

# **BG95&BG77&BG600L Series**

## **DFOTA Application Note**

**LPWA Module Series**

Version: 1.2

Date: 2022-04-01

Status: Released



At Quectel, our aim is to provide timely and comprehensive services to our customers. If you require any assistance, please contact our headquarters:

**Quectel Wireless Solutions Co., Ltd.**

Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai 200233, China

Tel: +86 21 5108 6236

Email: [info@quectel.com](mailto:info@quectel.com)

**Or our local offices. 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 us at: [support@quectel.com](mailto:support@quectel.com).

## Legal Notices

We offer information as a service to you. The provided information is based on your requirements and we make every effort to ensure its quality. You agree that you are responsible for using independent analysis and evaluation in designing intended products, and we provide reference designs for illustrative purposes only. Before using any hardware, software or service guided by this document, please read this notice carefully. Even though we employ commercially reasonable efforts to provide the best possible experience, you hereby acknowledge and agree that this document and related services hereunder are provided to you on an “as available” basis. We may revise or restate this document from time to time at our sole discretion without any prior notice to you.

## Use and Disclosure Restrictions

### License Agreements

Documents and information provided by us shall be kept confidential, unless specific permission is granted. They shall not be accessed or used for any purpose except as expressly provided herein.

### Copyright

Our and third-party products hereunder may contain copyrighted material. Such copyrighted material shall not be copied, reproduced, distributed, merged, published, translated, or modified without prior written consent. We and the third party have exclusive rights over copyrighted material. No license shall be granted or conveyed under any patents, copyrights, trademarks, or service mark rights. To avoid ambiguities, purchasing in any form cannot be deemed as granting a license other than the normal non-exclusive, royalty-free license to use the material. We reserve the right to take legal action for noncompliance with abovementioned requirements, unauthorized use, or other illegal or malicious use of the material.

## Trademarks

Except as otherwise set forth herein, nothing in this document shall be construed as conferring any rights to use any trademark, trade name or name, abbreviation, or counterfeit product thereof owned by Quectel or any third party in advertising, publicity, or other aspects.

## Third-Party Rights

This document may refer to hardware, software and/or documentation owned by one or more third parties ("third-party materials"). Use of such third-party materials shall be governed by all restrictions and obligations applicable thereto.

We make no warranty or representation, either express or implied, regarding the third-party materials, including but not limited to any implied or statutory, warranties of merchantability or fitness for a particular purpose, quiet enjoyment, system integration, information accuracy, and non-infringement of any third-party intellectual property rights with regard to the licensed technology or use thereof. Nothing herein constitutes a representation or warranty by us to either develop, enhance, modify, distribute, market, sell, offer for sale, or otherwise maintain production of any our products or any other hardware, software, device, tool, information, or product. We moreover disclaim any and all warranties arising from the course of dealing or usage of trade.

## Privacy Policy

To implement module functionality, certain device data are uploaded to Quectel's or third-party's servers, including carriers, chipset suppliers or customer-designated servers. Quectel, strictly abiding by the relevant laws and regulations, shall retain, use, disclose or otherwise process relevant data for the purpose of performing the service only or as permitted by applicable laws. Before data interaction with third parties, please be informed of their privacy and data security policy.

## Disclaimer

- a) We acknowledge no liability for any injury or damage arising from the reliance upon the information.
- b) We shall bear no liability resulting from any inaccuracies or omissions, or from the use of the information contained herein.
- c) While we have made every effort to ensure that the functions and features under development are free from errors, it is possible that they could contain errors, inaccuracies, and omissions. Unless otherwise provided by valid agreement, we make no warranties of any kind, either implied or express, and exclude all liability for any loss or damage suffered in connection with the use of features and functions under development, to the maximum extent permitted by law, regardless of whether such loss or damage may have been foreseeable.
- d) We are not responsible for the accessibility, safety, accuracy, availability, legality, or completeness of information, advertising, commercial offers, products, services, and materials on third-party websites and third-party resources.

**Copyright © Quectel Wireless Solutions Co., Ltd. 2022. All rights reserved.**

# About the Document

## Revision History

| Version | Date       | Author                                  | Description   |
|---------|------------|---|---|
| 1.0     | 2020-01-15 | Matt YE/<br>Lane HAO                    | Initial   |
| 1.1     | 2020-07-29 | Matt YE                                 | <ol style="list-style-type: none"> <li>Added the applicable module BG600L-M3.</li> <li>Added supported baud rates for reporting of DFOTA URCs and develop the main UART baud rate to follow the configuration by AT+IPR.</li> <li>Updated the range of the &lt;file_size&gt; parameter for delta package.</li> </ol>  |
| 1.2     | 2022-04-01 | Billie XING/<br>Matt YE/<br>Adonis CHEN | <ol style="list-style-type: none"> <li>Added parameters &lt;breakpoint&gt; and &lt;request_size&gt; in AT+QFOTADL=&lt;HTTP_URL&gt;.</li> <li>Added AT+QFOTADL=&lt;FTP_URL&gt;.</li> <li>Updated the note about &lt;file_size&gt; in AT+QFOTADL="COM:",&lt;file_size&gt;,&lt;timeout&gt;.</li> <li>Added &lt;para3&gt; for AT+QFOTADL=&lt;para1&gt;,&lt;para2&gt;,&lt;para3&gt;.</li> <li>Added AT+QFOTADL=&lt;file_name&gt;,&lt;type&gt;.</li> <li>Added AT+QCFGEXT="fota_http_header".</li> <li>Added AT+QCFGEXT="fota_wd_gpio".</li> <li>Added AT+QCFGEXT="sni".</li> <li>Added some examples to illustrate the use of new commands.</li> <li>Completed the list of error codes.</li> </ol> |

## Contents

|   |          |
|---|----------|
| About the Document .....  | 3        |
| Contents .....  | 4        |
| Table Index .....   | 6        |
| Figure Index .....  | 7        |
| <b>1 Introduction .....</b>   | <b>8</b> |
| <b>2 Trigger DFOTA by AT Commands .....</b>   | <b>9</b> |
| 2.1. Firmware Update Procedure .....  | 9        |
| 2.1.1. Get a Delta Firmware Package .....   | 10       |
| 2.1.2. Put Delta Package on the HTTP(S)/CoAP(S)/FTP(S) Server .....                                 | 10       |
| 2.1.3. Check APN Setting and Network Status .....   | 10       |
| 2.1.4. Execute AT Command to Update the Firmware .....  | 11       |
| 2.2. DFOTA Related AT Commands .....  | 11       |
| 2.2.1. AT Command Introduction .....  | 11       |
| 2.2.1.1. Definitions .....  | 11       |
| 2.2.1.2. AT Command Syntax .....  | 11       |
| 2.2.2. Declaration of AT Command Examples .....   | 12       |
| 2.2.3. Description of AT Command Mode and Data Mode .....   | 12       |
| 2.2.4. AT+QFOTADL Update Firmware via DFOTA .....   | 12       |
| 2.2.4.1. AT+QFOTADL=<HTTP_URL> Update Firmware When Delta Package is Stored on HTTP(S) Server ..... | 13       |
| 2.2.4.2. AT+QFOTADL=<CoAP_URL> Update Firmware When Delta Package is Stored on CoAP(S) Server ..... | 15       |
| 2.2.4.3. AT+QFOTADL=<FTP_URL> Update Firmware When Delta Package is Stored on FTP(S) Server .....   | 16       |
| 2.2.4.4. AT+QFOTADL="COM:",<file_size>,<timeout> Upload a Delta Package to Flash .....              | 18       |
| 2.2.4.5. AT+QFOTADL=<para1>[,<para2>[,<para3>]] Extended DFOTA Configurations .....                 | 19       |
| 2.2.4.6. AT+QFOTADL=<file_name>,<type> Copy Delta Package to Flash from File System .....           | 22       |
| 2.2.5. AT+QCFGEXT="fota_apn" Configure IP Family and APN for DFOTA .....                            | 23       |
| 2.2.6. AT+QCFGEXT="fota_http_header" Configure HTTP(S) Header for DFOTA over HTTP(S) .....          | 24       |
| 2.2.7. AT+QCFGEXT="fota_wd_gpio" Configure Watchdog Pin and Feeding Interval During DFOTA .....     | 24       |
| 2.2.8. AT+QCFGEXT="sni" Configure Whether to Enable Server Name Indication for DFOTA .....          | 26       |
| 2.3. Example .....  | 26       |
| 2.3.1. Update Firmware from HTTP(S) Server by DFOTA .....   | 26       |
| 2.3.2. Update Firmware from FTP(S) Server by DFOTA .....  | 27       |
| 2.3.3. Copy Delta Package to NAND Flash from AP EUFS .....  | 28       |
| 2.3.4. Set Breakpoint Continuous Downloading and Single Request Size for HTTP(S) DFOTA .....        |          |

|   |           |
|---|-----------|
| Downloading.....  | 28        |
| 2.3.5. Configure HTTP(S) Header for DFOTA over HTTP(S).....         | 29        |
| 2.3.6. Set Watchdog Configuration for DFOTA.....                    | 29        |
| <b>3 Trigger DFOTA with LwM2M.....</b>                              | <b>31</b> |
| 3.1. Firmware Update Procedure.....                                 | 31        |
| 3.1.1. Download a Firmware Image.....                               | 32        |
| 3.1.2. Update the Firmware.....                                     | 33        |
| 3.2. URC of Firmware Update with LwM2M .....                        | 34        |
| 3.2.1. Download Progress URCs .....                                 | 34        |
| 3.2.1.1. In-Band Mode .....   | 34        |
| 3.2.1.2. Out-of-Band HTTP(S) Mode.....                              | 34        |
| 3.2.1.3. Out-of-Band CoAP(S) Mode.....                              | 35        |
| 3.2.2. Update Progress URCs.....                                    | 36        |
| 3.2.2.1. +QIND: "FOTA","START" Updating Started.....                | 36        |
| 3.2.2.2. +QIND: "FOTA","UPDATING",<percent> Updating Progress ..... | 36        |
| 3.2.2.3. +QIND: "FOTA","END",<err> Updating Finished.....           | 36        |
| <b>4 Summary of Error Codes .....</b>                               | <b>37</b> |
| <b>5 Appendix A References.....</b>                                 | <b>39</b> |

## Table Index

|  |    |
|--|----|
| Table 1: Types of AT Commands .....                            | 11 |
| Table 2: Summary of <HTTP_err>/<CoAP_err>/<FTP_err> Codes..... | 37 |
| Table 3: Summary of <update_err> Codes.....                    | 38 |
| Table 4: Summary of <err> Codes .....                          | 38 |
| Table 5: Related Documents .....                               | 39 |
| Table 6: Terms and Abbreviations .....                         | 39 |

## Figure Index

|   |    |
|---|----|
| Figure 1: DFOTA Procedure (Triggered by AT Command).....                            | 9  |
| Figure 2: LwM2M Firmware Image Download .....                                       | 32 |
| Figure 3: Call Flow of Package URI Observing and Firmware Downloading.....          | 33 |
| Figure 4: Call Flow of “Update” Resource Observing and Update Event Processing..... | 34 |



# 1 Introduction

Quectel BG95 series, BG77 and BG600L-M3 modules support DFOTA (Delta Firmware Upgrade Over-The-Air) feature, which allows users to update firmware wirelessly.

In DFOTA, a delta firmware package, which contains only differences between the current and the target update versions, is needed before firmware updating. Therefore, DFOTA is time-saving and can reduce unnecessary data transmission.

This document mainly introduces how to update the firmware of Quectel BG95 series, BG77 and BG600L-M3 modules via DFOTA, which can be triggered either by **AT+QFOTADL** or via LwM2M.

## NOTE

1. It is mandatory to ensure a stable supply voltage during DFOTA process.
2. If there is an external watchdog monitoring the module, the DAM cannot run during DFOTA process, and the watchdog cannot be fed in the DAM app. In this case, it is necessary to remove the watchdog temporarily or feed it through AP kernel (see **Chapter 2.3.6** for details) to prevent unexpected restart behaviors that may interrupt the DFOTA process or cause possible damages to the module.

## 2 Trigger DFOTA by AT Commands

### 2.1. Firmware Update Procedure

The following chart illustrates the DFOTA procedure triggered by the AT command, provided that the firmware package is stored on an HTTP(S)/CoAP(S)/FTP(S) server.

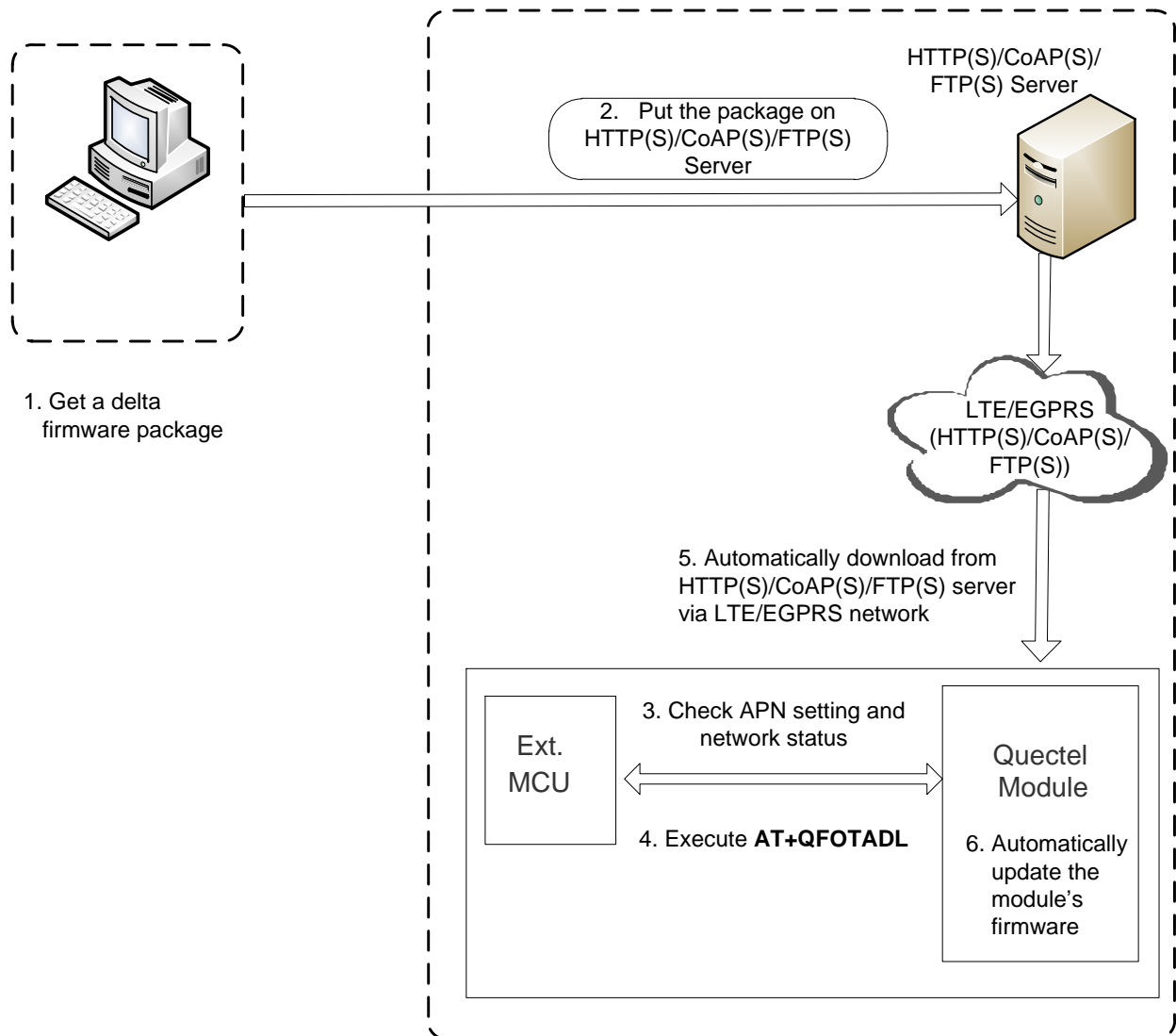


Figure 1: DFOTA Procedure (Triggered by AT Command)

As shown in the above figure, the following steps need to be performed to update the firmware when the firmware package is stored on an HTTP(S)/CoAP(S)/FTP(S) server:

- Step 1:** Get a delta firmware package from Quectel.
- Step 2:** Put the delta package on the HTTP(S)/CoAP(S)/FTP(S) server.
- Step 3:** Check APN setting and network status.
- Step 4:** Execute **AT+QFOTADL**. The package will be downloaded to the module from the HTTP(S)/CoAP(S)/FTP(S) server via LTE/EGPRS network (**Step 5** in the figure above). Then, the firmware of the module is updated automatically (**Step 6** in the figure above).

### 2.1.1. Get a Delta Firmware Package

Before updating, check the current firmware version with **AT+QGMR** and select a target firmware version, and then send the two firmware versions to Quectel or module supplier to get a delta package.

### 2.1.2. Put Delta Package on the HTTP(S)/CoAP(S)/FTP(S) Server

- Step 1:** Set up the HTTP(S)/CoAP(S)/FTP(S) server before using the DFOTA function (Quectel does not provide such servers.)
- Step 2:** Put the delta firmware package on the server and record its storage path.
- Step 3:** The module downloads the delta package using the URL path of the file uploaded on the server after executing the corresponding AT command.

### 2.1.3. Check APN Setting and Network Status

- Step 1:** Check APN setting: check whether the APN is set properly after putting the delta firmware package on the HTTP(S)/CoAP(S)/FTP(S) server. For instance, the APN for a Telstra (U)SIM card should be "Telstra.internet". The relevant AT commands are listed below:

- **AT+QCFGEXT="fota\_apn"**: Query DFOTA APN
- **AT+QCFGEXT="fota\_apn",0,"Telstra.internet"**: Set DFOTA APN

- Step 2:** Check network status: after the APN setting is confirmed, make sure that the data network is registered before firmware updating. The relevant AT commands are listed below:

- **AT+CSQ**: Query signal quality
- **AT+CEREG?**: Query network registration status (Optional: **AT+CGREG?**)
- **AT+COPS?**: Query the registered operator

For more details about the above commands, see **document [1]**.

### 2.1.4. Execute AT Command to Update the Firmware

After checking the APN setting and network status, execute **AT+QFOTADL** to download the delta firmware package to the module from the HTTP(S)/CoAP(S)/FTP(S) server via LTE/EGPRS network. Then, the firmware will be updated automatically. For an example, see **Chapter 2.3**.

## 2.2. DFOTA Related AT Commands

### 2.2.1. AT Command Introduction

#### 2.2.1.1. Definitions

- **<CR>** Carriage return character.
- **<LF>** Line feed character.
- **<...>** Parameter name. Angle brackets do not appear on the command line.
- **[...]** Optional parameter of a command or an optional part of TA information response. Square brackets do not appear on the command line. When an optional parameter is not given in a command, the new value equals its previous value or the default settings, unless otherwise specified.
- **Underline** Default setting of a parameter.

#### 2.2.1.2. AT Command Syntax

All command lines must start with **AT** or **at** and end with **<CR>**. Information responses and result codes always start and end with a carriage return character and a line feed character: **<CR><LF><response><CR><LF>**. In tables presenting commands and responses throughout this document, only the commands and responses are presented, and **<CR>** and **<LF>** are deliberately omitted.

**Table 1: Types of AT Commands**

| Command Type  | Syntax  | Description  |
|---------------|---|--|
| Test Command  | <b>AT+&lt;cmd&gt;=?</b>   | Test the existence of the corresponding command and return information about the type, value, or range of its parameter. |
| Read Command  | <b>AT+&lt;cmd&gt;?</b>  | Check the current parameter value of the corresponding command.  |
| Write Command | <b>AT+&lt;cmd&gt;=&lt;p1&gt;[,&lt;p2&gt;[,&lt;p3&gt;[...]]]</b> | Set user-definable parameter value.  |

Execution Command **AT+<cmd>**

Return a specific information parameter or perform a specific action.

### 2.2.2. Declaration of AT Command Examples

The AT command examples in this document are provided to help you learn about the use of the AT commands introduced herein. The examples, however, should not be taken as Quectel's recommendations or suggestions about how to design a program flow or what status to set the module into. Sometimes multiple examples may be provided for one AT command. However, this does not mean that there is a correlation among these examples, or that they should be executed in a given sequence.

### 2.2.3. Description of AT Command Mode and Data Mode

The COM port of BG95 series, BG77 and BG600L-M3 modules provide two working modes: AT command mode and data mode. In the AT command mode, the inputted data via COM port are treated as AT commands. In data mode, they are treated as data.

Inputting **+++** or pulling up MAIN\_DTR (**AT&D1** should be set first) can make the COM port exit data mode. To prevent **+++** from being misinterpreted as data, the following sequence should be followed:

1. Do not input any character for at least 1 second before and after inputting **+++**.
2. Input **+++** within 1 second, and wait until **OK** is returned.

After executing **AT+QFOTADL="COM:",<file\_size>,<timeout>**, the COM port enters the data mode. The port exits the data mode forcibly and the command is interrupted immediately if **+++** is inputted or if MAIN\_DTR is pulled up before the command response is returned.

### 2.2.4. AT+QFOTADL Update Firmware via DFOTA

#### AT+QFOTADL Update Firmware via DFOTA

|                     |           |
|---------------------|-----------|
| Test Command        | Response  |
| <b>AT+QFOTADL=?</b> | <b>OK</b> |

**AT+QFOTADL** initiates automatic module firmware updating via DFOTA. Once **AT+QFOTADL** is executed, the delta firmware package is automatically downloaded to the module. After downloading the delta package, the module automatically updates the firmware. The module reboots automatically if the firmware has been updated successfully, otherwise it returns an error message and exits DFOTA.

### 2.2.4.1. AT+QFOTADL=<HTTP\_URL> Update Firmware When Delta Package is Stored on HTTP(S)

#### Server

When the delta firmware package is stored on an HTTP(S) server, execute **AT+QFOTADL=<HTTP\_URL>[,<breakpoint>[,<request\_size>]]** to start automatic firmware updating via DFOTA. The module downloads the delta package from the HTTP(S) server over the air and updates the firmware automatically.

With **<breakpoint>** and **<request\_size>**, this command allows the module to download the entire delta package in separate segments. **<breakpoint>** specifies where to continue the download if a segmented download was implemented. When **<breakpoint>** is specified, the size of delta package to be downloaded can be configured with **<request\_size>**.

#### AT+QFOTADL=<HTTP\_URL> Update Firmware When Delta Package is Stored on HTTP(S) Server

Write Command

**AT+QFOTADL=<HTTP\_URL>[,<breakpoint>[,<request\_size>]]**

Response

If the optional parameters are omitted:

**OK**

**+QIND: "FOTA","HTTPSTART"**

**+QIND: "FOTA","DOWNLOADING",<percent>**

**+QIND: "FOTA","DOWNLOADING",<percent>**

...

**+QIND: "FOTA","HTTPEND",<HTTP\_err>**

**+QIND: "FOTA","RESETTING"**

**+QIND: "FOTA","START"**

**+QIND: "FOTA","UPDATING",<percent>**

**+QIND: "FOTA","UPDATING",<percent>**

...

**+QIND: "FOTA","END",<update\_err>**

If any of the optional parameters is specified, set the breakpoint position and request size:

**OK**

**+QIND: "FOTA","HTTPSTART"**

**+QIND: "FOTA","DOWNLOADING",<percent>**

**+QIND: "FOTA","DOWNLOADING",<percent>**

...

**+QIND: "FOTA","HTTPEND",<HTTP\_err>**

When **<HTTP\_err>=0** (which indicates the entire delta package has been downloaded successfully), the module

|                       |  |
|-----------------------|--|
|                       | <p>automatically reboots and starts firmware updating:</p> <p><b>+QIND: "FOTA","RESETTING"</b></p> <p><b>+QIND: "FOTA","START"</b></p> <p><b>+QIND: "FOTA","UPDATING",&lt;percent&gt;</b></p> <p><b>+QIND: "FOTA","UPDATING",&lt;percent&gt;</b></p> <p>...</p> <p><b>+QIND: "FOTA","END",&lt;update_err&gt;</b></p> <p>If there is any error:</p> <p><b>+CME ERROR: &lt;err&gt;</b></p> |
| Maximum Response Time | 300 ms   |
| Characteristics       | <p>The command takes effect immediately.</p> <p>The configuration will not be saved.</p>   |

## Parameter

|                   |   |
|-------------------|---|
| <HTTP_URL>        | String type. The maximum length is 700 bytes. It should be started with "HTTP://" or "HTTPS://". For example:<br><b>"HTTP://&lt;HTTP_server_URL&gt;:&lt;HTTP_port&gt;/&lt;HTTP_file_path&gt;"</b> .         |
| <breakpoint>      | Integer type. The offset location of the delta package. Default value: 0.<br>Unit: byte.  |
| <request_size>    | Integer type. Size of the delta package to be downloaded for this downloading request. It must be an integer multiple of 2048. Unit: byte.<br>The default value 0 indicates downloading all remained bytes. |
| <HTTP_server_URL> | String type. The IP address or domain name of the HTTP(S) server.   |
| <HTTP_port>       | Integer type. The port number of the HTTP(S) server. Default value: 80.<br>Range: 1–65535.  |
| <HTTP_file_path>  | String type. File path on HTTP(S) server.   |
| <HTTP_err>        | Integer type. HTTP(S) result code.<br>0 Updated successfully.<br>Any other value Errors. See <b>Chapter 4</b> for more details.   |
| <percent>         | Integer type. The download or update progress in percentage.  |
| <update_err>      | Integer type. Update result code.<br>0 Updated successfully.<br>Any other value Errors. See <b>Chapter 4</b> for more details.  |
| <err>             | Error codes. See <b>Chapter 4</b> for more details.   |

### NOTE

1. In the DFOTA mode, only the APP works normally and the modem is not loaded. In this period, the main UART cannot receive AT commands anymore, and can only report DFOTA URCs at the baud rate configured by **AT+IPR**. Supported baud rates for the reporting of URCs are 4800, 9600, 19200,

38400, 115200, 230400, 460800 and 921600 bps. The default baud rate 115200 bps is used if any unsupported baud rate is set by **AT+IPR** for the URC reporting. Additionally, the USB port cannot be used either.

2. After a smooth "**DOWNLOADING**" process, the module is reset to enter the DFOTA mode. When the "**UPDATING**" process is finished, the module restarts and start up in a normal mode.
3. If the module is powered down during "**UPDATING**" process, the module will automatically enter the DFOTA mode and continue the previous firmware update when it is powered up next time.
4. (1) If SSL certificates are needed while downloading the firmware package from an HTTPS server, they should be uploaded to /datatx in APP EFS, and must be renamed in accordance with the following rules:
  - 1) The root certificate is renamed *fota\_cacert.pem*;
  - 2) The client certificate is renamed *fota\_client\_cert.pem*;
  - 3) The private key is renamed *fota\_client\_key.pem*.
- (2) Certificate upload methods:
  - Quectel QEFS Explorer tool can be used to upload the certificates. For more details about the tool, refer to **document [2]**.
  - **AT+QFUPL** can also be used to upload the certificates. For more details about the command, see **document [3]**.

#### 2.2.4.2. **AT+QFOTADL=<CoAP\_URL>** Update Firmware When Delta Package is Stored on CoAP(S) Server

When the delta firmware package is stored on a CoAP(S) server, execute **AT+QFOTADL=<CoAP\_URL>** to start automatic firmware updating via DFOTA. The module downloads the delta package from the CoAP(S) server over the air and updates the firmware automatically.

#### **AT+QFOTADL=<CoAP\_URL>** Update Firmware When Delta Package is Stored on CoAP(S) Server

|   |   |
|---|---|
| Write Command<br><b>AT+QFOTADL=&lt;CoAP_URL&gt;</b> | Response<br><b>OK</b>   |
|   | <b>+QIND: "FOTA","COAPSTART"</b><br><b>+QIND: "FOTA","COAPEND",&lt;CoAP_err&gt;</b><br><b>+QIND: "FOTA","RESETTING"</b><br><b>+QIND: "FOTA","START"</b><br><b>+QIND: "FOTA","UPDATING",&lt;percent&gt;</b><br><b>+QIND: "FOTA","UPDATING",&lt;percent&gt;</b><br>...<br><b>+QIND: "FOTA","END",&lt;update_err&gt;</b> |
|   | If there is any error:<br><b>+CME ERROR: &lt;err&gt;</b>  |



|                       |                                       |
|-----------------------|---------------------------------------|
| Maximum Response Time | 300 ms                                |
| Characteristics       | The command takes effect immediately. |

## Parameter

|                                |   |
|--------------------------------|---|
| <b>&lt;CoAP_URL&gt;</b>        | String type. The maximum length is 700 bytes. It should be started with "CoAP://" or "CoAPS://", for example: " <b>CoAP://&lt;CoAP_server_URL&gt;:&lt; CoAP_port&gt;/&lt; CoAP_file_path&gt;</b> ". |
| <b>&lt;CoAP_server_URL&gt;</b> | String type. The IP address or domain name of the CoAP(S) server.   |
| <b>&lt;CoAP_port&gt;</b>       | Integer type. The port number of the CoAP(S) server. Default value: 80. Range: 1–65535.   |
| <b>&lt;CoAP_file_path&gt;</b>  | String type. File path on CoAP(S) server.   |
| <b>&lt;CoAP_err&gt;</b>        | Integer type. CoAP(S) result code.<br>0 Updated successfully.<br>Any other value Errors. See <b>Chapter 4</b> for more details.   |
| <b>&lt;percent&gt;</b>         | Integer type. The update progress in percentage.  |
| <b>&lt;update_err&gt;</b>      | Integer type. Update result code.<br>0 Updated successfully.<br>Any other value Errors. See <b>Chapter 4</b> for more details.  |
| <b>&lt;err&gt;</b>             | Error codes. See <b>Chapter 4</b> for more details.   |

### NOTE

The modules support two security modes for CoAP over DTLS: certificate mode and pre-shared key (PSK) mode.

- In certificate mode, certificates are necessary for downloading the delta package from the CoAPS server. They should be uploaded to */datatx/ssl/* in APP EFS, and must be renamed in accordance with the following rules:
  - 1) The root certificate is renamed as *coap-cert.pem*;
  - 2) The client certificate is renamed as *coap-cacert.pem*;
  - 3) The private key is renamed as *coap-key.pem*.
- In PSK mode, the PSK is required by server. The PSK key file should be renamed to *coap\_fota.psk* and uploaded into the */datatx/ssl/* folder.

## 2.2.4.3. AT+QFOTADL=<FTP\_URL> Update Firmware When Delta Package is Stored on FTP(S)

### Server

When the delta firmware package is stored on an FTP(S) server, execute **AT+QFOTADL=<FTP\_URL>** to start automatic firmware updating via DFOTA. The module downloads the delta package from the FTP(s) server over the air and updates the firmware automatically.

## AT+QFOTADL=<FTP\_URL> Update Firmware When Delta Package is Stored on FTP(S) Server

|                                       |  |
|---------------------------------------|--|
| Write Command<br>AT+QFOTADL=<FTP_URL> | <p>Response</p> <p>OK</p> <p>+QIND: "FOTA","FTPSTART"</p> <p>+QIND: "FOTA","DOWNLOADING",&lt;percent&gt;</p> <p>+QIND: "FOTA","DOWNLOADING",&lt;percent&gt;</p> <p>...</p> <p>+QIND: "FOTA","FTPEND",&lt;FTP_err&gt;</p> <p>+QIND: "FOTA","RESETTING"</p> <p>+QIND: "FOTA","START"</p> <p>+QIND: "FOTA","UPDATING",&lt;percent&gt;</p> <p>+QIND: "FOTA","UPDATING",&lt;percent&gt;</p> <p>...</p> <p>+QIND: "FOTA","END",&lt;update_err&gt;</p> <p>If there is any error:</p> <p>+CME ERROR: &lt;err&gt;</p> |
| Maximum Response Time                 | 300 ms   |
| Characteristics                       | <p>This command takes effect immediately.</p> <p>The configuration will not be saved.</p>  |

### Parameter

|                  |   |
|------------------|---|
| <FTP_URL>        | String type. The maximum length is 700 bytes. It should be started with "FTP://" or "FTPS://", for example:<br>"FTP://<FTP_USER>:<FTP_PASS>@<FTP_server_URL>:<FTP_port>/<FTP_file_path>". |
| <FTP_USER>       | String type. The username for FTP(S) server login.  |
| <FTP_PASS>       | String type. The password of FTP(S) server login.   |
| <FTP_server_URL> | String type. The IP address or domain name of FTP(S) server.  |
| <FTP_port>       | Integer type. The port number of FTP(S) server. Default value: 21.<br>Range: 1–65535.   |
| <FTP_file_path>  | String type. File path on FTP(S) server.  |
| <FTP_err>        | Integer type. FTP(S) result code.<br>0 Updated successfully.<br>Any other value Errors. Refer to <b>Chapter 4</b> for more details.   |
| <percent>        | Integer type. The download or update progress in percentage.  |
| <update_err>     | Integer type. Update result code.<br>0 Updated successfully.<br>Any other value Errors. See <b>Chapter 4</b> for more details.  |

<err> Error codes. See **Chapter 4** for more details.

**NOTE**

1. If SSL certificates are needed while downloading the firmware package from an FTPS server, they should be uploaded to /datatx in APP EFS, and must be renamed in accordance with the following rules:
  - 1) The root certificate must be renamed *ftp\_fota\_cacert.pem*;
  - 2) The client certificate must be renamed *ftp\_fota\_client\_cert.pem*;
  - 3) The private key must be renamed *ftp\_fota\_client\_key.pem*.
2. Certificate upload methods:
  - Quectel QEFS Explorer tool can be used to upload the certificates. For more details about the tool, see **document [2]**.
  - **AT+QFUPL** can also be used to upload the certificates. For more details about the command, see **document [3]**.

#### 2.2.4.4. AT+QFOTADL="COM:",<file\_size>,<timeout> Upload a Delta Package to Flash

This command uploads a delta package to the NAND flash. If a delta package already exists in the NAND flash, it will be overwritten when a new package is uploaded.

After this command is executed and **CONNECT** is returned, the module switches to the data mode. When the uploaded data reaches <file\_size>, or if no data are inputted when <timeout> is reached, the module exits the data mode automatically. During data transmission, inputting +++ or pulling up MAIN\_DTR can make the module exit the data mode. More information about the data mode is provided in **Chapter 2.2.3**.

#### AT+QFOTADL="COM:",<file\_size>,<timeout> Upload a Delta Package to Flash

| Write Command  | Response  |
|--|---|
| <b>AT+QFOTADL="COM:",&lt;file_size&gt;,&lt;timeout&gt;</b> | <b>CONNECT</b>  |
|  | TA switches to the data mode (transparent transmission mode), and the delta package in the binary form can be inputted. When the total size of the inputted data reaches <file_size>, TA returns to the command mode and provides the following response:<br><b>+QFOTADL: &lt;upload_size&gt;,&lt;check_sum&gt;</b> |
|  | <b>OK</b>   |
|  | If there is any error:<br><b>+CME ERROR: &lt;err&gt;</b>  |

|                       |                                       |
|-----------------------|---------------------------------------|
| Maximum Response Time | 300 ms                                |
| Characteristics       | The command takes effect immediately. |

### Parameter

|               |   |
|---------------|---|
| <file_size>   | Integer type. The file size expected to be uploaded. Unit: byte.                                    |
| <timeout>     | Integer type. The time waiting for data to be inputted to USB/UART. Default value: 5. Unit: second. |
| <upload_size> | Integer type. The file size that has been uploaded. Unit: byte.                                     |
| <check_sum>   | The checksum of the uploaded data.  |
| <err>         | Error code. See <b>Chapter 4</b> for more details.  |

### NOTE

1. <file\_size> is limited by the flash space for storing the delta firmware package. Since the flash for storing the delta package is also used for storing DFOTA algorithm, <file\_size> may vary with firmware versions. The maximum flash space for storing DFOTA package is calculated automatically by the module, and you can query it with **AT+QFOTADL=12** (see **Chapter 2.2.4.5** for details).
2. <check\_sum> is a 16-bit checksum based on bitwise Exclusive-OR (XOR). When the inputted data is odd byte size, the XOR operator sets the last inputted byte as the upper 8 bits, and sets the lower 8 bits as 0. The checksum verifies if the data have been uploaded correctly.
3. +++ sequence causes TA to end the command and switch to the command mode. However, the data previously uploaded are preserved into the file.
4. When executing the command, the data must be entered after **CONNECT** is returned.

### 2.2.4.5. AT+QFOTADL=<para1>[,<para2>[,<para3>]] Extended DFOTA Configurations

This command supports several extended configurations through different parameters, such as triggering firmware updating with a local delta firmware package, getting delta firmware package information, and deleting a delta firmware package.

#### AT+QFOTADL=<para1>[,<para2>[,<para3>]] Extended DFOTA Configurations

|                                    |  |
|------------------------------------|--|
| Read Command<br><b>AT+QFOTADL?</b> | Response<br><b>+QFOTADL: &lt;state&gt;</b><br><br><b>OK</b><br><br>If <state> is 3, return the size of the delta package that has been downloaded in DFOTA process:<br><b>+QFOTADL: 3,&lt;delta_size&gt;</b> |
|------------------------------------|--|

|   |   |
|---|---|
|   | OK  |
| <p>Write Command</p> <p><b>AT+QFOTADL=&lt;para1&gt;[,&lt;para2&gt;[,&lt;para3&gt;]]</b></p> | <p>Response</p> <p>If &lt;para1&gt;=1, start DFOTA updating:</p> <p>OK</p> <p>If &lt;para1&gt;=3, and both &lt;para2&gt; and &lt;para3&gt; are omitted, return the current DFOTA updating trigger mode:</p> <p><b>+QFOTADL: 3,&lt;para2&gt;</b></p> <p>OK</p> <p>If &lt;para1&gt;=3, and &lt;para2&gt; is specified, set DFOTA updating trigger mode:</p> <p>OK</p> <p>If &lt;para1&gt;=4, and both &lt;para2&gt; and &lt;para3&gt; are omitted, return the size of local delta package:</p> <p><b>+QFOTADL: 4,&lt;para2&gt;</b></p> <p>OK</p> <p>If &lt;para1&gt;=5, and both &lt;para2&gt; and &lt;para3&gt; are omitted, delete the local delta package:</p> <p>OK</p> <p>If &lt;para1&gt;=8, and both &lt;para2&gt; and &lt;para3&gt; are omitted, query retry count and intervals:</p> <p><b>+QFOTADL: 8,&lt;para2&gt;,&lt;para3&gt;</b></p> <p>OK</p> <p>If &lt;para1&gt;=8, and both &lt;para2&gt; and &lt;para3&gt; are specified, set retry count and intervals:</p> <p>OK</p> <p>If &lt;para1&gt;=10, &lt;para2&gt; is specified and &lt;para3&gt; is omitted, download files to EUFS file system from HTTP(S) server and the downloaded-file path will be /datatx/update.zip.</p> <p>OK</p> <p>If &lt;para1&gt;=10, and both &lt;para3&gt; and &lt;para2&gt; are specified, download files to EUFS file system from HTTP(S) server and the downloaded-file path is specified by &lt;para3&gt;.</p> |

|                       |  |
|-----------------------|--|
|                       | <p><b>OK</b></p> <p>If <b>&lt;para1&gt;</b>=12, and both <b>&lt;para2&gt;</b> and <b>&lt;para3&gt;</b> are omitted, return the maximum flash size for storing delta package.<br/> <b>+QFOTADL: 12,&lt;para2&gt;</b></p> <p><b>OK</b></p> <p>If there is any error:<br/> <b>+CME ERROR: &lt;err&gt;</b></p> <p>If there is any other error:<br/> <b>ERROR</b></p> |
| Maximum Response Time | 10 s   |
| Characteristics       | <p>The command takes effect immediately.</p> <p>The configuration will not be saved.</p>   |

## Parameter

|                      |   |
|----------------------|---|
| <b>&lt;para1&gt;</b> | <p>Integer type.</p> <ol style="list-style-type: none"> <li>1 Start DFOTA updating</li> <li>2 Reserved</li> <li>3 Configure whether to trigger DFOTA updating immediately after delta package is downloaded from HTTP(S) server</li> <li>4 Get the size of local delta package</li> <li>5 Delete the local delta package</li> <li>6 Cancel the current delta package downloading</li> <li>7 Reserved</li> <li>8 Configure retry count and intervals when the connection breaks down</li> <li>9 Reserved</li> <li>10 Download files from HTTP(S) server and save them into EUFS file system</li> <li>11 Reserved</li> <li>12 Get the maximum flash size for storing delta package</li> </ol> |
| <b>&lt;para2&gt;</b> | <p>When <b>&lt;para1&gt;</b>=3:</p> <p><b>&lt;para2&gt;</b> species the DFOTA updating trigger mode:</p> <ol style="list-style-type: none"> <li>0 Trigger DFOTA updating with <b>AT+QFOTADL=1</b>.</li> <li>1 Trigger DFOTA updating automatically when the package is downloaded.</li> </ol> <p>When <b>&lt;para1&gt;</b>=4:</p> <p><b>&lt;para2&gt;</b> is only available in the command response, and it represents the size of local delta package in integer type. Unit: byte.</p> <p>If <b>&lt;para1&gt;</b>=8:</p> <p><b>&lt;para2&gt;</b> is used to query/configure retry count. Range:1–2147483647.</p> <p>If <b>&lt;para1&gt;</b>=10:</p>  |

|                           |   |
|---------------------------|---|
|                           | <p><b>&lt;para2&gt;</b> represents the URL address of file stored on the HTTP(S) server.</p> <p>If <b>&lt;para1&gt;=12</b>:</p> <p><b>&lt;para2&gt;</b> represents the maximum flash size for storing delta package. Unit: byte.</p> <p>In other cases, <b>&lt;para2&gt;</b> is omitted.</p>                        |
| <b>&lt;para3&gt;</b>      | <p>If <b>&lt;para1&gt;=8</b>:</p> <p><b>&lt;para3&gt;</b> represents the time interval between two retries. Unit: second.</p> <p>if <b>&lt;para1&gt;=10</b>:</p> <p><b>&lt;para3&gt;</b> represents the local file path for saving the downloaded file.</p> <p>In other cases, <b>&lt;para3&gt;</b> is omitted.</p> |
| <b>&lt;state&gt;</b>      | <p>Integer type. Delta package downloading state.</p> <p>0    Downloading not started</p> <p>1    Downloading</p> <p>2    Downloaded</p> <p>3    Downloading failure</p>  |
| <b>&lt;delta_size&gt;</b> | <p>Integer type. Size of the current delta package that has been downloaded in DFOTA process. Unit: byte.</p>   |
| <b>&lt;err&gt;</b>        | <p>Error code. See <b>Chapter 4</b> for more details.</p>   |

#### 2.2.4.6. AT+QFOTADL=<file\_name>,<type> Copy Delta Package to Flash from File System

This command copies the delta package to the NAND flash from Modem or AP file system. If the delta package has existed on the NAND flash, it will be overwritten by the latest package.

| AT+QFOTADL=<file_name>,<type> Copy Delta Package to Flash from File System |  |
|--|--|
| Write Command<br><b>AT+QFOTADL=&lt;file_name&gt;,&lt;type&gt;</b>          | <p>Response</p> <p><b>OK</b></p> <p>If there is any error:</p> <p><b>+CME ERROR: &lt;err&gt;</b></p> |
| Maximum Response Time  | /  |
| Characteristics  | The command takes effect immediately.  |

#### Parameter

|                          |   |
|--------------------------|---|
| <b>&lt;file_name&gt;</b> | <p>String type. Name of the file to be copied. The maximum length is 128 bytes.</p> <p>"UFS:&lt;file&gt;"      Modem UFS file</p> <p>"EUFS:&lt;file&gt;"     AP EUFS file</p> <p><b>&lt;file&gt;</b> above indicates the exact name of the file in Modem or AP UFS.</p> |
| <b>&lt;type&gt;</b>      | <p>Integer type. The type of file to be copied to NAND flash.</p> <p>0    The first file to be copied to NAND flash</p> <p>1    The intermediate file(s) to be copied to NAND flash</p>   |

|       |   |   |
|-------|---|---|
|       | 2 | The last file to be copied to NAND flash            |
| <err> |   | Error Codes. See <b>Chapter 4</b> for more details. |

#### NOTE

1. UFS is a User File Storage directory on modem side. It is a special directory on the flash file system. EUFS is an Extended User File Storage on application side.
2. If there is only one file in AP EUFS/Modem UFS, specify <type> as 2.

### 2.2.5. AT+QCFGEXT="fota\_apn" Configure IP Family and APN for DFOTA

This command configures the IP family and APN for DFOTA or queries the current setting.

#### AT+QCFGEXT="fota\_apn" Configure IP Family and APN for DFOTA

|  |  |
|--|--|
| Write Command<br><b>AT+QCFGEXT="fota_apn",[&lt;iptype&gt;,&lt;apn&gt;[,&lt;username&gt;,&lt;password&gt;]]</b> | Response<br>If the optional parameters are omitted, query the current setting.<br><b>+QCFGEXT: "fota_apn",&lt;iptype&gt;,&lt;apn&gt;[,&lt;username&gt;,&lt;password&gt;]</b><br><br><b>OK</b><br><br>If any of the optional parameters is specified, set the IP family and APN for DFOTA.<br><b>OK</b><br><br>If there is any error:<br><b>ERROR</b> |
| Maximum Response Time  | 300 ms   |
| Characteristics  | The command takes effect immediately.<br>The configurations will be saved automatically.   |

#### Parameter

|            |  |
|------------|--|
| <iptype>   | Integer type. IP family.<br>0 IPv4 address family<br>1 IPv6 address family<br>2 IPv4 and IPv6 address family |
| <apn>      | String type. Access point name.  |
| <username> | String type. Username of the selected APN.   |
| <password> | String type. Password of the selected APN.   |



### 2.2.6. AT+QCFGEXT="fota\_http\_header" Configure HTTP(S) Header for DFOTA over HTTP(S)

This command configures HTTP(S) header for DFOTA over HTTP(S).

| AT+QCFGEXT="fota_http_header" Configure HTTP(S) Header for DFOTA over HTTP(S) |   |
|---|---|
| Write Command<br><b>AT+QCFGEXT="fota_http_header"</b><br>[,<key>,<value>]     | <p>Response</p> <p>If the optional parameters are omitted, query the current setting.<br/><b>+QCFGEXT: "fota_http_header"[,key:&lt;key&gt; value:&lt;value&gt;]</b></p> <p><b>OK</b></p> <p>If the optional parameters are specified, configure HTTP(S) header for DFOTA over HTTP(S).<br/><b>OK</b></p> <p>If there is any error:<br/><b>+CME ERROR: &lt;err&gt;</b></p> |
| Maximum Response Time   | 300 ms  |
| Characteristics   | <p>The command takes effect immediately.</p> <p>The configurations will not be saved.</p>   |

#### Parameter

|         |  |
|---------|--|
| <key>   | String type. Key of HTTP(S) header.                      |
| <value> | String type. Value of HTTP(S) header.                    |
| <err>   | Error codes. Refer to <b>Chapter 4</b> for more details. |

#### NOTE

1. The maximum number of HTTP(S) header is 5.
2. When <key> is specified while <value> is empty, this command deletes this key-value pair.
3. In the response of **AT+QCFGEXT="fota\_http\_header"**, <key> and <value> are strings without double quotation marks.

### 2.2.7. AT+QCFGEXT="fota\_wd\_gpio" Configure Watchdog Pin and Feeding Interval During DFOTA

As the DAM cannot run during DFOTA process, you cannot feed the watchdog in the DAM app if an external watchdog is used to monitor the module. In this case, the module may be reset by the external

watchdog because of watchdog timeout. In order to avoid this, Quectel implements a feature to feed the watchdog automatically in AP kernel during DFOTA. This command enables or disables the external watchdog feeding, and configures watchdog pins and the watchdog feeding interval during DFOTA.

### AT+QCFGEXT="fota\_wd\_gpio" Configure Watchdog Pin and Feeding Interval During DFOTA

|  |  |
|--|--|
| Write Command<br><b>AT+QCFGEXT="fota_wd_gpio"[,&lt;switch&gt;[,&lt;pin_num&gt;,&lt;feed_interval&gt;]]</b> | Response<br>If the optional parameters are omitted, query the current setting:<br><b>+QCFGEXT: "fota_wd_gpio",&lt;switch&gt;[,&lt;pin_num&gt;,&lt;feed_interval&gt;]</b><br><br><b>OK</b><br><br>If the optional parameters are specified, configure the watchdog pin and the feeding interval during DFOTA<br><b>OK</b><br><br>If there is any error:<br><b>ERROR</b> |
| Maximum Response Time  | 300 ms   |
| Characteristics  | The command takes effect immediately.<br>The configurations will be saved automatically.   |

### Parameter

|                              |   |
|------------------------------|---|
| <b>&lt;switch&gt;</b>        | Integer type. Enable/disable watchdog during DFOTA.<br>0    Disable<br>1    Enable  |
| <b>&lt;pin_num&gt;</b>       | Integer type. Pin number for watchdog.<br>BG95 series supports the pin numbers below:<br>4–7, 18, 19, 22, 23, 25–28, 40, 41, 64–66, 85–88.<br>BG600L-M3 supports the pin numbers below:<br>9–12, 22, 23, 29, 30, 53, 54, 57–62.<br>BG77 supports the pin numbers below:<br>1–5, 8, 9, 33–37, 40, 41, 48–51, 57, 60, 61, 63, 67–71, 77, 80–82, 90, 91, 93. |
| <b>&lt;feed_interval&gt;</b> | Integer type. The interval to feed the watchdog. The period of square wave is 2 × <b>&lt;feed_interval&gt;</b> . Unit: millisecond. Range: 100–3600000.   |
| <b>&lt;err&gt;</b>           | Error codes. See <b>Chapter 4</b> for more details.   |

## 2.2.8. AT+QCFGEXT="sni" Configure Whether to Enable Server Name Indication for DFOTA

### AT+QCFGEXT="sni" Configure Whether to Enable Server Name Indication for DFOTA

|  |   |
|--|---|
| Write Command<br><b>AT+QCFGEXT="sni"[,&lt;SNI&gt;]</b> | <p>Response</p> <p>If the optional parameter is omitted, query the current setting:<br/><b>+QCFGEXT: "sni",&lt;SNI&gt;</b></p> <p><b>OK</b></p> <p>If the optional parameter is specified, configure whether to enable server name indication for DFOTA:<br/><b>OK</b></p> <p>If there is any error:<br/><b>ERROR</b></p> |
| Maximum Response Time                                  | 300 ms  |
| Characteristics  | <p>The command takes effect immediately.</p> <p>The configuration will not be saved.</p>  |

#### Parameter

|                    |   |
|--------------------|---|
| <b>&lt;SNI&gt;</b> | Integer type. Whether to enable server name indication. |
| <u>0</u>           | Disable   |
| 1                  | Enable  |

## 2.3. Example

### 2.3.1. Update Firmware from HTTP(S) Server by DFOTA

//The delta firmware package has already been stored on an HTTP(S) server. The server address is "https://www.quectel.com:100/update.zip".

**AT+CSQ;+CEREG?;+CGREG?;+COPS?** //Query network status.

**+CSQ: 26,99**

**+CEREG: 0,1**

**+CGREG: 0,4**

```
+COPS: 0,0,"Telstra Mobile Telstra",8
```

```
OK
```

//Execute **AT+QFOTADL** command to enable automatic firmware update via DFOTA, and then the module starts to download the delta package and update firmware automatically.

```
AT+QFOTADL="https://www.quectel.com:100/update.zip"
```

```
OK
```

```
+QIND: "FOTA","HTTPSTART"
```

```
+QIND: "FOTA","DOWNLOADING",5%
```

```
...
```

```
+QIND: "FOTA","HTTPEND",0
```

//Finish downloading the package from the server.

```
+QIND: "FOTA","RESETTING"
```

//The module is reset and then enters DFOTA mode.

```
+QIND: "FOTA","START"
```

```
+QIND: "FOTA","UPDATING",1%
```

```
+QIND: "FOTA","UPDATING",2%
```

```
...
```

```
+QIND: "FOTA","UPDATING",100%
```

```
+QIND: "FOTA","END",0
```

//Finish updating the firmware.

### 2.3.2. Update Firmware from FTP(S) Server by DFOTA

//Update firmware when delta firmware package is stored on an FTP(S) server. The server address is "ftp://test:test@220.180.239.212:8309/TEST1/upgrade.bin". Execute **AT+QFOTADL** to enable automatic firmware updating via DFOTA, and then the module will download the delta package and update firmware automatically.

```
AT+QFOTADL="ftp://test:test@220.180.239.212:8309/TEST1/upgrade.bin"
```

```
OK
```

```
+QIND: "FOTA","FTPSTART"
```

```
+QIND: "FOTA","DOWNLOADING",5%
```

```
+QIND: "FOTA","DOWNLOADING",10%
```

```
+QIND: "FOTA","DOWNLOADING",15%
```

```
+QIND: "FOTA","DOWNLOADING",20%
```

```
...
```

```
+QIND: "FOTA","DOWNLOADING",95%
```

```
+QIND: "FOTA","DOWNLOADING",100%
```

```
+QIND: "FOTA","FTPEND",0
```

//Finish the package downloading from the server.

```
+QIND: "FOTA","RESETTING"
```

//The module resets and then enters DFOTA mode.

```
+QIND: "FOTA","START"
```

```
+QIND: "FOTA","UPDATING",1%
```

```
+QIND: "FOTA","UPDATING",4%
```

```
...
```

```
+QIND: "FOTA","UPDATING",99%
```

```
+QIND: "FOTA","UPDATING",100%
```

```
+QIND: "FOTA","END",0
```

//Finish the firmware update.

### 2.3.3. Copy Delta Package to NAND Flash from AP EUFS

//Assume that the delta package is divided the into four parts: *update1.bin*, *update2.bin*, *update3.bin* and *update4.bin*. Then, download the four packages to AP EUFS.

```
AT+QFOTADL="EUFS:update1.bin",0    //Copy update1.bin to NAND flash.
OK
AT+QFOTADL="EUFS:update2.bin",1    //Copy update2.bin to NAND flash.
OK
AT+QFOTADL="EUFS:update3.bin",1    //Copy update3.bin to NAND flash.
OK
AT+QFOTADL="EUFS:update4.bin",2    //Copy update4.bin to NAND flash.
OK
```

### 2.3.4. Set Breakpoint Continuous Downloading and Single Request Size for HTTP(S)

#### DFOTA Downloading

//Update firmware when delta firmware package is stored on HTTP(S) server. The server address is "https://www.quectel.com:100/update.zip".

```
AT+CSQ;+CEREG?;+CGREG?;+COPS?    //Query network status
+CSQ: 26,99
```

```
+CEREG: 0,1
```

```
+CGREG: 0,4
```

```
+COPS: 0,0,"Telstra Mobile Telstra",8
```

```
OK
```

//Enable automatic firmware updating via DFOTA.

```
AT+QFOTADL=3,1
```

```
OK
```

//Set breakpoint continuous downloading and the request size.

```
AT+QFOTADL="https://www.quectel.com:100/update.zip",0,10240    //Download the first 10240 bytes
                                                                of the delta package.
```

```
OK
```

```
+QIND: "FOTA","HTTPSTART"
```

```
+QIND: "FOTA","DOWNLOADING",5%
```

```
...
```

```
+QIND: "FOTA","HTTPEND",712    //Finished downloading of the segmented delta package from
                                the server.
```

```
AT+QFOTADL?                    //Query the size of the segmented delta package that has been
                                downloaded in DFOTA process.
```

```

+QFOTADL: 3,10240

OK

//Set the breakpoint position to 10240 and the request size to 0 (i.e., download the package from the
breakpoint until the entire package is downloaded completely).
AT+QFOTADL="https://www.quectel.com:100/update.zip",10240
OK
+QIND: "FOTA","HTTPSTART"
+QIND: "FOTA","DOWNLOADING",10%
...
+QIND: "FOTA","DOWNLOADING",100%
+QIND: "FOTA","HTTPEND",0
+QIND: "FOTA","RESETTING" //The module resets and enters DFOTA mode automatically.
+QIND: "FOTA","START"
+QIND: "FOTA","UPDATING",1%
+QIND: "FOTA","UPDATING",2%
...
+QIND: "FOTA","UPDATING",100%
+QIND: "FOTA","END",0 //Finish firmware updating.

```

### 2.3.5. Configure HTTP(S) Header for DFOTA over HTTP(S)

```

//Set the user-defined HTTP(S) header configuration.
AT+QCFGEXT="fota_http_header","Auth","test"
OK
//Query the user-defined HTTP(S) header configuration.
AT+QCFGEXT="fota_http_header"
+QCFGEXT: key:Auth    value:test

OK
//Delete this key-value pair.
AT+QCFGEXT="fota_http_header","Auth",""
OK

```

### 2.3.6. Set Watchdog Configuration for DFOTA

```

//Set the watchdog configuration for BG95 series.
AT+QCFGEXT="fota_wd_gpio",1,6,3000 //Enable the watchdog, set pin 6 as the pin for watchdog and
feed the watchdog at an interval of 3000 milliseconds.

OK
AT+QCFGEXT="fota_wd_gpio" //Query the current watchdog configuration.
+QCFGEXT: "fota_wd_gpio",1,6,3000

```

OK

`AT+QCFGEXT="fota_wd_gpio",0`

//Disable watchdog.

OK

# 3 Trigger DFOTA with LwM2M

## 3.1. Firmware Update Procedure

OMA defines the application layer communication protocol between a LwM2M server and a LwM2M client, which is located in a LwM2M device. For the modules, the LwM2M protocol is used to manage device provisioning and is a trigger of a DFOTA procedure. For the DFOTA update on the modules, a firmware update object is provided by the LwM2M specification.

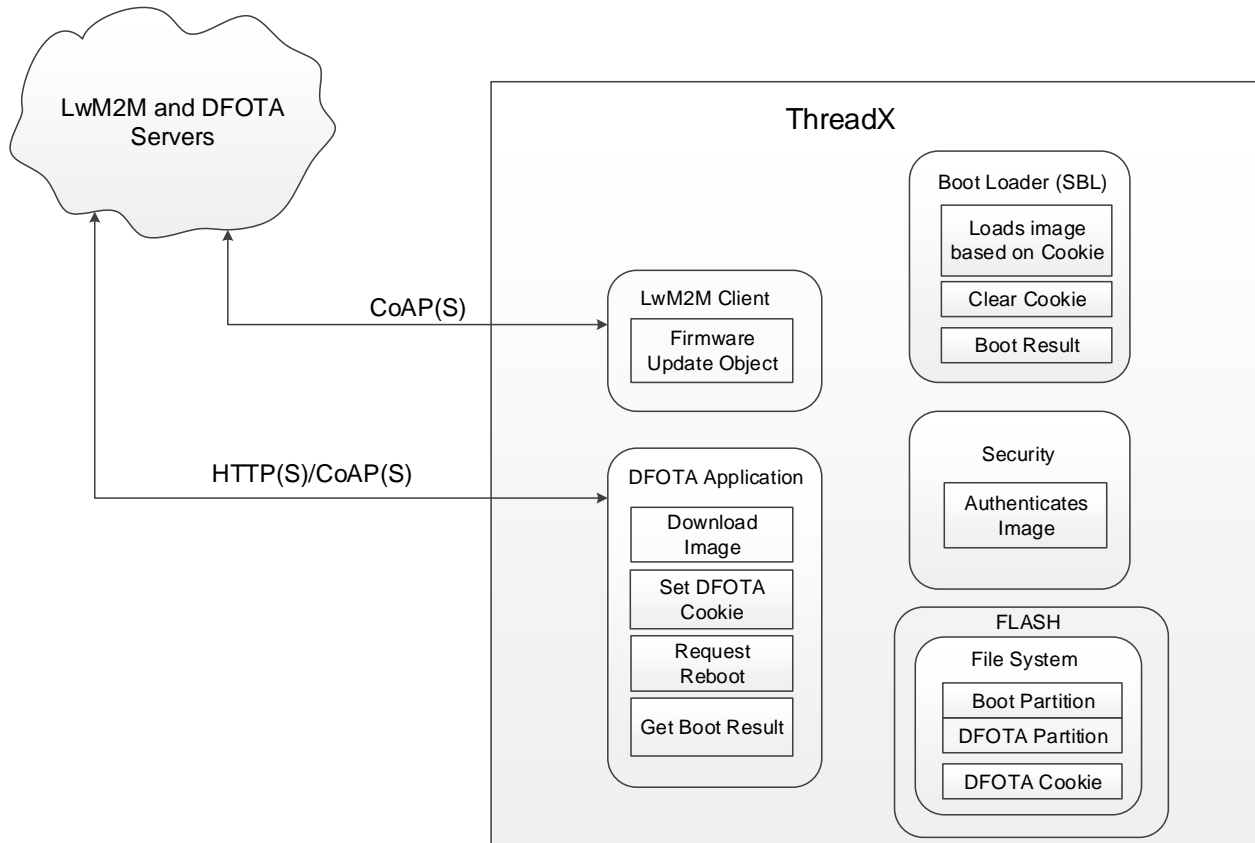
The following are the routine DFOTA steps with LwM2M:

1. Write “Package”/“Package URI” to trigger the downloading of a firmware image, so that the DFOTA application downloads the firmware image from LwM2M server automatically.
2. Post “Update” to trigger firmware update on the modules, so that the DFOTA application finishes firmware update automatically with the candidate firmware image.

The LwM2M client is notified about the new firmware by the LwM2M server using CoAP(S). The client sends indication to a registered DFOTA application. The DFOTA application downloads the firmware delta package from the server and stores it on the file system of BG95 series, BG77 and BG600L-M3 modules.

The following figure shows key components involved in the LwM2M DFOTA process on the modules.





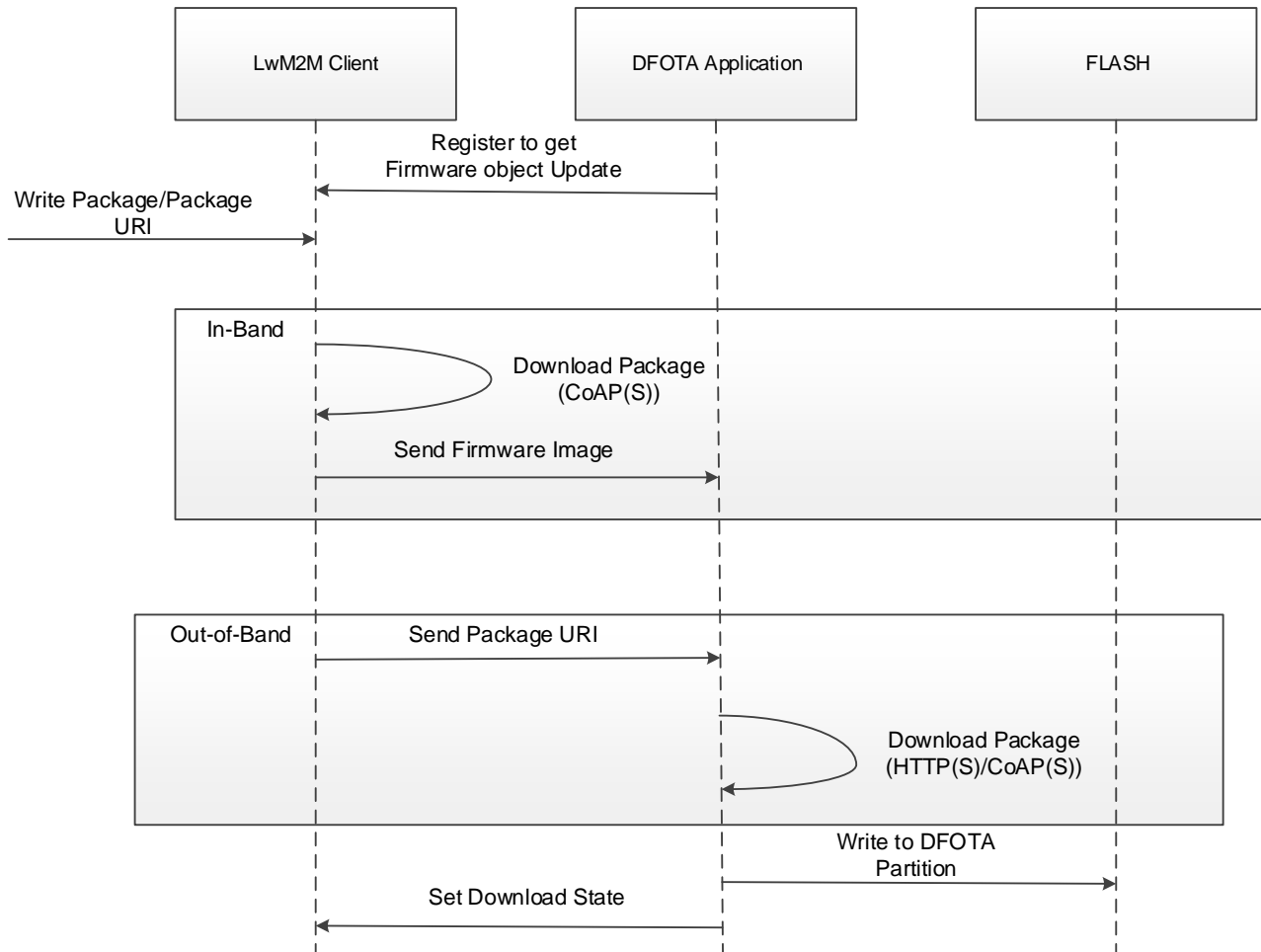
**Figure 2: LwM2M Firmware Image Download**

### 3.1.1. Download a Firmware Image

About “Firmware Update Object” in LwM2M, the detailed interaction process between LwM2M client and

LwM2M server is described as below.

The LwM2M client should register and observe the “Package URI” resource in the Firmware Update Object, which is shown in the server when the client is registered successfully.



**Figure 3: Call Flow of Package URI Observing and Firmware Downloading**

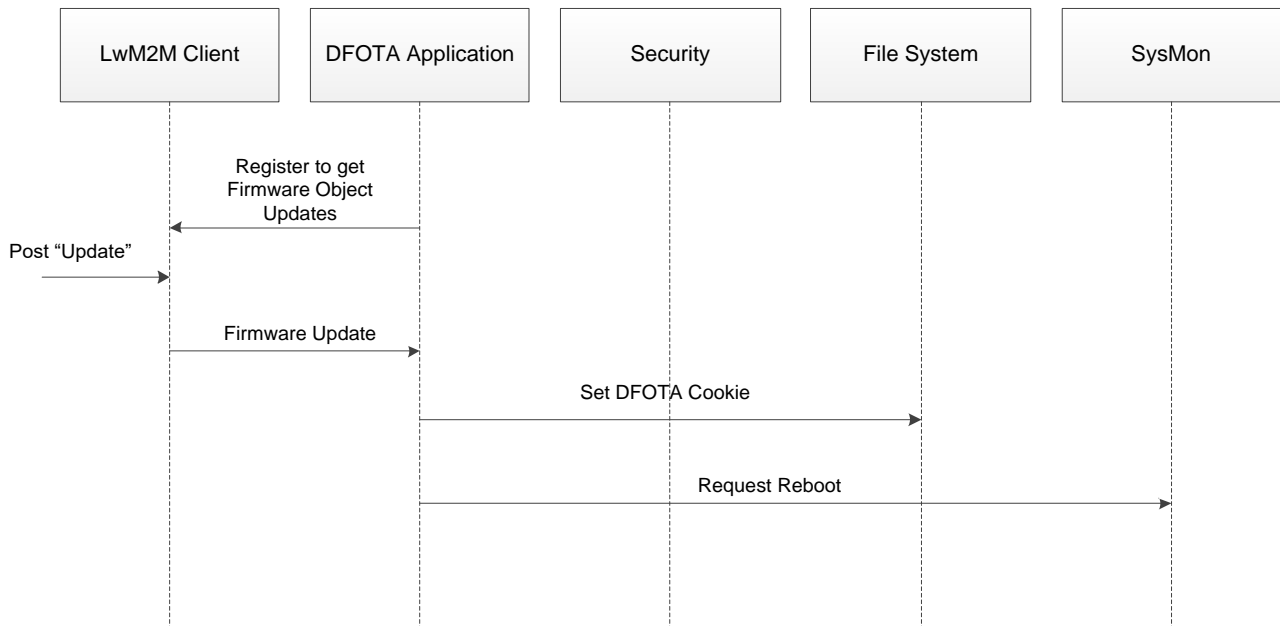
**NOTE**

In the in-band mode, the module downloads a firmware package from the LwM2M server directly via the CoAP(S) protocol and sends the downloaded package to the DFOTA application. All "firmware update" actions are done in the application. But in the out-of-band mode, the module downloads a firmware package from the HTTP(S)/CoAP(S) server to the module.

### 3.1.2. Update the Firmware

The LwM2M server should observe the “State” resource in the Firmware Update Object. If the “State” has

changed to “Downloaded”, “Update” resource is executed to trigger the firmware update process.



**Figure 4: Call Flow of “Update” Resource Observing and Update Event Processing**

## 3.2. URC of Firmware Update with LwM2M

### 3.2.1. Download Progress URCs

#### 3.2.1.1. In-Band Mode

When the DFOTA firmware update is triggered in the LwM2M in-band mode, the following URCs are outputted to notify the download progress.

#### **+LWM2M: "FOTA","DOWNLOADING" Notify the Download Progress**

|   |                             |
|---|-----------------------------|
| URC Format:                                     |                             |
| <b>+LWM2M: "FOTA","DOWNLOADING","START"</b>     | Download process started.   |
| URC Format:                                     |                             |
| <b>+LWM2M: "FOTA","DOWNLOADING","COMPLETED"</b> | Download process completed. |

#### 3.2.1.2. Out-of-Band HTTP(S) Mode

When the DFOTA firmware update is triggered in the LwM2M out-of-band HTTP(S) mode, the following URCs are outputted to notify the download progress.

### +QIND: "FOTA","HTTP" Notify the Download Progress

|  |                                  |
|--|----------------------------------|
| URC Format:<br><b>+QIND: "FOTA","HTTPSTART"</b>                          | Download process started.        |
| URC Format:<br><b>+QIND: "FOTA","DOWNLOADING",&lt;percent&gt;</b><br>... | Download progress in percentage. |
| URC Format:<br><b>+LWM2M: "FOTA","DOWNLOADED",&lt;HTTP_err&gt;</b>       | Download operation result.       |

#### Parameter

|                         |  |
|-------------------------|--|
| <b>&lt;percent&gt;</b>  | Integer type. The download progress in percentage.   |
| <b>&lt;HTTP_err&gt;</b> | Integer type. The HTTP(S) error code.<br>0 Downloaded successfully.<br>Any other value Error.<br>Refer to <b>Chapter 4</b> for more details. |

#### 3.2.1.3. Out-of-Band CoAP(S) Mode

When the DFOTA firmware update is triggered in the LwM2M out-of-band CoAP(S) mode, the following URCs are outputted to notify the download progress.

### +QIND: "FOTA","COAP" Notify the Download Progress

|  |                            |
|--|----------------------------|
| URC Format:<br><b>+QIND: "FOTA","COAPSTART"</b>                    | Download process started.  |
| URC Format:<br><b>+LWM2M: "FOTA","DOWNLOADED",&lt;CoAP_err&gt;</b> | Download operation result. |

#### Parameter

|                         |  |
|-------------------------|--|
| <b>&lt;CoAP_err&gt;</b> | Integer type. The CoAP(S) error code.<br>0 Downloaded successfully.<br>Any other value Error.<br>Refer to <b>Chapter 4</b> for more details. |
|-------------------------|--|

### 3.2.2. Update Progress URCs

#### 3.2.2.1. +QIND: "FOTA","START" Updating Started

| +QIND: "FOTA","START" Updating Started |                                   |
|--|-----------------------------------|
| URC Format:<br>+QIND: "FOTA","START"   | Notify the start of the updating. |

#### 3.2.2.2. +QIND: "FOTA","UPDATING",<percent> Updating Progress

| +QIND: "FOTA","UPDATING",<percent> Updating Progress |   |
|--|---|
| URC Format:<br>+QIND: "FOTA","UPDATING",<percent>    | Notify the updating progress in percentage. |

#### Parameter

|           |  |
|-----------|--|
| <percent> | Integer type. The updating progress in percentage. |
|-----------|--|

#### 3.2.2.3. +QIND: "FOTA","END",<err> Updating Finished

| +QIND: "FOTA","END",<err> Updating Finished |                                    |
|---|------------------------------------|
| URC Format:<br>+QIND: "FOTA","END",<err>    | Notify the completion of updating. |

#### Parameter

|       |  |
|-------|--|
| <err> | Error code. See <b>Chapter 4</b> for more details. |
|-------|--|

# 4 Summary of Error Codes

These error codes indicate errors related to mobile equipment or network. The details about **<HTTP\_err>**, **<CoAP\_err>**, **<FTP\_err>**, **<update\_err>** and **<err>** are described in the following tables.

**Table 2: Summary of <HTTP\_err>/<CoAP\_err>/<FTP\_err> Codes**

| <b>&lt;HTTP_err&gt;/&lt;CoAP_err&gt;/&lt;FTP_err&gt;</b> | <b>Meaning</b>                                   |
|--|--|
| 0  | Download successful                              |
| 701  | Unknown error                                    |
| 702  | Server connection failed                         |
| 703  | Request failed                                   |
| 704  | Download timeout                                 |
| 705  | URL error  |
| 706  | File does not exist                              |
| 707  | Write data to file failed                        |
| 708  | Downloaded file is too large                     |
| 709  | Download cancelled                               |
| 710  | Downloaded package check failed                  |
| 711  | Data call disconnected                           |
| 712  | Custom segmented download completed successfully |

**Table 3: Summary of <update\_err> Codes**

| <update_err> | Meaning   |
|--------------|---|
| 0            | Updated successfully                            |
| 504          | Firmware update failed                          |
| 505          | Update package does not exist                   |
| 506          | Update package check failed                     |
| 507          | Decompress failed                               |
| 508          | Compress failed                                 |
| 509          | Copy compressed data failed                     |
| 510          | Restore failed                                  |
| 511          | Package is mismatched with the current firmware |
| 512          | DFOTA unknown error                             |

**Table 4: Summary of <err> Codes**

| <err> | Meaning                                      |
|-------|--|
| 590   | Input parameter invalid                      |
| 591   | DFOTA is ongoing                             |
| 592   | DFOTA downloading is not in progress         |
| 593   | DFOTA package does not exist or open failure |
| 594   | DFOTA memory allocation failed               |

# 5 Appendix A References

**Table 5: Related Documents**

| Document Name  |
|--|
| [1] Quectel_BG95&BG77&BG600L_Series_AT_Commands_Manual       |
| [2] Quectel_BG95&BG77&BG600L_Series_QEFS_Explorer_User_Guide |
| [3] Quectel_BG95&BG77&BG600L_Series_FILE_Application_Note    |

**Table 6: Terms and Abbreviations**

| Abbreviation | Description                               |
|--------------|---|
| AP           | Application Processor                     |
| APN          | Access Point Name                         |
| bps          | Bit(s) Per Second                         |
| CoAP(S)      | Constrained Application Protocol (Secure) |
| DAM          | Downloadable Application Module           |
| DFOTA        | Delta Firmware Upgrade Over-the-Air       |
| EFS          | Embedded File System                      |
| EGPRS        | Enhanced General Packet Radio Service     |
| FOTA         | Firmware Over-the-Air                     |
| HTTP(S)      | Hyper Text Transport Protocol (Secure)    |
| FTP(S)       | File Transfer Protocol (Secure)           |
| IP           | Internet Protocol                         |
| IPv4         | Internet Protocol Version 4               |



---

|       |                                |
|-------|--------------------------------|
| IPv6  | Internet Protocol Version 6    |
| KB    | Kilobytes                      |
| LTE   | Long Term Evolution            |
| LPWA  | Low Power Wide Area            |
| LwM2M | Lightweight Machine to Machine |
| NAND  | Not And                        |
| OMA   | Open Mobile Alliance           |
| SSL   | Secure Sockets Layer           |
| URC   | Unsolicited Result Code        |
| URI   | Uniform Resource Identifier    |

---