该命令立即生效，参数配置不保存  |

#### 参数

|              |  |
|--------------|--|
| <URL_length> | 整型。URL 长度。范围：1~2048。单位：字节。             |
| <timeout>    | 整型。URL 的最大输入时间。范围：1~65535。默认值：60。单位：秒。 |
| <err>        | 整型。错误代码。参考第 5 章。                       |

### 2.2.3. AT+QHTTPGET 发送 GET 请求到 HTTP(S)服务器

根据在 **AT+QHTTPCFG="requestheader",<request\_header>** 命令中配置的参数 <request\_header>, **AT+QHTTPGET** 设置命令有两种形式。若 <request\_header> 为 1, 发送 **AT+QHTTPGET** 后, 若 125 秒内输出 **CONNECT**, 则表示 HTTP(S)服务器连接成功; 若 125 秒内未上报 **CONNECT**, 将输出 **+CME ERROR: <err>**。

发送 **AT+QHTTPGET** 设置命令后, 上报 **OK** 后, 需要等待一段时间 (参考最大响应时间) 才会输出 **URC +QHTTPGET: <err>,<httprspcode>,<content\_length>**。

**+QHTTPGET: <err>,<httprspcode>,<content\_length>** 中, 仅当 <err> 为 0 时, 才上报 <httprspcode>。若 HTTP(S)响应头信息包括 **CONTENT-LENGTH**, 将上报 <content\_length> 信息。

#### AT+QHTTPGET 发送 GET 请求到 HTTP(S)服务器

|   |  |
|---|--|
| <p>测试命令<br/><b>AT+QHTTPGET=?</b></p>  | <p>响应<br/><b>+QHTTPGET: (支持的 &lt;rsptime&gt; 范围),(支持的 &lt;data_length&gt;范围),(支持的 &lt;input_time&gt;范围)</b></p> <p><b>OK</b></p>   |
| <p>设置命令<br/>若 &lt;request_header&gt; 为 0 (禁用自定义 HTTP(S)请求头信息)<br/><b>AT+QHTTPGET[=&lt;rsptime&gt;]</b></p>                                      | <p>响应<br/>若参数格式正确且无其他错误发生:<br/><b>OK</b></p> <p>模块收到来自 HTTP(S)服务器的响应后, 将上报以下 URC:<br/><b>+QHTTPGET: &lt;err&gt;,&lt;httprspcode&gt;,&lt;content_length&gt;</b></p> <p>若参数格式不正确或有其他错误发生:<br/><b>+CME ERROR: &lt;err&gt;</b></p>   |
| <p>设置命令<br/>若 &lt;request_header&gt; 为 1 (启用自定义 HTTP(S)请求头信息)<br/><b>AT+QHTTPGET=&lt;rsptime&gt;,&lt;data_length&gt;,&lt;input_time&gt;</b></p> | <p>响应<br/>若 HTTP(S)服务器连接成功:<br/><b>CONNECT</b></p> <p>TA 切换到透传模式, 即可输入 HTTP(S) GET 请求头信息。当输入数据的总大小达到 &lt;data_length&gt;, TA 将切换回命令模式并上报以下结果:<br/><b>OK</b></p> <p>模块收到来自 HTTP(S)服务器的响应后, 将上报以下 URC:<br/><b>+QHTTPGET: &lt;err&gt;,&lt;httprspcode&gt;,&lt;content_length&gt;</b></p> <p>若输入时间达到 &lt;input_time&gt;, 但收到的数据长度小于 &lt;data_length&gt;, TA 将切换回命令模式并上报以下结果:<br/><b>+QHTTPGET: &lt;err&gt;</b></p> |

|        |   |
|--------|---|
|        | 若参数格式不正确或有其他错误发生：<br><b>+CME ERROR: &lt;err&gt;</b> |
| 最大响应时间 | 取决于<rsptime>  |
| 特性说明   | 该命令立即生效，参数配置不保存                                     |

## 参数

|                  |   |
|------------------|---|
| <rsptime>        | 整型。范围：1~65535。默认值：60。单位：秒。上报 <b>OK</b> 后，此参数可用来配置 HTTP(S) GET 响应+QHTTPGET: <err>,<httprcode>[,<content_length>]的最大响应时间。 |
| <data_length>    | 整型。HTTP(S)请求信息的长度，包括 HTTP(S)请求头信息和 HTTP(S)请求体。范围：1~2048。单位：字节。  |
| <input_time>     | 整型。HTTP(S)请求信息的最大输入时间。范围：1~65535。默认值：60。单位：秒。   |
| <httprcode>      | http 回复码。详细信息请参考第6章。  |
| <request_header> | 整型。禁用或启用自定义 HTTP(S)请求头信息。<br>0 禁用<br>1 启用   |
| <content_length> | 整型。HTTP(S)响应体长度。单位：字节。  |
| <err>            | 整型。错误代码。参考第5章。  |

### 2.2.4. AT+QHTTPGETEX 发送范围 GET 请求到 HTTP(S)服务器

与读取文件类似，MCU 可以通过 **AT+QHTTPGETEX** 从 HTTP(S)服务器获取指定位置和指定长度的数据，仅在 **AT+QHTTPCFG="requestheader",0** 时，方可执行该命令。之后，HTTP(S)服务器将始终以 **206** 代码响应范围 GET 请求。

#### AT+QHTTPGETEX 发送范围 GET 请求到 HTTP(S)服务器

|  |  |
|--|--|
| 测试命令<br><b>AT+QHTTPGETEX=?</b>   | 响应<br><b>+QHTTPGETEX: (支持的&lt;rsptime&gt;范围),&lt;start_position&gt;,&lt;read_len&gt;</b><br><br><b>OK</b>  |
| 设置命令<br><b>AT+QHTTPGETEX=&lt;rsptime&gt;,&lt;start_position&gt;,&lt;read_len&gt;</b> | 响应<br>若参数格式正确且无其他错误发生：<br><b>OK</b><br><br>模块收到来自 HTTP(S)服务器响应后，将上报以下 URC：<br><b>+QHTTPGET: &lt;err&gt;,&lt;httprcode&gt;[,&lt;content_length&gt;]</b> |

|        |   |
|--------|---|
|        | 若参数格式不正确或有其他错误发生：<br><b>+CME ERROR: &lt;err&gt;</b> |
| 最大响应时间 | 取决于<rsptime>  |
| 特性说明   | 该命令立即生效，参数配置不保存                                     |

### 参数

|                  |  |
|------------------|--|
| <rsptime>        | 整型。范围：1~65535。默认值：60。单位：秒。上报 OK 后，此参数可用来配置 HTTP(S) GET 响应+QHTTPGET: <err>,<httprspcode>[,<content_length>]的最大响应时间。 |
| <start_postion>  | 整型。HTTP(S)客户端需要 GET 数据的初始位置。   |
| <read_len>       | 整型。HTTP(S)客户端需要 GET 数据的长度。   |
| <httprspcode>    | http 回复码。详细信息请参考第 6 章。   |
| <content_length> | 整型。HTTP(S)响应体长度。单位：字节。   |
| <err>            | 整型。错误代码。参考第 5 章。   |

### 2.2.5. AT+QHTTPPOST 通过 UART/USB 发送 POST 请求到 HTTP(S)服务器

该命令用于发送 HTTP(S) POST 请求。根据在 AT+QHTTPCFG="requestheader"[,<request\_header>]命令中配置的<request\_header>，AT+QHTTPPOST 设置命令有两种形式。若<request\_header>为 0，则通过 UART/USB 口输入 POST 请求体；若<request\_header>为 1，则通过 UART/USB 口输入 POST 请求头信息和 POST 请求体。

发送 AT+QHTTPPOST 后，若 125 秒内返回 CONNECT，则表示 HTTP(S)服务器连接成功；若 125 秒内未返回 CONNECT，将输出+CME ERROR: <err>。

上报 OK 后，需要等待一段时间（参考最大响应时间）才会输出 URC +QHTTPPOST: <err>,<httprspcode>[,<content\_length>]。

#### AT+QHTTPPOST 通过 UART/USB 发送 POST 请求到 HTTP(S)服务器

|   |  |
|---|--|
| 测试命令<br>AT+QHTTPPOST=?  | 响应<br><b>+QHTTPPOST: (支持的 &lt;data_length&gt; 范围),(支持的 &lt;input_time&gt;范围),(支持的&lt;rsptime&gt;范围)</b><br><br><b>OK</b> |
| 设置命令<br>若<request_header>为 0（禁用自定义 HTTP(S)请求头信息）<br>AT+QHTTPPOST=<data_length>[,<input_time>,<rsptime>] | 响应<br>若参数格式正确，HTTP(S)服务器连接成功且 HTTP(S)请求头信息发送完成：<br><b>CONNECT</b><br><br>TA 切换到透传模式，即可输入 HTTP(S) POST 请求体。当输             |

|  |  |
|--|--|
|  | <p>入数据总大小达到&lt;data_length&gt;, TA 将切换回命令模式并上报以下结果:<br/> <b>OK</b></p> <p>模块收到来自 HTTP(S)服务器响应后, 将上报以下 URC:<br/> <b>+QHTTPPOST: &lt;err&gt;,&lt;httprspcode&gt;[,&lt;content_length&gt;]</b></p> <p>若输入时间达到 &lt;input_time&gt;, 但收到的数据长度小于 &lt;data_length&gt;, TA 将切换回命令模式并上报以下结果:<br/> <b>+QHTTPPOST: &lt;err&gt;</b></p> <p>若参数格式不正确或有其他错误发生:<br/> <b>+CME ERROR: &lt;err&gt;</b></p>  |
| <p>设置命令<br/>         若&lt;request_header&gt;为 1 (启用自定义 HTTP(S)请求头信息)<br/> <b>AT+QHTTPPOST=&lt;data_length&gt;[,&lt;input_time&gt;,&lt;rsptime&gt;]</b></p> | <p>响应<br/>         若参数格式正确且 HTTP(S)服务器连接成功:<br/> <b>CONNECT</b></p> <p>TA 切换到透传模式, 即可输入 HTTP(S) POST 请求体和 HTTP(S) POST 请求头信息。当输入数据总大小达到 &lt;data_length&gt;, TA 将切换回命令模式并上报以下结果:<br/> <b>OK</b></p> <p>模块收到来自 HTTP(S)服务器响应后, 将上报以下 URC:<br/> <b>+QHTTPPOST: &lt;err&gt;,&lt;httprspcode&gt;[,&lt;content_length&gt;]</b></p> <p>若输入时间达到 &lt;input_time&gt;, 但收到的数据长度小于 &lt;data_length&gt;, TA 将切换回命令模式并上报以下结果:<br/> <b>+QHTTPPOST: &lt;err&gt;</b></p> <p>若参数格式不正确或有其他错误发生:<br/> <b>+CME ERROR: &lt;err&gt;</b></p> |
| <p>最大响应时间</p>  | <p>取决于网络和&lt;rsptime&gt;</p>   |
| <p>特性说明</p>  | <p>该命令立即生效, 参数配置不保存</p>  |

### 参数

|                            |  |
|----------------------------|--|
| <p>&lt;data_length&gt;</p> | <p>整型。若&lt;request_header&gt;为0, 表示 POST 请求体长度; 若&lt;request_header&gt;为 1, 表示 HTTP(S)请求信息的长度, 包括 HTTP(S) POST 请求头信息和 HTTP(S) POST 请求体。范围: 1~1024000。单位: 字节。</p> |
| <p>&lt;input_time&gt;</p>  | <p>整型。POST 请求体或 HTTP(S)请求信息的最大输入时间。范围: 1~65535。默认值: 60。单位: 秒。</p>  |
| <p>&lt;rsptime&gt;</p>     | <p>整型。范围: 1~65535。默认值: 60。单位: 秒。上报 OK 后, 此参数可用来</p>  |

|                  |  |
|------------------|--|
|                  | 配置 HTTP(S) POST 响应+QHTTPPOST: <err>,<httprcode>[,<content_length>]的最大输出时间。 |
| <httprcode>      | http 回复码。详细信息请参考第 6 章。   |
| <request_header> | 整型。禁用或启用自定义 HTTP(S)请求头信息。<br>0 禁用<br>1 启用                                  |
| <content_length> | 整型。HTTP(S)响应体长度。单位：字节。   |
| <err>            | 整型。错误代码。参考第 5 章。   |

### 2.2.6. AT+QHTTPPOSTFILE 通过文件发送 POST 请求到 HTTP(S)服务器

该命令可用来通过文件发送 POST 请求到 HTTP(S)服务器。根据在 AT+QHTTPCFG="requestheader" [,<request\_header>]命令中配置的<request\_header>, AT+QHTTPPOSTFILE 操作的文件有两种形式。若<request\_header>为 0, 文件系统的文件将为 POST 请求体; 若<request\_header>为 1, 文件系统的文件将为 POST 请求头信息和 POST 请求体。

执行 AT+QHTTPPOSTFILE 后, 模块会上报 URC +QHTTPPOSTFILE: <err>,<httprcode>[,<content\_length>]表示命令执行结果。仅当<err>为 0 时, 才上报<httprcode>。

上报 OK 后, 需要等待一段时间(参考最大响应时间)才会输出 URC +QHTTPPOSTFILE: <err>,<httprcode>[,<content\_length>]信息。

| AT+QHTTPPOSTFILE 通过文件发送 POST 请求到 HTTP(S)服务器   |  |
|---|--|
| 测试命令<br>AT+QHTTPPOSTFILE=?  | 响应<br>+QHTTPPOSTFILE: <file_name>,(支持的<rsptime>范围),(支持的<post_mode>范围)<br>OK  |
| 设置命令<br>若<request_header>为 1, 指定的文件必须包含 HTTP(S)请求头信息。<br>AT+QHTTPPOSTFILE=<file_name>[,<rsptime>,<post_mode>] | 响应<br>若参数格式正确且 HTTP(S)服务器连接成功:<br>OK<br><br>当模块收到来自 HTTP(S)服务器响应后:<br>+QHTTPPOSTFILE: <err>,<httprcode>[,<content_length>]<br><br>若参数格式不正确或有其他错误发生:<br>+CME ERROR: <err> |
| 最大响应时间  | 取决于<rsptime>   |
| 特性说明  | 该命令立即生效, 参数配置不保存   |

参数

|                  |   |
|------------------|---|
| <file_name>      | 字符串类型。文件名称。最大长度为 80 字节。   |
| <rsptime>        | 整型。范围：1~65535。默认值：60。单位：秒。上报 OK 后，此参数可用来配置 HTTP(S) POST 响应+QHTTPPOSTFILE: <err>,<httprcode>[,<content_length>]的最大输出时间。          |
| <httprcode>      | http 回复码。详细信息请参考第 6 章。  |
| <request_header> | 整型。禁用或启用自定义 HTTP(S)请求头信息。<br>0 禁用<br>1 启用   |
| <content_length> | 整型。HTTP(S)响应体长度。单位：字节。  |
| <post_mode>      | 整型。HTTP(S)发送文件的模式。<br>0 直接发送文件内容<br>1 记录保存文件，暂不发送，等待和<post_mode>=2 时配置的文件一起发送<br>2 发送文件，连同<post_mode>=1 时保存的文件一起发送（只支持两个文件一起发送） |
| <err>            | 整型。错误代码。参考第 5 章。  |

2.2.7. AT+QHTTPREAD 通过 UART/USB 读取 HTTP(S)服务器响应信息

发送 HTTP(S) GET/POST 请求后，可使用 AT+QHTTPREAD 通过 UART/USB 端口从 HTTP(S)服务器读取 HTTP(S)响应信息。必须接收+QHTTPGET: <err>,<httprcode>[,<content\_length>]、+QHTTPPOST: <err>,<httprcode>[,<content\_length>]或+QHTTPPOSTFILE: <err>,<httprcode>[,<content\_length>]信息后才能执行 AT+QHTTPREAD。

AT+QHTTPREAD 通过 UART/USB 读取 HTTP(S)服务器响应信息

|                                    |  |
|------------------------------------|--|
| 测试命令<br>AT+QHTTPREAD=?             | 响应<br>+QHTTPREAD: (支持的<wait_time>范围)<br><br>OK   |
| 设置命令<br>AT+QHTTPREAD[=<wait_time>] | 响应<br>若参数格式正确：<br>CONNECT<br><输出 HTTP(S)响应信息><br>OK<br><br>当 body 读取完成或两个数据包接收间隔时间达到<wait_time>：<br>+QHTTPREAD: <err><br><br>若参数格式不正确或其他错误发生：<br>+CME ERROR: <err> |
| 特性说明                               | 该命令立即生效，参数配置不保存  |

参数

|             |   |
|-------------|---|
| <wait_time> | 整型。接收两个数据包之间的最大间隔时间。范围：1~65535。默认值：60。单位：秒。 |
| <err>       | 整型。错误代码。参考第5章。                              |

2.2.8. AT+QHTTPREADFILE 通过文件读取 HTTP(S)服务器响应信息

发送 HTTP(S) GET/POST 请求后，可以使用 AT+QHTTPREADFILE 通过文件从 HTTP(S)服务器读取 HTTP(S)响应信息，必须接收+QHTTPGET: <err>,<httprcode>[,<content\_length>]、+QHTTPPOST: <err>,<httprcode>[,<content\_length>]或+QHTTPPOSTFILE: <err>,<httprcode>[,<content\_length>]信息后才能执行 AT+QHTTPREADFILE。

**AT+QHTTPREADFILE 通过文件读取 HTTP(S)服务器响应信息**

|  |  |
|--|--|
| 测试命令<br>AT+QHTTPREADFILE=?                         | 响应<br>+QHTTPREADFILE: <file_name>,(支持的<wait_time>范围)<br><br>OK   |
| 设置命令<br>AT+QHTTPREADFILE=<file_name>[,<wait_time>] | 响应<br>若参数格式正确:<br>OK<br><br>当 body 读取完成或两个数据包接收间隔时间达到<wait_time>:<br>+QHTTPREADFILE: <err><br><br>若参数格式不正确或有其他错误发生:<br>+CME ERROR: <err> |
| 最大响应时间   | 取决于<wait_time>   |
| 特性说明   | 该命令立即生效，参数配置不保存  |

参数

|             |   |
|-------------|---|
| <wait_time> | 整型。接收两个数据包之间的最大间隔时间。范围：1~65535。默认值：60。单位：秒。 |
| <file_name> | 字符串类型。文件名称。最大长度为 80 字节。                     |
| <err>       | 整型。错误代码。参考第5章。                              |

### 2.2.9. AT+QHTTPSTOP 取消 HTTP(S)请求

MCU 可通过该命令取消 HTTP(S) GET/POST 请求，断开与 HTTP(S)的会话连接。

| AT+QHTTPSTOP 取消 HTTP(S)请求     |  |
|-------------------------------|--|
| 测试命令<br><b>AT+QHTTPSTOP=?</b> | 响应<br><b>OK</b>  |
| 执行命令<br><b>AT+QHTTPSTOP</b>   | 响应<br>若参数格式正确且无其他错误发生：<br><b>OK</b><br><br>若参数格式不正确或有其他错误发生：<br><b>+CME ERROR: &lt;err&gt;</b> |
| 最大响应时间                        | 10 秒   |
| 特性说明                          | 该命令立即生效，参数配置不保存  |

#### 参数

|                    |                |
|--------------------|----------------|
| <b>&lt;err&gt;</b> | 整型。错误代码。参考第5章。 |
|--------------------|----------------|

## 3 举例

### 3.1. 访问 HTTP 服务器

#### 3.1.1. 发送 HTTP GET 请求并读取响应信息

以下举例说明如何发送 HTTP GET 请求，启用 HTTP 响应头信息输出以及读取 HTTP GET 响应。

```
//发送 HTTP GET 响应示例
AT+QHTTPCFG="contextid",1           //配置PDP上下文ID为1。
OK
AT+QHTTPCFG="responseheader",1     //启用输出 HTTP 响应头信息。
OK
AT+QIACT?                            //查询 PDP 上下文状态。
OK
AT+QICSGP=1,1,"UNINET","",1       //配置 PDP 上下文为 1，APN 为中国联通的"UNINET"。需设置
AT+CFUN=1,1 使配置生效。
OK
AT+QIACT=1                          //激活 PDP 上下文 1。
OK                                  //激活成功。
AT+QIACT?                            //查询 PDP 上下文状态。
+QIACT: 1,1,1,"10.7.157.1"
OK
AT+QHTTPURL=23,80                   //设置要访问的 URL，设置超时时间为 80 秒。
CONNECT
HTTP://www.sina.com.cn/            //输入长度为 23 个字节的 URL。（此 URL 仅为示例。请根据实际
情况输入正确的 URL。）
OK
AT+QHTTPGET=80                     //发送 HTTP GET 请求，最大响应时间为 80 秒。
OK
+QHTTPGET: 0,200,601710            //若 HTTP 响应头信息中包括 CONTENT-LENGTH，则将上报
<content_length>信息。
//读取 HTTP GET 响应信息示例
//方法 1：读取 HTTP 响应信息并通过 UART 口将其输出。
AT+QHTTPREAD=80                   //读取 HTTP 响应信息并通过 UART 口将其输出。HTTP 会话关
闭的最长等待时间为 80 秒。
```

```

CONNECT
HTTP/1.1 200 OK <CR><LF>           //HTTP 响应头信息和响应体。
Server: nginx<CR><LF>
Date: Tue, 12 Sep 2017 05:57:29 GMT<CR><LF>
Content-Type: text/html<CR><LF>
Content-Length: 601710<CR><LF>
Connection: close<CR><LF>
Last-Modified: Tue, 12 Sep 2017 05:54:48 GMT<CR><LF>
Vary: Accept-Encoding<CR><LF>
Expires: Tue, 12 Sep 2017 05:58:28 GMT<CR><LF>
Cache-Control: max-age=60<CR><LF>
X-Powered-By: shci_v1.03<CR><LF>
Age: 1<CR><LF>
.....<CR><LF>                     //此处响应信息省略。
<CR><LF>
<body>
OK

+QHTTPREAD: 0                       //成功读取 HTTP 响应头信息和响应体。

//方法 2: 读取 HTTP 响应信息并将其储存到 UFS 文件中。

AT+QHTTPREADFILE="UFS:1.txt",80     //读取 HTTP 响应头信息和响应体并将其储存到 UFS:1.txt。
                                     HTTP 会话关闭的最长等待时间为 80 秒。

OK

+QHTTPREADFILE: 0                   //成功储存 HTTP 响应头信息和响应体。

```

### 3.1.2. 发送 HTTP POST 请求并读取响应信息

#### 3.1.2.1. 通过 UART/USB 获取 POST 请求体

以下举例说明如何发送 HTTP POST 请求,通过 UART 口读取 POST 请求体以及如何读取 HTTP POST 响应信息。

```

AT+QHTTPCFG="contextid",1           //配置PDP上下文ID为1。
OK
AT+QIACT?                           //查询 PDP 上下文状态。
OK
AT+QICSGP=1,1,"UNINET","",1        //配置 PDP 上下文为 1, APN 为中国联通的"UNINET"。需设置
AT+CFUN=1,1 使配置生效。
OK
AT+QIACT=1                           //激活 PDP 上下文 1。
OK                                     //激活成功

```

```

AT+QIACT? //查询 PDP 上下文状态。
+QIACT: 1,1,1,"172.22.86.226"

OK
AT+QHTTPURL=59,80 //设置要访问的 URL，设置超时时间为 80 秒。
CONNECT
http://api.efxnow.com/DEMOWebServices2.8/Service.asmx/Echo? //输入长度为 59 个字节的 URL。
//（此 URL 仅为示例，请根据实际情况输入正确的 URL。）

OK
AT+QHTTPPOST=20,80,80 //发送 HTTP POST 请求,通过 UART 获取 POST 请求体。POST
//请求体的最大输入时间和响应时间皆为 80 秒。
CONNECT
Message=HelloQuectel //输入长度为 20 个字节的 POST 请求体。（此 POST 请求体仅
//为示例，请根据实际情况输入正确的 POST 请求体。）

OK
+QHTTPPOST: 0,200,177 //若 HTTP 响应头信息包含 CONTENT-LENGTH，则将上报
//<content_length>信息。

AT+QHTTPREAD=80 //读取 HTTP 响应信息并通过 UART 口将其输出。HTTP 会话关
//闭的最长等待时间为 80 秒。
CONNECT
<?xml version="1.0" encoding="utf-8"?>
<string xmlns="httpHTTPs://api.efxnow.com/webservices2.3">Message='HelloQuectel' ASCII:72
101 108 108 111 81 117 101 99 116 101 108 </string> //输出 HTTP 响应信息。
OK
+QHTTPREAD: 0 //成功输出 HTTP 响应信息
    
```

### 3.1.2.2. 从文件系统获取 POST 请求体

以下举例说明如何发送 HTTP POST 请求，通过文件系统读取 POST 请求体，以及将 HTTP POST 响应储存在文件系统。

```

AT+QHTTPCFG="contextid",1 //配置PDP上下文ID为1。
OK
AT+QIACT? //查询 PDP 上下文状态。
OK
AT+QICSGP=1,1,"UNINET","",",",1 //配置 PDP 上下文为 1，APN 为中国联通的"UNINET"。需设置
//AT+CFUN=1,1 使配置生效。
OK
AT+QIACT=1 //激活 PDP 上下文 1。
OK //激活成功。
AT+QIACT? //查询 PDP 上下文状态。
    
```

```
+QIACT: 1,1,1,"172.22.86.226"

OK
AT+QHTTPURL=59,80 //设置要访问的 URL，设置超时时间为 80 秒。
CONNECT
http://api.efxnow.com/DEMOWebServices2.8/Service.asmx/Echo? //输入长度为 59 个字节的 URL。
//（此 URL 仅为示例，请根据实际情况输入正确的 URL。）

OK
//来自 UFS 文件的 POST 请求信息，读取 HTTP 响应信息并将其储存到 UFS 文件中。
AT+QHTTPPOSTFILE="UFS:2.txt",80 //发送 HTTP POST 请求，从 UFS:2.txt 中获取 POST 请求体，
//最大响应时间为 80 秒。

OK
+QHTTPPOSTFILE: 0,200,177 //HTTP POST 请求发送成功后，方可执行
AT+QHTTPREADFILE="UFS:3.txt",80 //读取 HTTP 响应信息并将其储存到 UFS:3.txt。HTTP 会话关
//闭的最长等待时间为 80 秒。

OK
+QHTTPREADFILE: 0 //HTTP 响应信息储存成功。
```

## 3.2. 访问 HTTPS 服务器

### 3.2.1. 发送 HTTPS GET 请求并读取响应信息

以下举例说明如何发送 HTTPS GET 请求，启用 HTTPS 响应头信息输出以及如何读取 HTTPS GET 响应信息。

```
//发送HTTPS GET请求示例
AT+QHTTPCFG="contextid",1 //配置PDP上下文ID为1。
OK
AT+QHTTPCFG="responseheader",1 //启用输出 HTTPS 响应头信息。
OK
AT+QIACT? //查询 PDP 上下文状态。
OK
AT+QICSGP=1,1,"UNINET","",1 //配置 PDP 上下文为 1，APN 为中国联通的"UNINET"。需设
//置 AT+CFUN=1,1 使配置生效。
OK
AT+QIACT=1 //激活 PDP 上下文 1。
OK //激活成功。
AT+QIACT? //查询 PDP 上下文状态。
+QIACT: 1,1,1,"10.7.157.1"
```

```

OK
AT+QHTTPCFG="sslctxid",1 //设置 SSL 上下文 ID 为 1。
OK
AT+QSSLCFG="sslversion",1,1 //设置 SSL 版本为 1，表示 TLSV1.0。
OK
AT+QSSLCFG="ciphersuite",1,0x0005 //设置 SSL 加密套件为 0x0005，表示 RC4-SHA。
OK
AT+QSSLCFG="secllevel",1,0 //设置 SSL 验证级别为 0，表示无身份验证模式。
OK
AT+QHTTPURL=22,80 //设置要访问的 URL，设置超时时间为 80 秒。
CONNECT
https://www.alipay.com //输入长度为 22 个字节的 URL。（此 URL 仅为示例，请根据实际
//情况输入正确的 URL。）
OK
AT+QHTTPGET=80 //发送 HTTPS GET 请求，最大响应时间为 80 秒。
OK
+QHTTPGET: 0,200,21472 //若 HTTPS 响应头信息包含 CONTENT-LENGTH，则将上报
<content_length>信息。

//读取 HTTPS 响应信息示例

//方法 1：读取 HTTPS 响应信息并将其通过 UART 输出。
AT+QHTTPREAD=80 //读取 HTTPS 响应信息并将其通过 UART 输出。HTTP 会
//话关闭的最长等待时间为 80 秒。
CONNECT //HTTPS 响应头信息和响应体。
HTTP/1.1 200 OK<CR><LF>
Server: Tengine/2.1.0<CR><LF>
Date: Tue, 12 Sep 2017 05:54:34 GMT <CR><LF>
Content-Type: text/html; charset=utf-8<CR><LF>
Content-Length: 21451<CR><LF>
Connection: keep-alive <CR><LF>
..... <CR><LF> //此处响应信息省略。
<CR><LF>
<body>
OK
+QHTTPREAD: 0 //HTTPS 响应头信息和响应体读取成功。

//方法 2：读取 HTTPS 响应信息并将其储存到 UFS 文件中。
AT+QHTTPREADFILE="UFS:4.txt",80 //读取 HTTPS 响应头信息和响应体并将其储存在 UFS:4.txt。
//HTTP 会话关闭的最长等待时间为 80 秒。
OK

```

+QHTTPREADFILE: 0 //HTTPS 响应头信息和响应体储存成功。

### 3.2.2. 发送 HTTPS POST 请求并读取响应信息

#### 3.2.2.1. 从 UART/USB 获取 POST 请求体

以下举例说明如何发送 HTTPS POST 请求,通过 UART 读取 POST 请求体以及如何读取 HTTPS POST 响应信息。

```

AT+QHTTPCFG="contextid",1 //配置PDP上下文ID为1。
OK
AT+QIACT? //查询 PDP 上下文状态。
OK
AT+QICSGP=1,1,"UNINET","",1 //配置 PDP 上下文为 1, APN 为中国联通的"UNINET"。需设置 AT+CFUN=1,1 使配置生效。
OK
AT+QIACT=1 //激活 PDP 上下文 1。
OK //激活成功。
AT+QIACT? //查询 PDP 上下文状态。
+QIACT: 1,1,1,"172.22.86.226"

OK
AT+QHTTPCFG="sslctxid",1 //设置 SSL 上下文 ID 为 1。
OK
AT+QSSLCFG="sslversion",1,1 //设置 SSL 版本为 1, 表示 TLSV1.0。
OK
AT+QSSLCFG="ciphersuite",1,0x0005 //设置 SSL 加密套件为 0x0005, 表示 RC4-SHA。
OK
AT+QSSLCFG="seclvl",1,2 //设置 SSL 验证级别为 2, 表示需通过 AT+QFUPL 上传 CA 证书, 客户端证书和客户端私钥。
OK
AT+QSSLCFG="cacert",1,"UFS:cacert.pem"
OK
AT+QSSLCFG="clientcert",1,"UFS:clientcert.pem"
OK
AT+QSSLCFG="clientkey",1,"UFS:clientkey.pem"
OK
AT+QHTTPURL=45,80 //设置要访问的 URL, 设置超时时间为 80 秒。
CONNECT
HTTPS://220.180.239.212:8011/processorder.php //输入长度为 45 个字节的 URL。(此 URL 仅为示例, 请根据实际情况输入正确的 URL。)
OK
    
```

```

AT+QHTTPPOST=48,80,80 //发送 HTTPS POST 请求，通过 UART 获取 POST 请求体。POST
                           请求体的最大输入时间和响应时间为 80 秒。

CONNECT
Message=1111&Appleqty=2222&Orangeqty=3333&find=1 //输入长度为 48 个字节的 POST 请求体。
                                                    (此 POST 请求体仅为示例，请根据实际情
                                                    况输入正确的 POST 请求体。)

OK

+QHTTPPOST: 0,200,285 //若 HTTPS 响应头信息包含 CONTENT-LENGTH，则将上报
                           <content_length>信息。

AT+QHTTPREAD=80 //读取 HTTPS 响应信息并将其通过 UART 输出。HTTP 会话关闭
                           的最长等待时间为 80 秒。

CONNECT //HTTPS 响应信息读取成功。
<html>
<head>
<title>Quectel's Auto Parts - Order Results</title>
</head>
<body>
<h1>Quectel's Auto Parts</h1>
<h2>Order Results</h2>

<p>Order processed at 02:49, 27th December</p><p>Your order is as follows: </p>1111
message<br />2222 apple<br />3333 orange<br /></body>
</html>
OK

+QHTTPREAD: 0 //HTTPS 响应信息输出成功。
    
```

### 3.2.2.2. 从文件系统获取 POST 请求体

以下举例说明如何发送 HTTPS POST 请求，从文件系统读取 POST 请求体以及如何将 HTTPS POST 响应信息储存到文件系统。

```

AT+QHTTPCFG="contextid",1 //配置PDP上下文ID为1。
OK
AT+QIACT? //查询 PDP 上下文状态。
OK
AT+QICSGP=1,1,"UNINET","",",",1 //配置 PDP 上下文为 1，APN 为中国联通的"UNINET"。需设置
AT+CFUN=1,1 使配置生效。
OK
AT+QIACT=1 //激活 PDP 上下文 1。
OK //激活成功。
AT+QIACT? //查询 PDP 上下文状态。
+QIACT: 1,1,1,"172.22.86.226"
    
```

```

OK
AT+QHTTPCFG="sslctxid",1           //设置 SSL 上下文 ID 为 1。
OK
AT+QSSLCFG="sslversion",1,1       //设置 SSL 版本为 1，表示 TLSV1.0。
OK
AT+QSSLCFG="ciphersuite",1,0x0005 //设置 SSL 加密套件为 0x0005，表示 RC4-SHA。
OK
AT+QSSLCFG="secllevel",1,2        //设置 SSL 验证级别为 2，表示需通过 AT+QFUPL 上传 CA 证书，客户端证书和客户端私钥。
OK
AT+QSSLCFG="cacert",1,"UFS:cacert.pem"
OK
AT+QSSLCFG="clientcert",1,"UFS:clientcert.pem"
OK
AT+QSSLCFG="clientkey",1,"UFS:clientkey.pem"
OK
AT+QHTTPURL=45,80                 //设置要访问的 URL，设置超时时间为 80 秒。
CONNECT
https://220.180.239.212:8011/processorder.php //输入长度为 45 个字节的 URL。（此 URL 仅为示例，请根据实际情况输入正确的 URL。）
OK
//POST 请求信息来自 UFS 文件，读取 HTTPS 响应信息并将其储存在 UFS 文件中。
AT+QHTTPPOSTFILE="UFS:5.txt",80     //发送 HTTPS POST 请求，从 UFS:5.txt 获取 POST 请求体，最大响应时间为 80 秒。
OK
+QHTTPPOSTFILE: 0,200,177          //HTTPS POST 请求发送成功后，方可执行 AT+QHTTPREADFILE。
AT+QHTTPREADFILE="UFS:6.txt",80    //读取 HTTPS 响应信息并将其储存在 UFS:6.txt"。HTTP 会话关闭的最长等待时间为 80 秒。
OK
+QHTTPREADFILE: 0                 //HTTPS 响应信息储存成功。

```

# 4 常见问题处理

## 4.1. HTTP(S) AT 命令执行失败

执行 HTTP(S) AT 命令后，若返回 **+CME ERROR: <err>**，请检查是否插入了(U)SIM 卡，并注意执行 **AT+CPIN?**后是否返回了 **+CPIN: READY**。

## 4.2. PDP 激活失败

若使用 **AT+QIACT** 激活 PDP 失败，请检查以下配置：

1. 查询是否通过 **AT+CGATT?**附着了 PS 域。若未附着 PS 域，请先执行 **AT+CGATT=1** 附着 PS 域。
2. 通过 **AT+CGREG?**查询 PS 域状态并确保已注册了 PS 域。
3. 通过 **AT+QICSGP** 查询 PDP 上下文参数，并确保已设置了指定 PDP 上下文的 APN。
4. 确保指定 PDP 上下文 ID 既不被 PPP 使用，也不通过 **AT+CGACT** 激活。
5. 根据 3GPP 规范，模块仅支持同时激活三个 PDP 上下文，因此激活 PDP 上下文的数量必须小于 3。

若上述配置无误，使用 **AT+QIACT** 激活 PDP 上下文仍失败，请重启模块。重启模块后，请至少检查三次上述提及的配置，每次间隔 10 分钟，以避免频繁重启模块。

## 4.3. DNS 解析失败

执行 **AT+QHTTPGET**、**AT+QHTTPPOST** 和 **AT+QHTTPPOSTFILE** 后，若返回 **+CME ERROR: 714**（714: HTTP(S) DNS 错误），请检查以下配置：

1. 确认 HTTP(S)服务器域名有效。
2. 通过 **AT+QIACT?**查询 PDP 上下文状态，确保成功激活了指定的 PDP 上下文。
3. 通过 **AT+QIDNSCFG** 查询 DNS 服务器地址，确保 DNS 服务器的地址不是 **"0.0.0.0"**。

若 DNS 服务器的地址为 **"0.0.0.0"**，有两种解决方法：

1. 通过 **AT+QIDNSCFG** 重新分配一个有效 DNS 服务器地址。
2. 通过 **AT+QIDEACT** 去激活 PDP 上下文，通过 **AT+QIACT** 重新激活 PDP 上下文。

## 4.4. 数据模式进入失败

执行 **AT+QHTTPURL**、**AT+QHTTPGET**、**AT+QHTTPPOST** 和 **AT+QHTTPREAD** 后，若返回**+CME ERROR: 704**（704: HTTP(S) UART 繁忙），请检查数据模式下是否存在其他端口，由于模块仅支持一个端口处于数据模式，若存在其他端口，请退出其他端口数据模式后重新执行这些命令。

## 4.5. GET/POST 请求发送失败

若执行 **AT+QHTTPGET**、**AT+QHTTPGETEX**、**AT+QHTTPPOST** 和 **AT+QHTTPPOSTFILE** 失败，请检查以下配置：

1. 确保通过 **AT+QHTTPURL** 输入的 URL 有效且可以访问。
2. 确保指定服务器支持 GET/POST 相关命令。
3. 确保 PDP 上下文已激活。

若上述配置无误，通过 **AT+QHTTPGET**、**AT+QHTTPPOST** 和 **AT+QHTTPPOSTFILE** 发送 GET/POST 请求仍失败，请通过 **AT+QIDEACT** 去激活 PDP 上下文，并通过 **AT+QIACT** 重新激活 PDP 上下文。若激活 PDP 上下文失败，详细信息请参考 [第 4.2 章](#) 解决此问题。

## 4.6. 响应信息读取失败

通过 **AT+QHTTPREAD** 和 **AT+QHTTPREADFILE** 读取响应信息之前，首先要执行 **AT+QHTTPGET**、**AT+QHTTPPOST** 和 **AT+QHTTPPOSTFILE**，之后将上报以下 URC：

```
+QHTTPGET: <err>,<httprcode>[,<content_length>]
+QHTTPPOST: <err>,<httprcode>[,<content_length>]
+QHTTPPOSTFILE: <err>,<httprcode>[,<content_length>]
```

执行 **AT+QHTTPREAD** 和 **AT+QHTTPREADFILE** 时，若发生错误，如上报**+CME ERROR: 717**（717: HTTP(S) socket 读取错误），请通过 **AT+QHTTPGET**、**AT+QHTTPPOST** 和 **AT+QHTTPPOSTFILE** 向 HTTP(S)服务器重新发送 HTTP(S) GET/POST 请求；若请求仍发送失败，详细信息请参考 [第 4.5 章](#) 解决此问题。

## 5 错误代码

<err>表示与移动设备或网络有关的错误代码。相关详细信息参见下表。

表 3: 错误代码列表

| <err>错误代码 | 英文描述                             | 中文描述                  |
|-----------|----------------------------------|-----------------------|
| 0         | Operation successful             | 操作成功                  |
| 701       | HTTP(S) unknown error            | HTTP(S) 未知错误          |
| 702       | HTTP(S) timeout                  | HTTP(S) 超时            |
| 703       | HTTP(S) busy                     | HTTP(S) 繁忙            |
| 704       | HTTP(S) UART busy                | HTTP(S) UART 繁忙       |
| 705       | HTTP(S) no GET/POST requests     | HTTP(S) 无 GET/POST 请求 |
| 706       | HTTP(S) network busy             | HTTP(S) 网络繁忙          |
| 707       | HTTP(S) network open failed      | HTTP(S) 网络连接失败        |
| 708       | HTTP(S) network no configuration | HTTP(S) 网络未配置         |
| 709       | HTTP(S) network deactivated      | 去激活 HTTP(S) 网络        |
| 710       | HTTP(S) network error            | HTTP(S) 网络错误          |
| 711       | HTTP(S) URL error                | HTTP(S) URL 错误        |
| 712       | HTTP(S) empty URL                | HTTP(S) 空 URL         |
| 713       | HTTP(S) IP address error         | HTTP(S) IP 地址错误       |
| 714       | HTTP(S) DNS error                | HTTP(S) DNS 错误        |
| 715       | HTTP(S) socket create error      | HTTP(S) socket 创建错误   |
| 716       | HTTP(S) socket connect error     | HTTP(S) socket 连接错误   |
| 717       | HTTP(S) socket read error        | HTTP(S) socket 读取错误   |

|     |                               |                     |
|-----|-------------------------------|---------------------|
| 718 | HTTP(S) socket write error    | HTTP(S) socket 写入错误 |
| 719 | HTTP(S) socket closed         | HTTP(S) socket 关闭   |
| 720 | HTTP(S) data encode error     | HTTP(S) 数据编码错误      |
| 721 | HTTP(S) data decode error     | HTTP(S) 数据解码错误      |
| 722 | HTTP(S) read timeout          | HTTP(S) 读取超时        |
| 723 | HTTP(S) response failed       | HTTP(S)响应失败         |
| 724 | Incoming call busy            | 来电忙                 |
| 725 | Voice call busy               | 语音通话忙               |
| 726 | Input timeout                 | 输入超时                |
| 727 | Wait data timeout             | 数据等待超时              |
| 728 | Wait HTTP(S) response timeout | HTTP(S)等待响应超时       |
| 729 | Memory allocation failed      | 内存分配不足              |
| 730 | Invalid parameter             | 无效参数                |

## 6 HTTP(S)响应错误代码

<httpsrcode>表示 HTTP(S)服务器响应代码。相关详细信息参见下表。

表 4: HTTP(S)响应代码列表

| <httpsrcode>响应代码 | 英文描述                  | 中文描述    |
|------------------|-----------------------|---------|
| 200              | OK                    | 成功      |
| 403              | Forbidden             | 禁止      |
| 404              | Not found             | 未发现     |
| 409              | Conflict              | 冲突      |
| 411              | Length required       | 需输入长度   |
| 500              | Internal server error | 内部服务器错误 |

# 7 附录 参考文档和术语缩写

表 5: 参考文档

| 序号  | 文档名称   | 描述  |
|-----|--|---|
| [1] | Quectel_EC200x&EG912Y&EC600S 系列_TCP(IP)_应用指导 | 适用于 EC200x 系列、EG912Y 系列和 EC600S-CN 模块 TCP/IP 应用指导 |
| [2] | Quectel_EC200x&EC600S&EG912Y 系列_SSL_应用指导     | 适用于 EC200x 系列、EG912Y 系列和 EC600S-CN 模块 SSL 应用指导    |

表 6: 术语缩写

| 缩写    | 英文全称  | 中文全称        |
|-------|---|-------------|
| DNS   | Domain Name Server                          | 域名服务器       |
| DTR   | Data Terminal Ready                         | 数据终端就绪      |
| HTTP  | Hypertext Transfer Protocol                 | 超文本传输协议     |
| HTTPS | Hypertext Transfer Protocol Secure          | 超文本传输安全协议   |
| ID    | Identification                              | 身份识别        |
| PDP   | Packet Data Protocol                        | 分组数据协议      |
| PPP   | Point-to-Point Protocol                     | 点对点协议       |
| PS    | Packet Switch                               | 分组交换        |
| SSL   | Security Socket Layer                       | 安全 Socket 层 |
| UART  | Universal Asynchronous Receiver/Transmitter | 通用异步收发传输器   |
| URI   | Uniform Resource Identifier                 | 统一资源标示符     |
| URL   | Uniform Resource Locator                    | 统一资源定位符     |

---

|        |  |               |
|--------|--|---------------|
| URC    | Unsolicited Result Code                | 未经请求的结果代码     |
| USB    | Universal Serial Bus                   | 通用串行总线        |
| (U)SIM | (Universal) Subscriber Identity Module | (通用) 用户身份识别模块 |
| UFS    | UNIX File System                       | UNIX 文件系统     |

---