BC95-G&BC68 CoAP Application Note

NB-IoT Module Series

Rev. Quectel_BC95-G&BC68_CoAP_Application_Note_V1.0

Date: 2018-12-18

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://www.quectel.com/support/sales.htm

For technical support, or to report documentation errors, please visit:
http://www.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	2018-12-18	Oven TAO/ Benjamin LU	Initial

Contents

Table Index

1 Introduction

This document mainly introduces how to use the CoAP function of Quectel BC95-G and BC68 modules through AT commands.

1.1. Definitions

- <CR>: Carriage return character;
- <LF>: Line feed character;
- <..>: Parameter name. Angle brackets do not appear on command line;
- [..]: Optional parameter. Square brackets do not appear on the command line.

1.2. AT Command Syntax

Table 1: AT Command Syntax

Test Command	**AT+<cmd>=?**	Check possible sub-parameter values
Read Command	**AT+<cmd>?**	Check current sub-parameter values
Write Command	**AT+<cmd>=p1[,p2[,p3[.....]]]**	Write command
Execution Command	**AT+<cmd>**	Execution command

Multiple commands can be placed on a single line using a semicolon (";") between commands. Only the first command should have AT prefix. Commands can be in upper case or lower case.

When entering AT commands spaces are ignored except in the following cases:

- Within quoted strings, where they are preserved;
- Within an unquoted string or numeric parameter;
- Within an IP address;
- Within the AT command name up to and including a '=', '?' or '=?'.

They can be used to make the input more human-readable. On input, at least a carriage return is required. A newline character is ignored so it is permissible to use carriage return/line feed pairs on the input.

If no command is entered after the AT token, **OK** will be returned. If an invalid command is entered, **ERROR** will be returned.

Optional parameters, unless explicitly stated, need to be provided up to the last parameter being entered..

1.3. AT Command Responses

When the AT command processor has finished processing a line, it will output **OK**, **ERROR** or **+CME ERROR:<err>** to indicate that it is ready to accept a new command. Solicited informational responses are sent before the final **OK**, **ERROR** or **+CME ERROR:<err>**.

Responses will be of the format below:

<CR><LF>+CMD1:<parameters><CR><LF>
<CR><LF>OK<CR><LF>

Or

<CR><LF><parameters><CR><LF>
<CR><LF>OK<CR><LF>

2 CoAP AT Commands

2.1. AT+QCOAPCREATE Create CoAP Context

The command is used to create a CoAP context. **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPCRETAE Create CoAP Context	
Test Command **AT+QCOAPCREATE=?**	Response **+QCOAPCREATE:** (the range of supported values of **<local_port>**) **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Write Command **AT+QCOAPCREATE=<local_port>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<local_port>	The CoAP context local port. Range: 1-65535.

Example

```
AT+QCOAPCREATE=?                    //Check possible parameter values
+QCOAPCREATE: <1-65535>
```

```
OK

AT+QCOAPCREATE=56830                    //Create a CoAP context.
OK
```

2.2. AT+QCOAPDEL Delete CoAP Context

The command is used to delete the CoAP context. **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPDEL Delete CoAP Context	
Execution Command **AT+QCOAPDEL**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Example

```
AT+QCOAPCREATE=56830                    //Create a CoAP context.
OK

AT+QCOAPDEL                             //Delete the CoAP context.
OK
```

2.3. AT+QCOAPADDRES Add CoAP Client Resources

The command is used to configure the CoAP resources. **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPADDRES Add CoAP Client Resources	
Test Command **AT+QCOAPADDRES=?**	Response **+QCOAPADDRES:** (the range of supported values of **<length>**),**"<resource>"**

	OK If there is any error: **ERROR** Or **+CME ERROR:<err>**
Write Command **AT+QCOAPADDRES=<length>,"<resource>"**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<length>	The CoAP client resources. Range: 1-50. Unit: character.
<resource>	The resource name.

Example

```
AT+QCOAPADDRES=4,"/t/d"        //Add the CoAP resource, and the value is “/t/d”.
OK
```

2.4. AT+QCOAPHEAD Configure CoAP Head

The command is used to configure CoAP head value (consisting of message ID and token value). **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPHEAD Configure CoAP Head	
Test Command **AT+QCOAPHEAD=?**	Response **+QCOAPHEAD: <mode>[,[<msgid>][,<tkl>,<token>]]** **OK** If there is any error: **ERROR**

	Or **+CME ERROR:<err>**
Write Command **AT+QCOAPHEAD=<mode>[,[<msgid>][,<tkl>,<token>]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<mode>	The message ID and token parameter. Range: 1-5. 1 Generate message ID and token values randomly. 2 Generate message ID randomly; configure token values. 3 Configure message ID randomly only; token value is not needed. 4 Configure message ID; generate token values randomly. 5 Configure message ID and token values.
<msgid>	The message ID, which only needs configuring when the **<mode>** value is 3, 4, 5. Range: 0-65535.
<tkl>	The length of token values, which only needs configuring when the <**mode**> value is 2 or 5. Range: 1-8.
<token>	The token values. Hexadecimal format. Only needs configuring when the **<mode>** value is 2 or 5.

NOTE

If this command is not set, the module will generate message ID randomly without need of token values.

Example

```
AT+QCOAPHEAD=1                          //Generate the message ID and token value randomly.
OK

AT+QCOAPHEAD=2,8,0102030405060708       //Generate message ID randomly, and configure the token value to 0102030405060708.
OK

AT+QCOAPHEAD=3,13940                    //Configure the message ID to 13940, and the token value is not needed.
OK
```

AT+QCOAPHEAD=4,13940 **OK**	//Configure the message ID to 13940, and generate the token value randomly.
AT+QCOAPHEAD=5,13940,4,02040608 **OK**	//Configure the message ID to 13940, and configure the token value to 02040608.

2.5. AT+QCOAPOPTION Configure CoAP Options

The command is used to configure the CoAP options. **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPOPTION Configure CoAP Options	
Test Command **AT+QCOAPOPTION=?**	Response **+QCOAPOPTION: <opt_count>,<opt_name>,"<opt_value>"[,...]** **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Write Command **AT+QCOAPOPTION=<opt_count>,<opt_name>,"<opt_value>"[,...]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<opt_count>	The option count. Range: 1-12.
<opt_name>	The option name. Refer to the *RFC 7252*. 1 If-Match 3 Uri-Host

	4	ETag
	5	If-None-Match
	6	Observe
	7	Uri-Port
	8	Location-Path
	11	Uri-Path
	12	Content-Format
	14	Max-Age
	15	Uri-Query
	17	Accept
	20	Location-Query
	23	Block2
	27	Block1
	28	SIZE
	35	Proxy-Uri
	39	Proxy-Scheme
	60	Size1
<opt_value>	String type. The option value. The length of value string: 1-180 byte(s).	
	If the **<opt_name>** is 12 or 17, the **<opt_value>** should be the value below:	
	0	Text-plain
	40	Application/link-format
	41	Application/xml
	42	Application/octet-stream
	47	Application/exi
	50	Application/json

Example

```
AT+QCOAPOPTION=1,11,"rd"                    //Configure the CoAP option 11 (Uri-Path) and the value is “rd”.

OK

AT+QCOAPOPTION=2,11,"rd",15,"ep=86370303"   //Configure the CoAP option 11 (Uri-Path), the value is “rd”, and configure the option 15 (Uri-Query), the value is “ep=86370303”.

OK

AT+QCOAPOPTION=2,11,".well-know",11,”core"  //Configure the CoAP option 11 (Uri-Path), the values are “.well-know” and “core”.

OK
```

2.6. AT+QCOAPSEND Send Data

This command is used to send data to the CoAP server. After sending CON data, the sending result will be automatically notified to the terminal. The status of the CON data that has been sent can also be queried by the terminal with the command **AT+QCOAPDATASTATUS?**. **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPSEND Send Data	
Test Command **AT+QCOAPSEND=?**	Response **+QCOAPSEND: <type>,<method/rspcode>,<ip_addr>,<port>** **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Write Command **AT+QCOAPSEND=<type>,<method/rspcode>,<ip_addr>,<port>** After "**>**" is responded, input the data to be sent. Tap "**CTRL+Z**" to send, and tap "**ESC**" to cancel the operation.	Response **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<type>	The message type of CoAP Protocol. Range: 0-3. Refer to the *RFC 7252*. 0 CON 1 NON 2 ACK 3 RST
<method>	The method of CoAP Protocol. Refer to the *RFC 7252*. 1 GET 2 POST 3 PUT 4 DELETE
<rspcode>	The response code of CoAP Protocol. Refer to the *RFC 7252*. 0 Empty Message 201 2.01, Created

	202	2.02, Deleted
	203	2.03, Valid
	204	2.04, Changed
	205	2.05, Content
	400	4.00, Bad Request
	401	4.01, Unauthorized
	402	4.02, Bad Option
	403	4.03, Forbidden
	404	4.04, Not Found
	405	4.05, Method Not Allowed
	406	4.06, Not Acceptable
	412	4.12, Precondition Failed
	413	4.13, Request Entity Too Large
	415	4.15, Unsupported Content-Format
	500	5.00, Internal Server Error
	501	5.01, Not Implemented
	502	5.02, Bad Gateway
	503	5.03, Service Unavailable
	504	5.04, Gateway Timeout
	505	5.05, Proxying Not Supported
<ip_addr>	The CoAP server address.	
<port>	The CoAP server port.	

Example

```
AT+QCOAPSEND=1,1,139.196.41.136,5683      //Send GET request with NON type to the CoAP server.
                                          //After receiving ">", input data and then send it. The maximum length of the data is 1024 bytes and the data that beyond 1024 bytes will be omitted. After inputting the data, tap "Ctrl+Z" to send.
OK

AT+QCOAPSEND=0,2,139.196.41.136,5683      //Send POST request in CON type to the CoAP server.
>0102
OK

//Received response URC. The message ID is 61440, the data length is 25 bytes, and the response code is 2.05.
+QCOAPURC: "rsp" 2.05,61440,25,4E692048616F2066726F6D20436F41502E4E45542052464320
```

NOTES

1. The max data length is 1024 bytes.
2. If CON data is to be sent, the status (not being sent/sent, waiting for the response of IoT

platform/sending failed/timeout/success/got reset message) of CON data sending must be acquired before sending the next CON or NON data through **AT+QCOAPDATASTATUS**.

2.7. AT+QCOAPDATASTATUS Query CON Messages Delivery Status

This command queries the status of the CON data sent to CoAP server. **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPDATASTATUS	**Query CON Messages Delivery Status**
Read Command **AT+QCOAPDATASTATUS?**	Response **+QCOAPDATASTATUS: <status>** **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<status>	Status of the sent CON data 0 Not being sent 1 Sent, waiting for the response of IoT platform 2 Sending failed 3 Timeout 4 Success 5 Got reset message

Example

```
AT+QCOAPSEND=0,1,139.196.41.136,5683          //Send GET request in CON type to CoAP server.
>                                             //After receiving ">", input data and then send it.
OK

AT+QCOAPDATASTATUS?
+QCOAPDATASTATUS:1                            //Sent, waiting for the response of CoAP server.

OK
```

```
//Received response URC. The message ID is 61440, the data length is 29 bytes, and the response code is 2.05.
+QCOAPURC: "rsp",2.05,61440,25, 4E692048616F2066726F6D20436F41502E4E455420524643

AT+QCOAPDATASTATUS?
+QLWDATASTATUS:4                    //Success.

OK
```

NOTE

This command only queries the status of the CON data that has been sent.

2.8. AT+QCOAPCFG CoAP Configuration Command

The command is used to configure the CoAP function parameters. **ERROR** or **+CME ERROR:<err>** will be responded if there is any error. Please refer to ***Chapter 4*** for possible **<err>** codes.

AT+QCOAPCFG CoAP Configuration Command	
Test Command **AT+QCOAPCFG=?**	Response **OK**
Read Command **AT+QCOAPCFG?**	Response **OK**
Write Command **AT+QCOAPCFG="Showra"[,<Showra>]**	Response If the parameter **<Showra>** is present, configure whether to display the IP address and port of sender: **OK** If the parameter **<Showra>** is omitted, query whether the IP address and port of sender is displayed: **+QCOAPCFG: "Showra",<Showra>** **OK** If error is related to ME functionality: **ERROR** Or **+CME ERROR:<err>**
Write Command **AT+QCOAPCFG="Showrspopt"[,<Sh**	Response If the parameter **<Showrspopt>** is present, configure whether

owrspopt>]	to display the CoAP option of sender: **OK** If the parameter **<Showrspopt>** is omitted, query whether the CoAP option of sender is displayed: **+QCOAPCFG: "Showrspopt",<Showrspopt>** **OK** If error is related to ME functionality: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<Showra>	Configure whether or not to display the IP address and port of sender. 0 Do not display the IP address and port. 1 Display the IP address and port in the URC; the format is as follows: **+QCOAPURC: "rsp",[<ip_addr>,<port>],<type>,<rspcode>,<msgid>[,<length>,<data>]**
<Showrspopt>	Configure whether or not to display the CoAP option of sender. 0 Do not display the CoAP option. 1 Display the CoAP option in the URC; the format is as follows: **+QCOAPURC: "rsp",<type>,<rspcode>,<msgid>[,<opt_count>,<opt_name>,"<opt_value>"[,...]][,<length>,<data>]**.

NOTES

1. Considering that the URC will be returned timely after the data is sent, it is recommended to configure with **AT+QCOAPCFG** before **AT+QCOAPSEND**.
2. **<Showra>** and **<Showrspopt>** will not be saved in the flash, thus, the configuration will be reset to default value (0) after powering down.
3. Please refer to ***Chapter 3*** for details about URCs.

Example

```
AT+QCOAPCFG?
OK

AT+QCOAPCFG="Showra",1          //Configure to display the IP address and port in the URC.
OK
```

3 CoAP Related URCs

This chapter gives CoAP related URCs and their descriptions.

Table 2: CoAP Related URCs

Index	URC	Description
[1]	**+QCOAPURC: "rsp"[<ip_addr>,<port>,]<type>,<rspcode>,<msgid>[,<opt_count>,<opt_name>,"<opt_value>"[,…]][,<length>,<data>]**	When sending the CON message, the CoAP server will respond ACK or RST. If the module receives such response, this URC will be reported.
[2]	**+QCOAPURC: "req",[<ip_addr>,<port>,]<type>,<method>,<msgid>,<mode>[,<tkl>,<token>][,<opt_name>,"<opt_value>"[,...]][,<length>,<data>]**	When the CoAP server sends a request, if the module receives such request, this URC will be reported.

3.1. Description of Response URC

The module reports the CoAP response event to the device.

Notify the Device of the Response Data from the Server	
URC Format: **+QCOAPURC: "rsp",[<ip_addr>,<port>,]<type>,<rspcode>,<msgid>[,<opt_count>,<opt_name>,"<opt_value>"[,...]][,<length>,<data>]**	Notify the response from CoAP server.

Parameter

<ip_addr>	The CoAP server IP address, which will be present with setting **AT+QCOAPCFG="Showra",1**.
<port>	The CoAP server port, which will be present with setting **AT+QCOAPCFG="Showra",1**.
<type>	The message type of CoAP Protocol. Range: 0-3. Refer to the *RFC 7252*.

	0 CON 1 NON 2 ACK 3 RST
<rspcode>	The response code of CoAP Protocol. Refer to the *RFC 7252*. 0 Empty Message 2.01 Created 2.02 Deleted 2.03 Valid 2.04 Changed 2.05 Content 4.00 Bad Request 4.01 Unauthorized 4.02 Bad Option 4.03 Forbidden 4.04 Not Found 4.05 Method Not Allowed 4.06 Not Acceptable 4.12 Precondition Failed 4.13 Request Entity Too Large 4.15 Unsupported Content-Format 5.00 Internal Server Error 5.01 Not Implemented 5.02 Bad Gateway 5.03 Service Unavailable 5.04 Gateway Timeout 5.05 Proxying Not Supported
<msgid>	The CoAP message ID.
<opt_count>	The count of option, which will be present with setting. **AT+QCOAPCFG="Showrspopt",1**
<opt_name>	The option name, which will be present with setting. **AT+QCOAPCFG="Showrspopt",1**.
<opt_value>	The option value, which will be present with setting. **AT+QCOAPCFG="Showrspopt",1**.
<length>	The data length. The max length is 512. Unit: byte.
<data>	If the **<opt_name>** is 12, and the **<opt_value>** is 0 “text-plain”, 41 “application/xml” or 50 “application/json”, the **<data>** format will be text string, otherwise, the format will be hexadecimal.

3.2. Description of Request URC

Notify the TE to Respond the Request from CoAP Server	
URC Format: **+QCOAPURC: "req",[<ip_addr>,<port>,]<type>,<method>,<msgid>,<mode>[,<tkl>,<token>][,<opt_name>,"<opt_value>"[,...]][,<length>,<data>]**	Notify the TE to respond the request from CoAP server.

Parameter

<ip_addr>	The CoAP server IP address, which will be present with setting **AT+QCOAPCFG="Showra",1**.
<port>	The CoAP server port, which will be present with setting **AT+QCOAPCFG="Showra",1**.
<type>	The message type of CoAP Protocol. Range: 0-3. Refer to the *RFC 7252*. 0 CON 1 NON 2 ACK 3 RST
<method>	The method of CoAP Protocol. Refer to the *RFC 7252*. 1 GET 2 POST 3 PUT 4 DELETE
<msgid>	The CoAP message ID.
<mode>	Indicates the existence of token, option, and data. Hexadecimal format. Bit 0: The existence of token. Bit 1-6: The count of option. Bit 7: The existence of data.
<tkl>	The token value length.
<token>	The token value. Hexadecimal format.
<opt_name>	The option name.
<opt_value>	The option value.
<length>	The data length. The max length is 1024. Unit: byte.
<data>	If the **<opt_name>** is 12, and the **<opt_value>** is 0 “text-plain”, 41 “application/xml” or 50 “application/json”, the data format will be text string, otherwise, the format will be hexadecimal.

4 Summary of Error Codes

This chapter summarizes the error codes related to BC95-G and BC68 modules.

The error codes listed in the following two tables are compliant with the 3GPP specifications. Please refer to *3GPP TS 27.007 V13.5.0*, sub-clause 9.2 for all possible **<err>** values.

Table 3: General Errors (27.007)

Code of <err>	Description
3	Operation not allowed
4	Operation not supported
23	Memory failure
30	No network service
50	Incorrect parameters
51	Command implemented but currently disabled
52	Command aborted by user
159	Uplink busy/flow control

Table 4: General Errors (27.005)

Code of <err>	Description
300	ME failure
301	SMS service of ME reserved
302	Operation not allowed
303	Operation not supported
304	Invalid PDU mode parameter

305	Invalid text mode parameter
310	USIM not inserted
311	USIM PIN required
312	PH-USIM PIN required
313	USIM failure
314	USIM busy
315	USIM wrong
316	USIM PUK required
317	USIM PIN2 required
318	USIM PUK2 required
320	Memory failure
321	Invalid memory index
322	Memory full
330	SMSC address unknown
331	No network service
332	Network timeout
340	No +CNMA acknowledgement expected
500	Unknown error

NOTE

AT+CMEE=<n> command disables (**<n>**=0) or enables (**<n>**=1) the use of final result code "**+CME ERROR:<err>**". When **<n>**=1, a limited set of error codes will be returned. Please refer to *Quectel_BC95-G&BC68_AT_Commands_Manual* for details of this command.

5 Examples

5.1. Send CoAP Request to the CoAP Server

```
AT+CGATT?                                   //Query the PS service attach status.
+CGATT: 1                                   //Attached to the PS service.
OK

AT+QCOAPCREATE=56830                        //Create the CoAP context.
OK

AT+QCOAPSEND=0,1,5.39.83.206,5683           //Send GET request in CON type to the server.
OK

AT+QCOAPDATASTATUS?                         //Query the data sending status.
+QCOAPDATASTATUS: 1                         //Sent, waiting response from IoT platform.

OK

//Received response URC. The response code is 2.05, the message ID is 6685, and the data length is 25 bytes. The data is in hexadecimal format.
+QCOAPURC: "rsp",2,2.05,6685,25,4E692048616F2066726F6D20436F41502E4E455420524643

AT+QCOAPDATASTATUS?                         //Query the data sending status.
+QCOAPDATASTATUS: 4                         //Success.

OK

AT+QCOAPHEAD=3,5566                         //Configure the message ID to 5566 and the token is not needed.

OK

AT+QCOAPSEND=0,1,139.196.187.107,5683       //Send GET request in CON type to the server.
>                                           //Type "Ctrl+Z", payload is none.
OK

//Received response URC. The response code is 2.05, the message ID is 5566 (the message ID is the
```

same as the set ID), and the data length is 25 bytes. The data is in hexadecimal format.
+QCOAPURC: "rsp",2,2.05,5566,29,4E692048616F2066726F6D20436F41502E4E455420524643

AT+QCOAPOPTION=1,11,"rd" //Configure the CoAP option, the Uri-Path is “/rd”.
OK

AT+QCOAPSEND=0,1,139.196.187.107,5683 //Send GET request in CON type to the server.
> //Type “**Ctrl+Z**”, payload is none.
OK

//Received response URC. The response code is 2.05, the message ID is 61441, and the data length is 17 bytes. The data is in hexadecimal format.
+QCOAPURC:"req",2,2.05,61441,17,323031382F312F392031313A31313A3139

AT+QCOAPHEAD=5,6677,4,30323436 //Configure the message ID to 6677 and the token value to 30323436.
OK

AT+QCOAPOPTION=1,11,"hello" //Configure the CoAP option, the Uri-Path is “/hello”.
OK

AT+QCOAPSEND=0,1,139.196.187.107,5683 //Send GET request in CON type to the server.
> //Type “**Ctrl+Z**”, payload is none.
OK

//Received response URC. The response code is 2.05, the message ID is 6677, and the data length is 22 bytes. The data is in hexadecimal format.
+QCOAPURC:"rsp",2,2.05,6677,22,68656C6C6F20776F72642066726F6D20736572766572

//Set to display the IP address and port. When receiving the URC, the IP address and port will be shown.
AT+QCOAPCFG="Showra",1
OK

AT+QCOAPADDRES=4,"/t/d" //Add the CoAP resource, the value is “/t/d”.
OK

//Configure the value of CoAP option 11 (Uri-Path) to “t” and “r”, the value of CoAP option 15 (Uri-Query) to “ep=863703030822519”.
AT+QCOAPOPTION=3,11,"t",11,"r",15,"ep=863703030822519"
OK

AT+QCOAPSEND=0,2, 139.196.187.107,5683 // Send POST request in CON type to the server.
> //Type “**Ctrl+Z**”, payload is none.
OK

//Received response URC from the server 180.101.147.115:5683. The type is ACK, the response code is 2.04, and the message ID is 6677.
+QCOAPURC:"rsp",180.101.147.115,5683,2,2.04,2802

//Received request URC from the server 180.101.147.115:5683. The message type is CON, the method is GET, the message ID is 50670, and the **<mode>** is 09 (meaning the option count is 4 and the token and option exists).
+QCOAPURC:"req",180.101.147.115,5683,0,1,50670,09,8,1A84BFE989C01C08,6,"0",7,"54321",11,"t",11,"d"

//When receiving the request data, the client should respond it.
AT+QCOAPHEAD=5,50670,8,1A84BFE989C01C08 //Configure the message ID and token values.
OK

AT+QCOAPSEND=1,205,139.196.187.107,5683 //Send message with response code of 2.05 in NON type to the server.
> //Type "**Ctrl+Z**", payload is none
OK

AT+QCOAPDEL //Delete the CoAP context.
OK

5.2. Register to China Telecom IoT Platform

AT+QCOAPCREATE=56830 //Create a CoAP context
OK

AT+QCOAPHEAD=1 //Generate the message ID and the token value randomly.
OK

//Configure the value of CoAP option 11 (Uri-Path) to "rd", the value of CoAP option 12 to "42" and the value of CoAP option 15 (Uri-Query) to "lwm2m=1.0" and "ep=867725030002525;460041850403693"…
AT+QCOAPOPTION=6,11,"rd",12,"42",15,"lwm2m=1.0",15,"ep=867725030002525;460041850403693",15,"b=U",15,"lt=86400"
OK

AT+QCOAPADDRES=6,"/4/0/2" //Add the CoAP resource, the value is "/4/0/2"
OK

AT+QCOAPADDRES=6,"/4/0/3" //Add the CoAP resource, the value is "/4/0/3"
OK

```
AT+QCOAPADDRES=6,"/4/0/8"      //Add the CoAP resource, the value is “/4/0/8”
OK

AT+QCOAPADDRES=6,"/3/0/9"      //Add the CoAP resource, the value is “/3/0/9”
OK

AT+QCOAPADDRES=6,"/3/0/7"      //Add the CoAP resource, the value is “/3/0/7”
OK

AT+QCOAPADDRES=7,"/19/0/0"     //Add the CoAP resource, the value is “/19/0/0”
OK

AT+QCOAPADDRES=7,"/19/1/0"     //Add the CoAP resource, the value is “/19/1/0”
OK

//Send register message to China Telecom IoT Platform
AT+QCOAPSEND=0,2,180.101.147.115,5683
>3c2f3e3b72743d226f6d612e6c776d326d222c3c2f312f303e2c3c2f332f303e2c3c2f342f303e2c3c2f352f303e2c3c2f32302f303e2c3c2f31392f303e
OK

//Received response URC. The message type is ACK, the response code is 2.01, and the message ID is 55018.
+QCOAPURC: "rsp",2,2.01,55018

//Received request URC. The message type is CON, the method is GET, the message ID is 62084, and the <mode> is 09 (meaning the option count is 4 and the token and option exists).
+QCOAPURC: "req",0,1,62084,09,8,1A98EB114F501C07,6,"0",11,"19",11,"0",11,"0"

//When receiving the request data, the client should respond it.
//Configure the CoAP token the same as the request message
AT+QCOAPHEAD=5,62084,8,1A98EB114F501C07   //Configure the message ID to 62084 and the token value to 1A98EB114F501C07.
OK

AT+QCOAPSEND=2,205,180.101.147.115, 5683   //Send message with response code 2.05 in ACK type to the server.
>                                          //Type “Ctrl+Z”, payload is none.
OK

AT+QCOAPDEL                                //Delete the CoAP context.
OK
```

5.3. Example of using the configure commands

AT+QCOAPCREATE=56830 //Create a CoAP context
OK

AT+QCOAPCFG="Showrspopt",1 //Configure to show the CoAP option of sender
OK

//Configure the value of CoAP option 11 (Uri-Path) to “rd”, the value of CoAP option 12 to “40” and the value of CoAP option 15 (Uri-Query) to “ep= I@#@G7pjZBAQUoVav0yXQcQ6v4Ad1nFGXcKhnTWYJt3AE0TpZSj3xDrYMOm6NmohixF@#@M100000089”, “b=U” and “lt=2700”.
AT+QCOAPOPTION=5,11,"rd",12,"40",15,"ep=I@#@G7pjZBAQUoVav0yXQcQ6v4Ad1nFGXcKhnTWYJt3AE0TpZSj3xDrYMOm6NmohixF@#@M100000089",15,"b=U",15,"lt=2700"
OK

AT+QCOAPSEND=0,2,220.180.239.212,8063
>3c2f3e3b72743d226f6d612e6c776d326d222c3c2f30
OK

AT+QCOAPDATASTATUS? //Query the data sending status.
+QCOAPDATASTATUS: 4 //Success

OK

//Received response URC. The values of option 8 (location-path) are “rd” and “BEHJvP29Lb”.
+QCOAPURC: "rsp",2,2.01,13357,2,8,"rd",8,"BEHJvP29Lb"

//Set to display the IP address and port of sender. When receiving the URC, the IP address and port will be shown.
AT+QCOAPCFG="Showra",1
OK

AT+QCOAPOPTION=5,11,"rd",12,"40",15,"ep=I@#@G7pjZBAQUoVav0yXQcQ6v4Ad1nFGXcKhnTWYJt3AE0TpZSj3xDrYMOm6NmohixF@#@M100000089",15,"b=U",15,"lt=2700"
OK

AT+QCOAPSEND=0,2,220.180.239.212,8063 //Send POST request in CON type to the server.
>3c2f3e3b72743d226f6d612e6c776d326d222c3c2f30 //Type the payload and **“Ctrl+Z”** to send.
OK

//Received response URC. The values of option 8 (location-path) are “rd” and “dTuzhqieJe”, the CoAP server IP address is 220.180.239.212 and the port is 8063.

```
+QCOAPURC: "rsp",220.180.239.212,8063,2,2.01,13358,2,8,"rd",8,"dTuzhqieJe"

//Set not to display the CoAP option. When receiving the URC, the CoAP option will not be shown.
AT+QCOAPCFG="Showrspopt",0
OK

//Set not to display the IP address and port. When receiving the URC, the IP address and port will not be shown.
AT+QCOAPCFG="Showra",0
OK

//Configure the value of CoAP option 11 (Uri-Path) to “hello” and the value of CoAP option 15 (Uri-Query) to “name=aaaa”.
AT+QCOAPOPTION=2,11,"hello",15,"name=aaaa"
OK

AT+QCOAPSEND=0,1,220.180.239.212,8098    //Send GET request in CON type to the server.
>                                        //Type “Ctrl+Z”, payload is none.
OK

//Received response URC. The content format is “0”, and the data format is text string.
+QCOAPURC: "rsp",2,2.05,54750,1,12,"0",41, hi name=aaaa,this is quectel IoT platform

AT+QCOAPDEL                              //Delete the CoAP context.
OK
```