GSM HTTP

AT Commands Manual

GSM/GPRS Module Series

Rev. GSM_HTTP_AT_Commands_Manual_V1.4

Date: 2017-11-13

Status: Released

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
7th Floor, Hongye Building, No.1801 Hongmei Road, Xuhui District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://quectel.com/support/sales.htm

For technical support, or to report documentation errors, please visit:
http://quectel.com/support/technical.htm
Or email to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2009-01-04	Jay XIN	Initial
1.1	2012-06-04	Will SHAO	Added Command AT+QHTTPDL
1.2	2015-04-13	Will SHAO	Added applicable modules
1.3	2015-12-29	Jelly WANG	Modified the problem that some AT commands are displayed incompletely.
1.4	2017-11-13	Sherlock ZHAO	Added command AT+QHTTPCFG

Contents

1 Introduction

Quectel module provides an internal TCP/IP stack that is driven by AT commands and enables the host application to easily access the Internet services which include TCP service, UDP service, FTP service and HTTP service, etc. This document is a reference guide to all the AT commands and responses defined for HTTP service. The advantage of this solution is that it eliminates the need for the application manufacturer to implement own HTTP protocol, thus minimizing cost and time to integrate Internet connectivity into a new or existing host application.

This document is applicable to all Quectel GSM modules.

1.1. AT Command Syntax

Test Command	**AT+<*x*>=?**	This command returns the list of parameters and value ranges set by the corresponding Write Command or internal processes.
Read Command	**AT+<*x*>?**	This command returns the currently set value of the parameter or parameters.
Write Command	**AT+<x>=<…>**	This command sets the user-definable parameter values.
Execution Command	**AT+<x>**	This command reads non-variable parameters affected by internal processes in the GSM engine.

2 Description of HTTP AT Commands

2.1. AT+QHTTPURL Set URL of HTTP Server

AT+QHTTPURL Set URL of HTTP Server	
Test Command **AT+QHTTPURL=?**	Response **+QHTTPURL: (1-2048),(1-65535)** **OK**
Write Command **AT+QHTTPURL=<url_len>[,<input_time>]**	Response **CONNECT** If there is an error related to ME functionality: **ERROR** **+CME ERROR: <err>**
Maximum Response Time	Determined by **<input_time>** (default 120s)

Parameter

<url_len>	Numeric type. The length of URL. The range is 1-2048. Unit: byte.
<input_time>	Numeric type. The maximum time for inputting URL. The range is 1-65535, and the default value is 120. Unit: second.
<err>	Integer type. The error code of the operation. Please refer to ***Chapter 4***.

2.2. AT+QHTTPGET Send GET Request to HTTP Server

AT+QHTTPGET Send GET Request to HTTP Server	
Test Command **AT+QHTTPGET=?**	Response **+QHTTPGET: (1-65535)[,(15-65535)[,(0-102400)]]** **OK**
Write Command	Response

AT+QHTTPGET=<to_read_time>[,<wait_time>[,<data_size>]]	**OK** If there is an error related to ME functionality: **ERROR** **+CME ERROR: <err>**
Maximum Response Time	Determined by **<wait_time>** (default 120s)

Parameter

<to_read_time>	The maximum timeout waiting for reading data from server after sending HTTP GET request to server successfully. **AT+QHTTPREAD** will be invalid if idle time after **AT+QHTTPGET** is longer than the time of **<to_read_time>**. The range is 1-65535, and the default value is 120. Unit: second.
<wait_time>	The time waiting for data from server after sending HTTP GET request to server successfully. If there is no data coming from server within timeout value, an error will be returned. The range is 15-65535, and the default value is 120. Unit: second.
<data_size>	Integer type. Indicates the data bytes to download from HTTP server. The range is 0-102400, and the default value is 0.
<err>	Integer type. The error code of the operation. Please refer to ***Chapter 4***.

2.3. AT+QHTTPPOST Send POST Request to HTTP Server

AT+QHTTPPOST Send POST Request to HTTP Server	
Test Command **AT+QHTTPPOST=?**	Response **+QHTTPPOST: (1-29696),(1-65535),(1-65535)** **OK**
Write Command **AT+QHTTPPOST=<body_size>[,<input_time>[,<to_read_time>]]**	Response **CONNECT** **<data>** **OK** If there is an error related to ME functionality: **ERROR** **+CME ERROR: <err>**
Maximum Response Time	120s

Parameter

<body_size>	Size of the post body. The range is 1-29696. Unit: byte.
<input_time>	Maximum time for inputting the data. The range is 1-65535, and the default value is 120.
<to_read_time>	The maximum timeout waiting for reading data from server after sending HTTP POST request to server successfully. **AT+QHTTPREAD** will be invalid if the idle time after **AT+QHTTPGET** is longer than the time of **<to_read_time>**. The range is 1-65535, and the default value is 120. Unit: second.
<data>	The POST data to input from UART.
<err>	Integer type. The error code of the operation. Please refer to ***Chapter 4***.

2.4. AT+QHTTPREAD Read Response from HTTP Server

AT+QHTTPREAD	**Read Response from HTTP Server**
Test Command **AT+QHTTPREAD=?**	Response **+QHTTPREAD: (1-65535)** **OK**
Write Command **AT+QHTTPREAD=<wait_time>**	Response **CONNECT** **<data>** **OK** If there is an error related to ME functionality: **ERROR** **+CME ERROR: <err>**
Maximum Response Time	Determined by **<wait_time>** (default 120s)

Parameter

<wait_time>	The maximum waiting time for reading response from HTTP Server. HTTP session will be closed when timeout. The range is 1-65535, and the default value is 120. Unit: second.
<data>	The data of HTTP server responds.
<err>	Integer type. The error code of the operation. Please refer to ***Chapter 4***.

2.5. AT+QHTTPDL Download a File from HTTP Server

AT+QHTTPDL Download a File from HTTP Server	
Test Command **AT+QHTTPDL=?**	Response **+QHTTPDL: "filename"[,<length>[,(1-65535)]]** **OK**
Write Command **AT+QHTTPDL="<filename>"[,<length>[,<wait_time>]]**	Response **OK** Finally, if the file is downloaded successfully, response: **+QHTTPDL: <dl size>,<content-length>,<errcode>** If there is an error related to ME functionality: **ERROR** **+CME ERROR: <err>**
Reference	Determined by **<wait_time>** (default 120s)

Parameter

<filename>	The path of the file to be stored, such as "RAM:1.txt".
<length>	The maximum size of the file to be downloaded. The default value is 10240. Unit: byte. It is only used for RAM file.
<wait_time>	The maximum waiting time for downloading a file from HTTP Server. HTTP session will be closed when timeout. The range is 1-65535, and the default value is 120. Unit: second.
<dl size>	The length of the data downloaded.
<content-length>	The content length. If the **<content-length>** is unknown, then set it to -1.
<errcode>	If all data has been downloaded, the **<errcode>** is 0, else it is a numeric to indicate the type of error. Please refer to ***Chapter 4***.
<err>	Integer type. The error code of the operation. Please refer to ***Chapter 4***.

2.6. AT+QHTTPCFG Configure Parameters for HTTP Service

AT+QHTTPCFG Configure Parameters for HTTP Service	
Test Command **AT+QHTTPCFG=?**	Response **+QHTTPCFG: ("requestheader","responseheader"),(0,1)** **OK**

Write Command **AT+QHTTPCFG="requestheader"[,<requestheader>]**	Response If **<requestheader>** is not omitted, response: **OK** or **+CME ERROR: <errorcode>** Else, query the current setting **+QHTTPCFG: "requestheader",<requestheader>** **OK**
Write Command **AT+QHTTPCFG="responseheader"[,<responseheader>]**	Response If **<responseheader>** is not omitted, response: **OK** or **+CME ERROR: <errorcode>** Else, query the current setting **+QHTTPCFG: "responseheader",<responseheader>** **OK**
Reference	

Parameter

<requestheader>	Numeric type. It indicates if customizing HTTP request header is allowed or not. 0 Disable the customization HTTP request header 1 Enable the customization HTTP request header
<responseheader>	Numeric type. It indicates if outputting HTTP response header is allowed or not 0 Disable the output of HTTP response header 1 Enable the output of HTTP response header
<errorcode>	Integer type. The error code of the operation. Please refer to ***Chapter 4***.

3 Examples

3.1. Send GET Request to HTTP Server

```
AT+QIFGCNT=0
OK
AT+QICSGP=1,"CMNET"          //Set APN
OK
AT+QIREGAPP                  //Optional
OK
AT+QIACT                     //Optional
OK
AT+QHTTPURL=79,30            //Set URL
CONNECT
<Input data>
//For example, input 79 bytes:
http://api.efxnow.com/DEMOWebServices2.8/Service.asmx/Echo?Message=helloquectel.
OK

AT+QHTTPGET=60               //Send GET Request to HTTP server.
OK
AT+QHTTPREAD=30              //Read the response of HTTP server.
CONNECT
<Output data>                //Output the response data of HTTP server to UART.

//For example, UART outputs:
<?xml version="1.0" encoding="utf-8"?>
<string xmlns="https://api.efxnow.com/webservices2.3">Message='helloquectel' ASCII:104 101 108 108 111 113 117 101 99 116 101 108 </string>.
OK

AT+QIDEACT                   //Deactivate PDP context.
DEACT OK
```

3.2. Send POST Request to HTTP Server

```
AT+QIFGCNT=0
OK

AT+QICSGP=1,"CMNET"          //Set APN
OK

AT+QIREGAPP                  //Optional
OK

AT+QIACT                     //Optional
OK

AT+QHTTPURL=58,30            //Set URL
CONNECT
<Input data>
//For example, input 58 bytes:
http://api.efxnow.com/DEMOWebServices2.8/Service.asmx/Echo

OK

//POST the data whose size is 18 bytes and the maximum latency time for inputting is 50s.
//It is recommended to set the latency time as long as enough to download all the data in the latency time.
AT+QHTTPPOST=18,50,10
CONNECT
//This means module enters into data mode and is ready to receive data from UART.
//For example, input 18 bytes: Message=helloworld.
OK
//This means all data has been received, and DCD is set to high.

AT+QHTTPREAD=30              //Read the response of HTTP server.

CONNECT
<Output data>                //Output the response data of HTTP server to UART.
//For example, UART outputs:
<?xml version="1.0" encoding="utf-8"?>
<string xmlns="https://api.efxnow.com/webservices2.3">Message='helloworld' ASCII:104 101 108 108 111 119 111 114 108 100 </string>

OK
AT+QIDEACT                   //Deactivate PDP context.
DEACT OK
```

3.3. Download a File From HTTP Server

```
AT+QIFGCNT=0
OK
AT+QICSGP=1,"CMNET"          //Set APN
OK
AT+QIREGAPP                  //Optional
OK
AT+QIACT                     //Optional
OK

AT+QHTTPURL=36,30            //Set URL
CONNECT
<Input data>
//for example, input 36 bytes:
http://220.180.239.212:8005/100K.txt

OK

AT+QHTTPGET=60               //Send GET Request to HTTP server.
OK

AT+QHTTPDL="RAM:1.TXT",1024
//Download the file to RAM:1.TXT, and the maximum size of the file is 1024 bytes.

+QHTTPDL: 1024,102400,401

OK
```

3.4. Customize HTTP Header and Show HTTP Response Header

```
AT+QHTTPCFG="requestheader",1

OK
AT+QHTTPCFG ="responseheader",1

OK

AT+QHTTPURL=58
CONNECT
//Input URL "http://api.efxnow.com/DEMOWebServices2.8/Service.asmx/Echo".
OK
```

```
AT+QHTTPPOST=229
CONNECT
//Input customers' own header like below:
POST /DEMOWebServices2.8/Service.asmx/Echo HTTP/1.1
Host: api.efxnow.com
Accept: */*
User-Agent: QUECTEL_MODULE
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 18

Message=helloworld

OK

AT+QHTTPREAD=30
CONNECT
//The data returned from server is as below:
HTTP/1.1 200 OK
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Server: Microsoft-IIS/7.0
X-AspNet-Version: 2.0.50727
X-Powered-By: ASP.NET
Date: Sat, 17 Aug 2013 05:06:23 GMT
Content-Length: 170

<?xml version="1.0" encoding="utf-8"?>
<string xmlns="https://api.efxnow.com/webservices2.3">Message='helloworld' ASCII:104 101 108
108 111 119 111 114 108 100 </string>
OK
```

4 Summary of ERROR Codes

The error code <err> indicates an error related to mobile equipment or network. The details about <err> are described in the following table.

Table 1: Summary of Error Codes

<err>	Meaning
3801	HTTP timeout
3802	HTTP busy
3803	HTTP UART busy
3804	HTTP get no request
3805	HTTP network busy
3806	HTTP network open failed
3807	HTTP network no configuration
3808	HTTP network deactivated
3809	HTTP network error
3810	HTTP URL error
3811	HTTP empty URL
3812	HTTP IP address error
3813	HTTP DNS error
3814	HTTP socket create error
3815	HTTP socket connect error

3816	HTTP socket read error
3817	HTTP socket write error
3818	HTTP socket closed
3819	HTTP data encode error
3820	HTTP data decode error
3821	HTTP read timeout
3822	HTTP response failed
3823	Incoming call busy
3824	Voice call busy
3825	Input timeout
3826	Wait data timeout
3827	Wait HTTP response timeout
3828	Memory allocation failed
3829	HTTP need relocation
3830	Queclocator location failed
4000	Exceed max length
4001	Open file failed
4002	Write file failed
4003	Get size failed
4004	Read failed
4005	List file failed
4006	Delete file failed
4007	Get Disk info failed

4008	No space
4009	Timeout
4010	File not found
4011	File too large
4012	File already exist
4013	Invalid parameter
4014	Driver error
4015	Create failed
4016	Access denied
4017	File too large