

BG95-S5 FILE Application Note

Satellite Communication Module Series

Version: 1.0

Date: 2024-10-10

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

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.

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. 2024. All rights reserved.

About the Document

Revision History

Version	Date	Author	Description
-	2024-08-21	Allen LIU	Creation of the document
1.0	2024-10-10	Allen LIU	First official release

Contents

About the Document	3
Contents	4
Table Index	5
1 Introduction	6
1.1. The Process of Using FILE AT Commands	6
1.2. Description of Data Mode	7
2 Description of FILE AT Commands	8
2.1. AT Command Syntax.....	8
2.1.1. Definitions.....	8
2.1.2. AT Command Syntax	8
2.2. Declaration of AT Command Examples	9
2.3. Description of FILE AT Commands.....	9
2.3.1. AT+QFLDS Get Space Information of Storage Medium	9
2.3.2. AT+QFLST List File Information	10
2.3.3. AT+QFDEL Delete File(s).....	12
2.3.4. AT+QFUPL Upload a File.....	13
2.3.5. AT+QFDWL Download a File	14
2.3.6. AT+QFOPEN Open a File	15
2.3.7. AT+QFREAD Read a File	17
2.3.8. AT+QFWRITE Write Data into a File	18
2.3.9. AT+QFSEEK Set File Pointer to a Specified Position	18
2.3.10. AT+QFPOSITION Get the Offset of a File Pointer	19
2.3.11. AT+QFTUCAT Truncate a File from the File Pointer	20
2.3.12. AT+QFCLOSE Close a File	21
2.3.13. AT+QFCRC Calculate the CRC of a Specified UFS File.....	21
2.3.14. AT+QFCPY Make a Copy of a Specified File.....	22
3 Examples	24
3.1. Upload and Download Files.....	24
3.1.1. Upload a File.....	24
3.1.2. Download a File	25
3.2. Write and Read Files	25
3.2.1. Write and Read a UFS File	25
3.2.2. Write and Read an EUFS File.....	25
4 Summary of Error Codes.....	27
5 Appendix References	29

Table Index

Table 1: Types of AT Commands	8
Table 2: Summary of Error Codes	27
Table 3: Terms and Abbreviations.....	29

1 Introduction

The Quectel BG95-S5 module provides AT commands to manage files on different physical storage mediums. This document is a reference guide to these commands.

The supported storage media include:

- **UFS:** User File Storage on the modem side.
- **EUFS:** Extended User File Storage on the application side.

The **<filename>** of the FILE AT command contains the storage location. When it begins with “UFS:” or with no storage media prefix, it means that the file is stored in the UFS. When it begins with “EUFS:”, it means that the file is stored in the EUFS.

1.1. The Process of Using FILE AT Commands

The following outlines the general procedure to upload/download, open/create, read, and write to a file in the storage:

1. Upload a file to the storage with **AT+QFUPL** and output/download it with **AT+QFDWL** to verify its content and data integrity.
2. Open the file with **AT+QFOPEN**, and then you can read or write to it at any time and any location until it is closed by **AT+QFCLOSE**.
 - When opening the file with **AT+QFOPEN**, you can set it to overwrite mode, read-only mode, or other modes with **<mode>** (see **Chapter 2.3.6** for details). Upon opening, the file will be assigned a **<filehandle>** which is used for subsequent file operations.
 - After the file is opened, you can write to it with **AT+QFWRITE** or read it with **AT+QFREAD**, starting from the current file position.
 - You can adjust the file position with **AT+QFSEEK** and check the current position with **AT+QFPOSITION**.
 - **AT+QFTUCAT** allows you to truncate the file from the current position to the end of the file.
3. Close the file with **AT+QFCLOSE**, after which the **<filehandle>** turns invalid and cannot be used for further operations.

The following are several commonly used commands to manage files in storage media:

1. **AT+QFLDS**: Get the space information of storage.
2. **AT+QFLST**: List the file information in the specified storage medium.
3. **AT+QFDEL**: Delete the file(s) in the specified storage medium.

1.2. Description of Data Mode

The module's COM port has two working modes: the AT command mode and the data mode. In the AT command mode, any data inputted through the COM port is treated as AT commands. While in the data mode, the inputted data is treated as raw data.

- **Enter the data mode**

To enter the data mode, execute **AT+QFUPL**, **AT+QFDWL**, **AT+QFREAD**, or **AT+QFWRITE**. When the module returns **CONNECT**, it indicates that the COM port has successfully entered the data mode. Additionally, you can execute **ATO** to re-enter the data mode.

- **Exit the data mode**

To exit the data mode, input **+++** or pull up MAIN_DTR pin (**AT&D1** should be set first). After **+++** is inputted or MAIN_DTR is pulled up, the execution of above data-mode-entering commands will be interrupted before the module responds **OK**, and at this time, the COM port cannot re-enter the data mode if you execute **ATO**.

To prevent the **+++** from being misinterpreted as data, the following requirements should be followed:

1. Do not input any character for at least 1 second before you input **+++**.
2. Finish Inputting **+++** within 1 second, during which no other character can be inputted.
3. Do not input any character until the **OK** response after **+++** has been inputted.

2 Description of FILE AT Commands

2.1. AT Command Syntax

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 omitted, the new value equals to the previous value or the default settings, unless otherwise specified.
- **Underline** Default setting of a parameter.

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	AT+<cmd>=?	Test the existence of the corresponding command and return information about the type, value, or range of its parameter.
Read Command	AT+<cmd>?	Check the current parameter value of the corresponding command.
Write Command	AT+<cmd>=<p1>[,<p2>[,<p3>[...]]]	Set user-definable parameter value.
Execution Command	AT+<cmd>	Return a specific information parameter or perform a specific action.

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. The URLs, domain names, IP addresses, usernames/accounts, and passwords (if any) in the AT command examples are provided for illustrative and explanatory purposes only, and they should be modified to reflect your actual usage and specific needs.

2.3. Description of FILE AT Commands

2.3.1. AT+QFLDS Get Space Information of Storage Medium

This command gets the space information of the specified storage medium.

AT+QFLDS Get Space Information of Storage Medium	
Test Command AT+QFLDS=?	Response OK
Write Command AT+QFLDS=<storage>	Response +QFLDS: <free_size>,<total_size> OK If there is any error: +CME ERROR: <err>
Execution Command AT+QFLDS	Response +QFLDS: <UFS_file_size>,<UFS_file_number> OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<storage>	String type. Storage medium type.
------------------------	-----------------------------------

	"UFS"	UFS on the modem side
	"EUFS"	Extended UFS on the application side
<free_size>	Integer type.	Free space size of <storage>. Unit: byte.
<total_size>	Integer type.	Total space size of <storage>. Unit: byte.
<UFS_file_size>	Integer type.	Size of all files in <i>ufs</i> directory of UFS. Unit: byte.
<UFS_file_number>	Integer type.	Number of files in <i>ufs</i> directory of UFS.
<err>	Integer type.	Error code. See Chapter 4 for possible <err> values.

Example

```

AT+QFLDS="UFS" //Query the space information of UFS on the modem side.
+QFLDS: 1249984,1562624

OK
AT+QFLDS //Query the size and number of all files in ufs directory of UFS.
+QFLDS: 2330,5

OK

```

2.3.2. AT+QFLST List File Information

This command lists the information of a single file or all files on the specified storage medium.

AT+QFLST List File Information	
Test Command AT+QFLST=?	Response OK
Write Command AT+QFLST=<filename>	Response +QFLST: <filename>,<file_size> [+QFLST: <filename>,<file_size> [...]] OK If there is any error: +CME ERROR: <err>
Execution Command AT+QFLST	Response Return the information of UFS files. +QFLST: <filename>,<file_size> [+QFLST: <filename>,<file_size> [...]] OK

	If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filename>	String type. File to be listed. The maximum length of <filename> is 80 bytes.
"*"	All files in <i>ufs</i> directory of UFS
"UFS:*"	All files in <i>ufs</i> directory of UFS
"UFS:/ufs/*"	All files in <i>ufs</i> directory of UFS
"EUFS:*"	All files in <i>ufs</i> directory of EUFS
"EUFS:/ufs/*"	All files in <i>ufs</i> directory of EUFS
"EUFS:/datatx/*"	All files in <i>datatx</i> directory of EUFS
"<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"UFS:<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"UFS:/ufs/<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"EUFS:<file>"	Specified file <file> in <i>ufs</i> directory of EUFS
"EUFS:/ufs/<file>"	Specified file <file> in <i>ufs</i> directory of EUFS
"EUFS:/datatx/<file>"	Specified file <file> in <i>datatx</i> directory of EUFS
	The <file> in values above indicates the exact file name.
<file_size>	Integer type. File size. Unit: byte.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

Example

```

AT+QFLST="" //List all files in ufs directory of UFS.
+QFLST: "F_M12-1.bmp",562554
+QFLST: "F_M12-10.bmp",562554
+QFLST: "F_M12-11.bmp",562554

OK

AT+QFLST="Test1.txt" //List a specified file "Test1.txt" in ufs directory of UFS.
+QFLST: "Test1.txt",2

OK

AT+QFLST="EUFS:/ufs/*" //List all files in ufs directory of EUFS.
+QFLST: "EUFS:test.txt",10
+QFLST: "EUFS:test2.txt",20

OK

AT+QFLST="EUFS:test.txt" //List a specified file "test.txt" in ufs directory of EUFS.
+QFLST: "EUFS:test.txt",10

```

OK

2.3.3. AT+QFDEL Delete File(s)

This command deletes a single file or all the files on the specified storage medium.

AT+QFDEL Delete File(s)	
Test Command AT+QFDEL=?	Response +QFDEL: <filename> OK
Write Command AT+QFDEL=<filename>	Response OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filename>	String type. File to be deleted. The maximum length of <filename> is 80 bytes.
"*"	All files in <i>ufs</i> directory of UFS (<i>ufs</i> directory remains)
"UFS:.*"	All files in <i>ufs</i> directory of UFS
"UFS: /ufs/*"	All files in <i>ufs</i> directory of UFS
"EUFS:/ufs/*"	All files in <i>ufs</i> directory of EUFS (<i>ufs</i> directory remains)
"EUFS:/datatx/*"	All files in <i>datatx</i> directory of EUFS (<i>datatx</i> directory remains)
"EUFS:.*"	All files in <i>ufs</i> directory of EUFS
"<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"UFS:<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"UFS: /ufs/<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"EUFS:<file>"	Specified file <file> in <i>ufs</i> directory of EUFS
"EUFS:/ufs/<file>"	Specified file <file> in <i>ufs</i> directory of EUFS
"EUFS:/datatx/<file>"	Specified file <file> in <i>datatx</i> of EUFS
	The <file> in values above indicates the exact file name.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

Example

```
AT+QFDEL="*" //Delete all files in ufs directory of UFS.
OK
```

AT+QFDEL="Test1.txt"
OK

//Delete a specified file "Test1.txt" in *ufs* directory of UFS

2.3.4. AT+QFUPL Upload a File

This command uploads a file to the storage medium. If any file in the storage medium has the same name with the file uploaded, an error is reported.

After the Write Command is executed and **CONNECT** is returned, the module switches to data mode. When the uploaded data reaches the specified **<file_size>** or the elapsed time without input reaches the specified **<timeout>**, the module exits the data mode automatically.

During data transmission, you can send **+++** or use **MAIN_DTR** to make the module exit the data mode. For more details, see **Chapter 1.2**.

AT+QFUPL Upload a File

Test Command AT+QFUPL=?	Response +QFUPL: <filename>[(1-<free_size>)],(list of supported <timeout>s),[(list of supported <ackmode>s)] OK
Write Command AT+QFUPL=<filename>[,<file_size>[,<timeout>[,<ackmode>]]]	Response CONNECT TA switches to the data mode (transparent transmission mode), and the binary data of the file can be inputted. When the total size of the inputted data reaches <file_size> , the TA receives +++ , or the time waiting for data input reaches <timeout> , the TA will return to the AT command mode and respond: +QFUPL: <upload_size>,<checksum> OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<free_size>	Integer type. The free space size of <storage> . See AT+QFLDS for more details of <storage> .
--------------------------	--

<filename>	String type. File to be uploaded. The maximum length of <filename> is 80 bytes. "<file>" Specified file <file> in <i>ufs</i> directory of UFS "UFS:<file>" Specified file <file> in <i>ufs</i> directory of UFS "UFS:/ufs/<file>" Specified file <file> in <i>ufs</i> directory of UFS "EUFS:<file>" Specified file <file> in <i>ufs</i> directory of EUFS "EUFS:/ufs/<file>" Specified file <file> in <i>ufs</i> directory of EUFS "EUFS:/datatx/<file>" Specified file <file> in <i>datatx</i> directory of EUFS The <file> in values above indicates the exact file name.
<file_size>	Integer type. The file size expected to be uploaded. Default: 10240. Unit: byte. The maximum length is not greater than <free_size> .
<upload_size>	Integer type. The actual size of the uploaded data. Unit: byte.
<timeout>	Integer type. The time waiting for data to be inputted to USB/UART. Range: 1–65535. Default: 5. Unit: s.
<ackmode>	Integer type. Whether to use ACK mode. 0 Turn off the ACK mode. 1 Turn on the ACK mode.
<checksum>	Integer type. The checksum of the uploaded data.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

NOTE

1. It is strongly recommended to use DOS 8.3 file name format for **<filename>**.
2. **<checksum>** is a 16-bit checksum based on bitwise Exclusive OR (XOR). If the number of characters is odd, the XOR operator will calculate the checksum with the last character set as the high 8 bits and the low 8 bits as 0. Inputting **+++** will make the TA end the data mode and switch to the AT command mode. However, the data previously uploaded will be preserved in the file.
3. When executing the command, the data must be entered after **CONNECT** is returned.
4. The ACK mode is a safeguard against data loss during the upload of large files when the hardware flow control does not work. The ACK mode works as follows:
 - 1) Run **AT+QFUP=<filename>,<file_size>,<timeout>,1** to enable the ACK mode.
 - 2) The module outputs **CONNECT**.
 - 3) MCU sends 1K bytes of data, to which the module responds with an **A**.
 - 4) MCU receives the **A** and then sends the next 1K bytes of data.
 - 5) Steps 3) and 4) are repeated until the transfer is completed.

2.3.5. AT+QFDWL Download a File

This command downloads a specified file from the storage medium.

AT+QFDWL Download a File

Test Command	Response
AT+QFDWL=?	+QFDWL: <filename>

	OK
Write Command AT+QFDWL=<filename>	Response CONNECT TA switches to the data mode, and the binary data of the file will be outputted. When the file is read over or the TA receives +++ , the TA will return to the AT command mode and respond: +QFDWL: <download_size>,<checksum> OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filename>	String type. File to be downloaded. The maximum length of <filename> is 80 bytes.
"<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"UFS:<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"UFS:/ufs/<file>"	Specified file <file> in <i>ufs</i> directory of UFS
"EUFS:<file>"	Specified file <file> in <i>ufs</i> directory of EUFS
"EUFS:/ufs/<file>"	Specified file <file> in <i>ufs</i> directory of EUFS
"EUFS:/datatx/<file>"	Specified file <file> in <i>datatx</i> directory of EUFS
	The <file> in values above indicates the exact file name.
<download_size>	Integer type. The size of the downloaded data. Unit: byte.
<checksum>	Integer type. The checksum of the downloaded data.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

NOTE

1. Inputting **+++** will make the TA end the data mode and switch to the AT command mode.
2. **<checksum>** is a 16-bit checksum based on bitwise XOR.

2.3.6. AT+QFOPEN Open a File

This command opens a file and gets the file handle to be used in commands such as **AT+QFREAD**, **AT+QFWRITE**, **AT+QFSEEK**, **AT+QFPOSITION**, **AT+QFTUCAT**, and **AT+QFCLOSE**.

AT+QFOPEN Open a File	
Test Command AT+QFOPEN=?	Response +QFOPEN: <filename>[(list of supported <mode>s)] OK
Read Command AT+QFOPEN?	Response If one or more files are opened. +QFOPEN: <filename>,<filehandle>,<mode> [+QFOPEN: <filename>,<filehandle>,<mode> [...]] OK If no file is opened. OK
Write Command AT+QFOPEN=<filename>[,<mode>]	Response +QFOPEN: <filehandle> OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filename>	String type. Name of the file to be opened. The maximum length of <filename> is 80 bytes. <div> <div>"<file>"</div> <div>Specified file <file> in <i>ufs</i> directory of UFS</div> </div> <div> <div>"UFS:<file>"</div> <div>Specified file <file> in <i>ufs</i> directory of UFS</div> </div> <div> <div>"UFS:/ufs/<file>"</div> <div>Specified file <file> in <i>ufs</i> directory of UFS</div> </div> <div> <div>"EUFS:<file>"</div> <div>Specified file <file> in <i>ufs</i> directory of EUFS</div> </div> <div> <div>"EUFS:/ufs/<file>"</div> <div>Specified file <file> in <i>ufs</i> directory of EUFS</div> </div> <div> <div>"EUFS:/datatx/<file>"</div> <div>Specified file <file> in <i>datatx</i> directory of EUFS</div> </div> <div>The <file> in values above indicates the exact file name.</div>
<filehandle>	Integer type. The handle of the file to be operated.
<mode>	Integer type. The file opening mode. <div> <div>0</div> <div>If the file exists, it will be opened with the file pointer set to the beginning of the file, otherwise, it will be created. In both cases, the file can be read and written to.</div> </div>

- 1 If the file exists, it will be overwritten, otherwise, it will be created. In both cases, the file can be read and written to.
- 2 If the file exists, it will be opened and read-only, otherwise, an error will be returned.
- 3 If the file exists, it will be opened with the file pointer positioned at the end of the current data, otherwise, it will be created. In both cases, the file can be read and written to.

<err> Integer type. Error code. See **Chapter 4** for possible **<err>** values.

2.3.7. AT+QFREAD Read a File

This command reads the data of a file which is specified by the file handle. The data starts from the current position of the file pointer which belongs to the file handle.

AT+QFREAD Read a File

Test Command AT+QFREAD=?	Response +QFREAD: <filehandle>[,<length>] OK
Write Command AT+QFREAD=<filehandle>[,<length>]	Response CONNECT <read_length> TA switches to the data mode. When the total size of the data reaches <length> or the TA receives +++ , the TA will return to the AT command mode, display the result and respond: OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filehandle>	Integer type. The handle of the file to be operated.
<length>	Integer type. The expected length of the file to be read and the default is the file length. Unit: byte.
<read_length>	Integer type. The actual read length. Unit: byte.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

2.3.8. AT+QFWRITE Write Data into a File

This command writes data into a file. The data starts from the current position of the file pointer which belongs to the file handle.

AT+QFWRITE Write a File	
Test Command AT+QFWRITE=?	Response +QFWRITE: <filehandle>[,<length>[,<timeout>]] OK
Write Command AT+QFWRITE=<filehandle>[,<length>[,<timeout>]]	Response CONNECT TA switches to the data mode. When the total size of the written data reaches <length> , the TA receives +++ , or the waiting time reaches <timeout> , the TA will return to the AT command mode and respond: +QFWRITE: <written_length>,<total_length> OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filehandle>	Integer type. The handle of the file to be operated.
<length>	Integer type. The expected written length. The maximum value of this parameter is determined by <free_size> of AT+QFUPL . Default: 10240. Unit: byte.
<timeout>	Integer type. The time waiting for data to be inputted to USB/UART. Range: 1–65535. Default: 5. Unit: s.
<written_length>	Integer type. The actual written length. Unit: byte.
<total_length>	Integer type. The total length of the file. Unit: byte.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

2.3.9. AT+QFSEEK Set File Pointer to a Specified Position

This command sets a file pointer to a specified position. This will decide the starting position of commands such as **AT+QFREAD**, **AT+QFWRITE**, **AT+QFPOSITION**, and **AT+QFTUCAT**.

AT+QFSEEK Set File Pointer to a Position

Test Command AT+QFSEEK=?	Response +QFSEEK: <filehandle>,<offset>[,<position>] OK
Write Command AT+QFSEEK=<filehandle>,<offset>[,<position>]	Response OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filehandle>	Integer type. The handle of the file to be operated.
<offset>	Integer type. The number of bytes of the file pointer movement.
<position>	Integer type. Pointer movement mode. <u>0</u> The beginning of the file. 1 The current position of the pointer. 2 The end of the file.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

NOTE

1. If **<position>=0** and **<offset>** exceeds the file size, the command returns **ERROR**.
2. If **<position>=1** and the total size of **<offset>** and the current position of the pointer exceed the file size, the command returns **ERROR**.
3. If **<position>=2**, the pointer moves backwards.

2.3.10. AT+QFPOSITION Get the Offset of a File Pointer

This command gets the offset of a file pointer from the beginning of the file.

AT+QFPOSITION Get the Offset of a File Pointer

Test Command AT+QFPOSITION=?	Response +QFPOSITION: <filehandle> OK
--	---

Write Command AT+QFPOSITION=<filehandle>	Response +QFPOSITION: <offset> OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filehandle>	Integer type. The handle of the file to be operated.
<offset>	Integer type. The offset from the beginning of the file.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

2.3.11. AT+QFTUCAT Truncate a File from the File Pointer

This command truncates all the data behind the position that the file pointer indicates.

AT+QFTUCAT Truncate a File from the File Pointer

Test Command AT+QFTUCAT=?	Response +QFTUCAT: <filehandle> OK
Write Command AT+QFTUCAT=<filehandle>	Response OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filehandle>	Integer type. The handle of the file to be operated.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

2.3.12. AT+QFCLOSE Close a File

This command closes a file and ends all operations to it. After that, the file handle is released and should not be used again, unless the file is re-opened with **AT+QFOPEN**.

AT+QFCLOSE Close a File	
Test Command AT+QFCLOSE=?	Response +QFCLOSE: <filehandle> OK
Write Command AT+QFCLOSE=<filehandle>	Response OK If there is any error +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filehandle>	Integer type. The handle of the file to be operated.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

2.3.13. AT+QFCRC Calculate the CRC of a Specified UFS File

This command calculates the CRC16, CRC16_CCITT, and CRC32 checksum of a specified UFS file.

AT+QFCRC Calculate the CRC of a Specified UFS File	
Test Command AT+QFCRC=?	Response +QFCRC: <filename> OK
Write Command AT+QFCRC=<filename>	Response When the specified file exists, the command returns 32-bit CRC, 16-bit CRC, and 16-bit CRC CCITT values: +QFCRC: <crc32>,<crc16>,<crc16_ccitt> OK If the file exists but has no byte: +QFCRC: 0,0,0

	If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<filename>	String type. The input format is "UFS:filename" where "UFS:" can be omitted.
<crc32>	The 32-bit CRC value.
<crc16>	The 16-bit CRC value.
<crc16_ccitt>	The 16-bit CRC CCITT value.
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

NOTE

This command is applicable to files in the *ufs* directory of UFS only.

2.3.14. AT+QFCPY Make a Copy of a Specified File

This command copies the entire content of a source file to a destination file.

- If the destination file already exists, the command will return an **ERROR**, unless **<enable>** is set to 1.
- If the destination file has the same name as the source file, the command will return an **ERROR**.
- If the remaining space on the file system is less than the size of the file to be copied, the command also will return an **ERROR**.

AT+QFCPY Make a Copy of a Specified File

Test Command AT+QFCPY=?	Response +QFCPY: <src_file>,<dest_file>[,<enable>] OK
Write Command AT+QFCPY=<src_file>,<dest_file>[,<enable>]	Response OK If there is any error: +CME ERROR: <err>
Maximum Response Time	300 ms
Characteristics	/

Parameter

<src_file>	String type. The source file name. The input format is "UFS:filename" where "UFS:" can be omitted.
<dest_file>	String type. The destination file name. The input format is "UFS:filename" where "UFS:" can be omitted.
<enable>	Integer type. Whether to overwrite the existing file which has the same name as the destination file. <u>0</u> Disable 1 Enable
<err>	Integer type. Error code. See Chapter 4 for possible <err> values.

NOTE

This command is applicable to files in the *ufs* directory of UFS only.

3 Examples

3.1. Upload and Download Files

3.1.1. Upload a File

3.1.1.1. Non ACK Mode

```

AT+QFUPL="test1.txt",10 //Upload the text file "test1.txt" to ufs directory of UFS.
CONNECT
<Input file bin data>
+QFUPL: 10,3938

OK
AT+QFUPL="EUFS:test.txt",4 //Upload the text file "test.txt" to ufs directory of EUFS.
CONNECT
<Input file bin data>
+QFUPL: 4,6a05

OK

```

3.1.1.2. ACK Mode

The ACK mode can make the data transmission more reliable. When transmitting a large file without hardware flow control, the ACK mode is recommended to be used to prevent the data from being lost. For more details about ACK mode, please refer to **AT+QFUPL** command.

```

AT+QFUPL="test.txt",3000,5,1 //Upload the text file "test.txt" to ufs directory of UFS.
CONNECT
<input file bin data of 1024bytes>
A //After receiving 1024 bytes data, the module returns A. And
then the next 1024 bytes data can be inputted.
<input file bin data of 1024bytes>
A
<input the rest file bin data>
+QFUPL: 3000,B34A

```

OK

3.1.2. Download a File

```
AT+QFDWL="test.txt"           //Download the text file "test.txt" from ufs directory of UFS.
CONNECT
<Output Data>
+QFDWL: 10,613e               //Get the bytes of the downloaded data and the checksum.

OK
```

3.2. Write and Read Files

3.2.1. Write and Read a UFS File

```
AT+QFOPEN="test",0           //Open the file to get the file handle.
+QFOPEN: 1

OK
AT+QFWRITE=1,10              //Write 10 bytes to the file.
CONNECT
<Write Data>
+QFWRITE: 10,10              //The actual bytes written and the size of the file are returned.

OK
AT+QFSEEK=1,0,0              //Set the file pointer to the beginning of the file.
OK
AT+QFREAD=1,10               //Read 10 bytes from the file.
CONNECT 10
<Read Data>

OK
AT+QFCLOSE=1                 //Close the file.
OK
```

3.2.2. Write and Read an EUFS File

```
AT+QFLDS="EUFSS"             //Query the space information of EUFS.
+QFLDS: 1388544,2435072

OK
```

```
AT+QFOPEN="EUFS:test",0           //Open the file to get the file handle.
+QFOPEN: 20000

OK
AT+QFWRITE=20000,10                //Write 10 bytes to the file.
CONNECT
<Write Data>
+QFWRITE: 10,10                    //The actual bytes written and the size of the file are returned.

OK
AT+QFSEEK=20000,0,0                //Set the file pointer to the beginning of the file.
OK
AT+QFREAD=20000,10                 //Read 10 bytes from the file.
CONNECT 10
<Read Data>

OK
AT+QFCLOSE=20000                   //Close the file.
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, and these error codes are only related to file operation of the modules.

Table 2: Summary of Error Codes

<err>	Meaning
400	Invalid input value
401	Larger than the size of the file
402	Read zero byte
403	Drive full
405	File not found
406	Invalid file name
407	File already existed
409	Fail to write the file
410	Fail to open the file
411	Fail to read the file
413	Reach the max number of file allowed to be opened
414	The file read-only
416	Invalid file descriptor
417	Fail to list the file
418	Fail to delete the file
419	Fail to get disk info

420	No space
421	Time out
423	File too large
425	Invalid parameter
426	File already opened

5 Appendix References

Table 3: Terms and Abbreviations

Abbreviation	Description
ACK	Acknowledgement
CCITT	Consultative Committee of International Telegraph and Telephone
COM	Communication Port
CRC	Cyclic Redundancy Check
DOS	Disk Operating System
EUFS	Extended User File Storage
ME	Mobile Equipment
TA	Terminal Adapter
UART	Universal Asynchronous Receiver/Transmitter
UFS	User File Storage
USB	Universal Serial Bus
XOR	Exclusive OR