Wireless Module Expert

UC20 Firmware Upgrade User Guide

UMTS/HSPA Module Series

Rev. UC20_Firmware_Upgrade_User_Guide_V1.0

Date: 2014-05-19

www.quectel.com

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

Quectel Wireless Solutions Co., Ltd.
Room 501, Building 13, No.99, Tianzhou Road, Shanghai, China, 200233
Tel: +86 21 5108 6236
Mail: info@quectel.com

Or our local office, for more information, please visit:
http://www.quectel.com/support/salesupport.aspx

For technical support, to report documentation errors, please visit:
http://www.quectel.com/support/techsupport.aspx

GENERAL NOTES

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

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2014-05-19	Arno WANG/Kent XU	Initial

Contents

About the Document **2**
Contents **3**
Table Index **4**
Figure Index **5**

1 Introduction **6**
1.1. Firmware Download Tool 6
1.2. Download Exception 6

2 Download **7**
2.1. Flow Chart 7
2.1.1. Enter PRG Download 8
2.1.2. Download PRG 9
2.1.3. Download Firmware 10
2.1.3.1. Send Partition File 10
2.1.3.2. Download Firmware 11
2.2. Packet Format 13

3 AT+QDL Introduction **15**

4 Example **16**

5 Appendix A Reference **17**

Table Index

TABLE 1: COMMAND PARAMETERS 16
TABLE 2: NO-OP COMMAND 17
TABLE 3: QUERY STATUS COMMAND 17
TABLE 4: QUERY STATUS RESPONSE 17
TABLE 5: ACK RESPONSE 17
TABLE 6: PARAMETERS REQUEST COMMAND 17
TABLE 7: PARAMETERS RESPONSE 18
TABLE 8: WRITE 32-BIT COMMAND 18
TABLE 9: GO COMMAND 18
TABLE 10: HELLO COMMAND 18
TABLE 11: HELLO RESPONSE 19
TABLE 12: SECURITY MODE 21
TABLE 13: SECURITY MODE RESPONSE 21
TABLE 14: PARTITION TABLE COMMAND 21
TABLE 15: PARTITION TABLE RESPONSE 21
TABLE 16: CLOSE COMMAND 22
TABLE 17: CLOSED COMMAND 22
TABLE 18: OPEN MULTI-IMAGE COMMAND 22
TABLE 19: OPENED MULTI-IMAGE COMMAND 22
TABLE 20: FIRMWARE WRITE COMMAND 23
TABLE 21: FIRMWARE WRITTEN COMMAND 23
TABLE 22: RESET COMMAND 23
TABLE 23: RESET ACK COMMAND 23

Figure Index

FIGURE 1: FLOW CHART 7
FIGURE 2: MODULE STATUS DETECTION 8
FIGURE 3: DOWNLOAD PRG 9
FIGURE 4: SEND PARTITION FILE 10
FIGURE 5: DOWNLOAD NEW FIRMWARE 12
FIGURE 6: PACKET FORMAT 14
FIGURE 7: UPGRADE IN LINUX OS 16

1 Introduction

UC20 module can be upgraded by USB Modem port, USB AT port, UART DM port or UART AT port. This document mainly introduces how to upgrade firmware of UC20 module by AT port, such as USB Modem port, USB AT port and UART AT port.

1.1. Firmware Download Tool

Quectel provides different OS tools, which can not only work on Windows OS, but also on embedded OS, such as Linux, Android and WinCE6.0. If you need these tools for further demands, please feel free to contact Quectel technical support.

If you need firmware package to upgrade, you'd better provide the version of your current module to us first, thus we can offer the related firmware package to you.

NOTE

Module's version can be obtained by sending "ATI" command through AT port.

1.2. Download Exception

When upgrading UC20, you may encounter some exceptions, such as power outage or accidental disconnection of USB/UART, which will cause failure of upgrade. After rebooting the module, it will enter download mode directly when it is rebooted. Hence before downloading, you should send "Query status command" to identify whether the module is in PRG status or AT command status. Please refer to Chapter 2.1.

2 Download

This section will introduce the download process in details, including packet format and the protocol.

2.1. Flow Chart

The following figure is a general description about how to upgrade firmware by AT port. For more detailed interactive process between host and module, please refer to Section 2.1.1, 2.1.2 and 2.1.3.

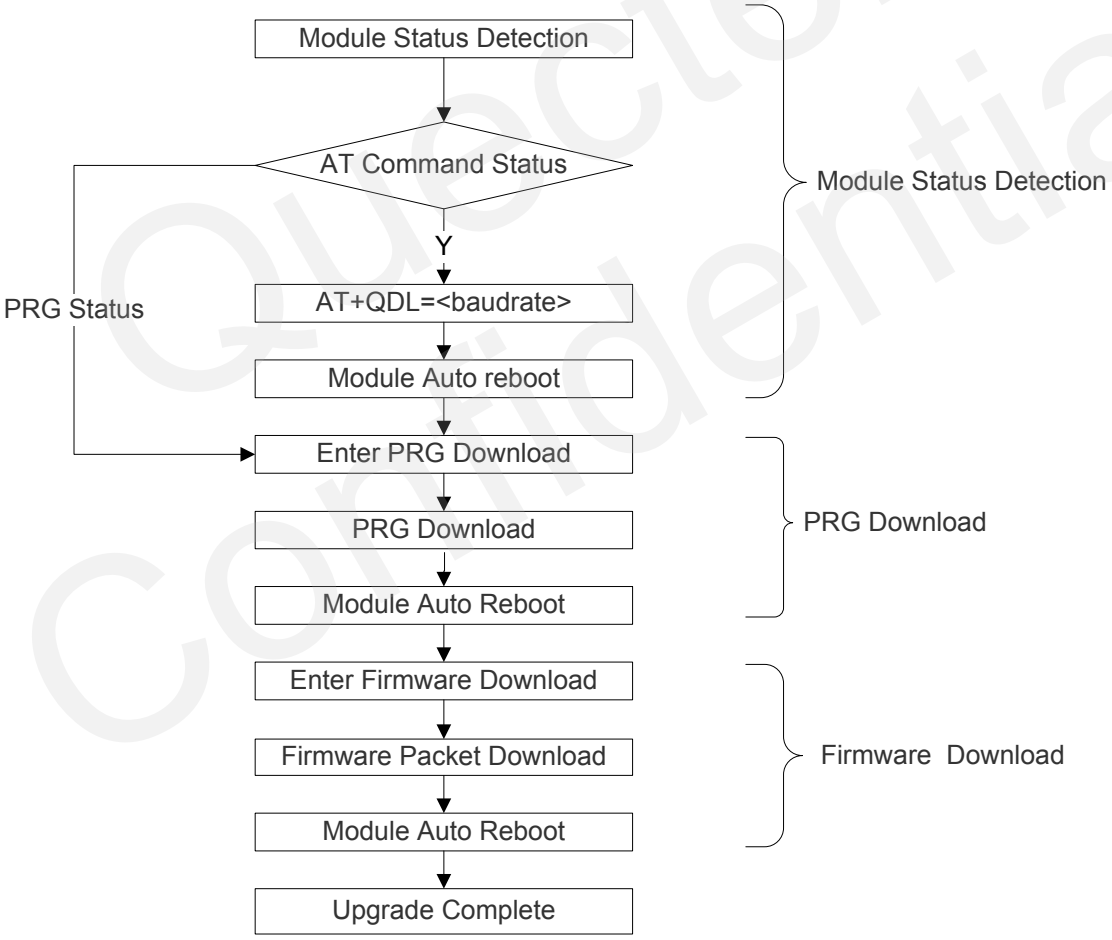

Figure 1: Flow Chart

2.1.1. Enter PRG Download

When module is powered on, it could be in two kinds of status:

- PRG Status
- AT Command Status

If the module is in PRG status, it will continue the upgrade process, host can also skip this section. If the module is in AT Command status, host must send “AT+QDL=<baudrate>” command to enter PRG status. The following figure gives a detailed description about it.

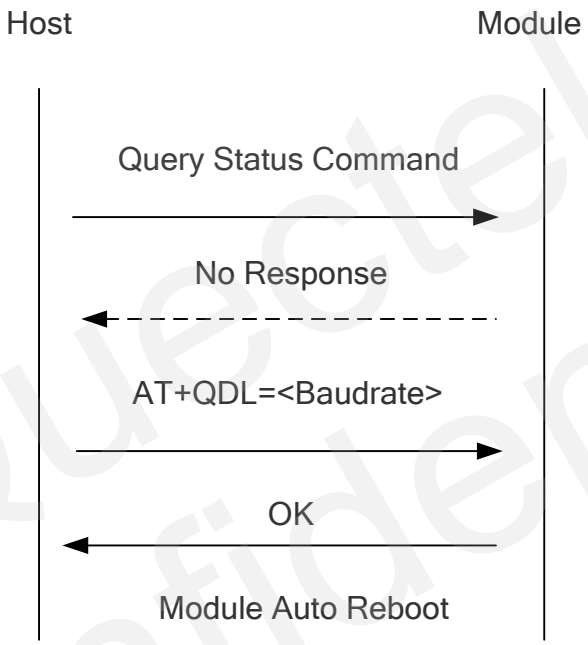

Figure 2: Module Status Detection

Step 1: Host sends “Query status command” to module to query module’s current status.

Step 2: Then host tries to read module’s response. Normally, module will return the response immediately (about 20ms), this indicates that module is in RPG status. But if host couldn’t get any response, it would read again. If it fails to read response for five times (about 100ms), this indicates the module is in AT Command status.

Step 3: If module is in “AT Command status”, host should send “AT+QDL=<baudrate>” to the module.

Step 4: If the host receives “OK” from the module, it indicates “AT+QDL” executes successfully, and then the module will automatically reboot and enter PRG status.

2.1.2. Download PRG

The following figure describes the detailed interactive process.

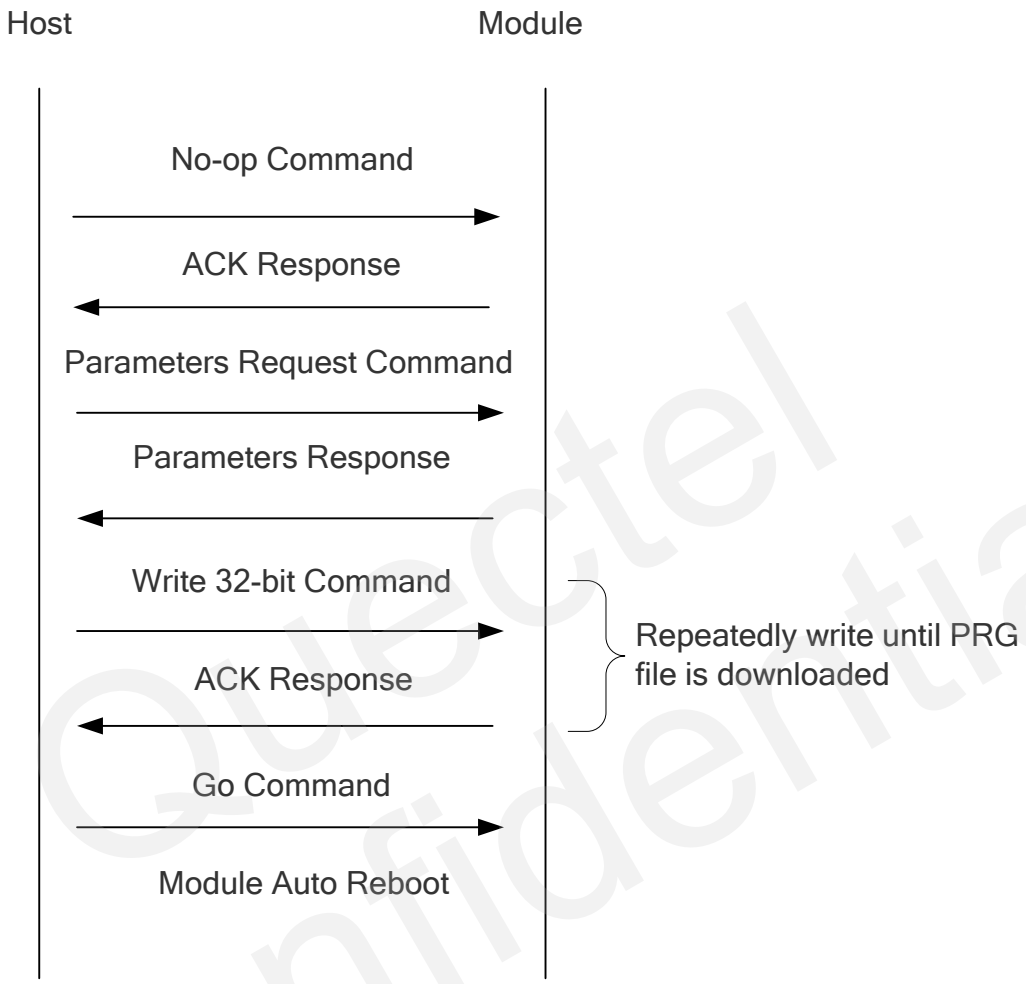

Figure 3: Download PRG

Step 1: Host sends “No-op command” to module.
Step 2: If module receives valid “No-op command”, then it will reply “ACK response”.
Step 3: If host receives “ACK response”, then the host should send “Parameters request command”.
Step 4: If module receives valid “Parameters request command”, the module will reply “Parameters response”.
Step 5: If host receives “Parameters response”, then the host should send the data from PRG file to module by “Write 32-bit command”.
Step 6: If module receives the data, then it will reply “ACK response”.
Step 7: If host receivers “ACK response”, then it will send “Go command” to module.

Finally, the module will automatically reboot and be ready to download firmware.

2.1.3. Download Firmware

After automatically rebooting the module, it will be ready to download firmware.

2.1.3.1. Send Partition File

The following figure describes the interactive process of sending partition file to the module by host.

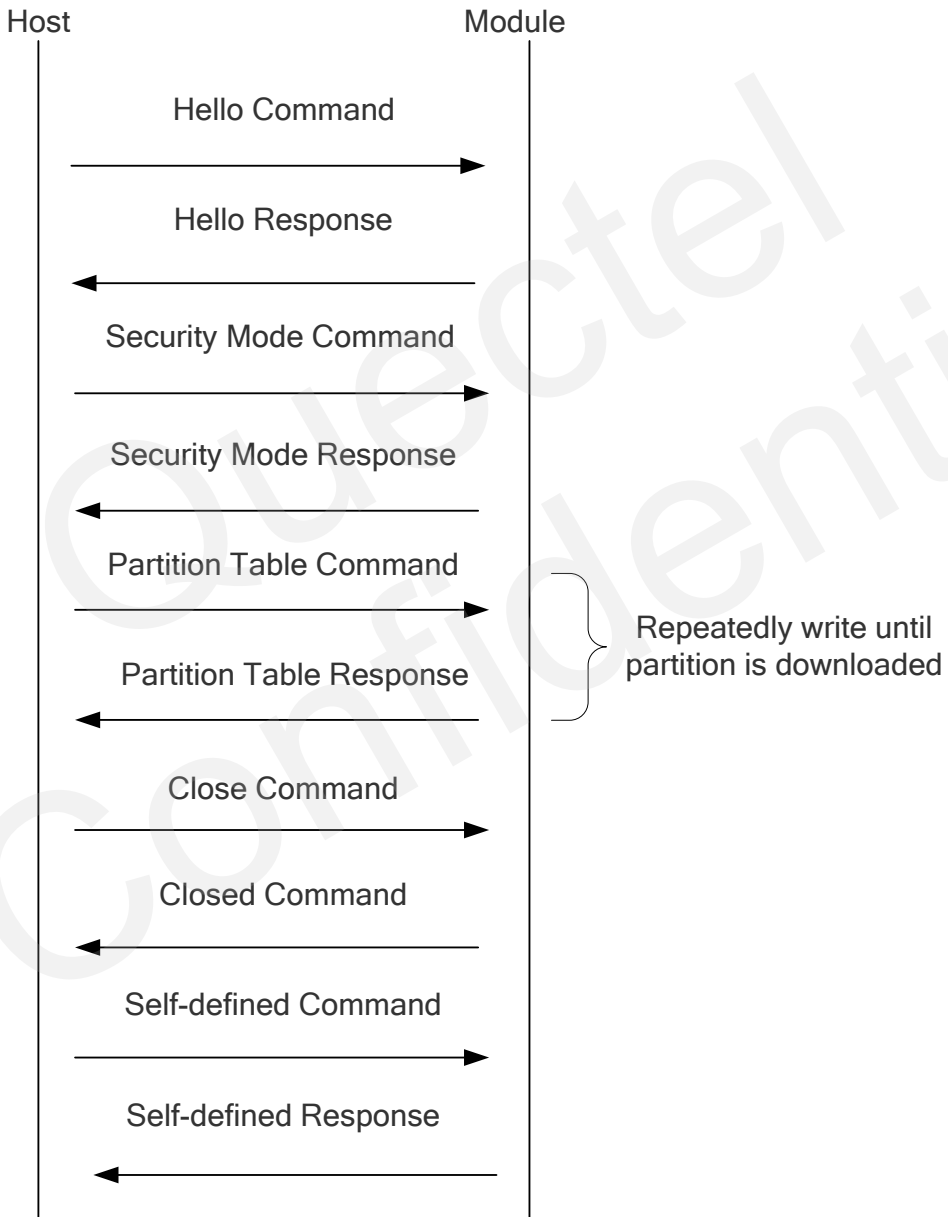

Figure 4: Send Partition File

Step 1: Host sends “Hello command” to module.
Step 2: If module receives “Hello command”, then it will reply “Hello response”.
Step 3: If host receives “Hello response”, then host should send “Security mode command”.
Step 4: If module receives “Hello response”, then it will reply “Security mode response” and enter security mode.
Step 5: If host receives “Security mode response”, then host should send data from “partition.mbn” file by “Partition Table command”.
Step 6: If module receives the data successfully, then it will reply “Partition Table response” packet.
Step 7: If host receives “Partition Table response”, then host should send “Close command”.
Step 8: If module receives “Close command”, then it will reply “Closed command”.
Step 9: If host receives “Closed command“, then host should send “Self-defined command”.
Step 10: If module receives “Self-defined command”, then it will reply “Self-defined response”.

2.1.3.2. Download Firmware

The following figure describes the interactive process about downloading “amss.mbn” file between host and module. When the upgrade process is completed, module will reboot and upgrade the firmware.

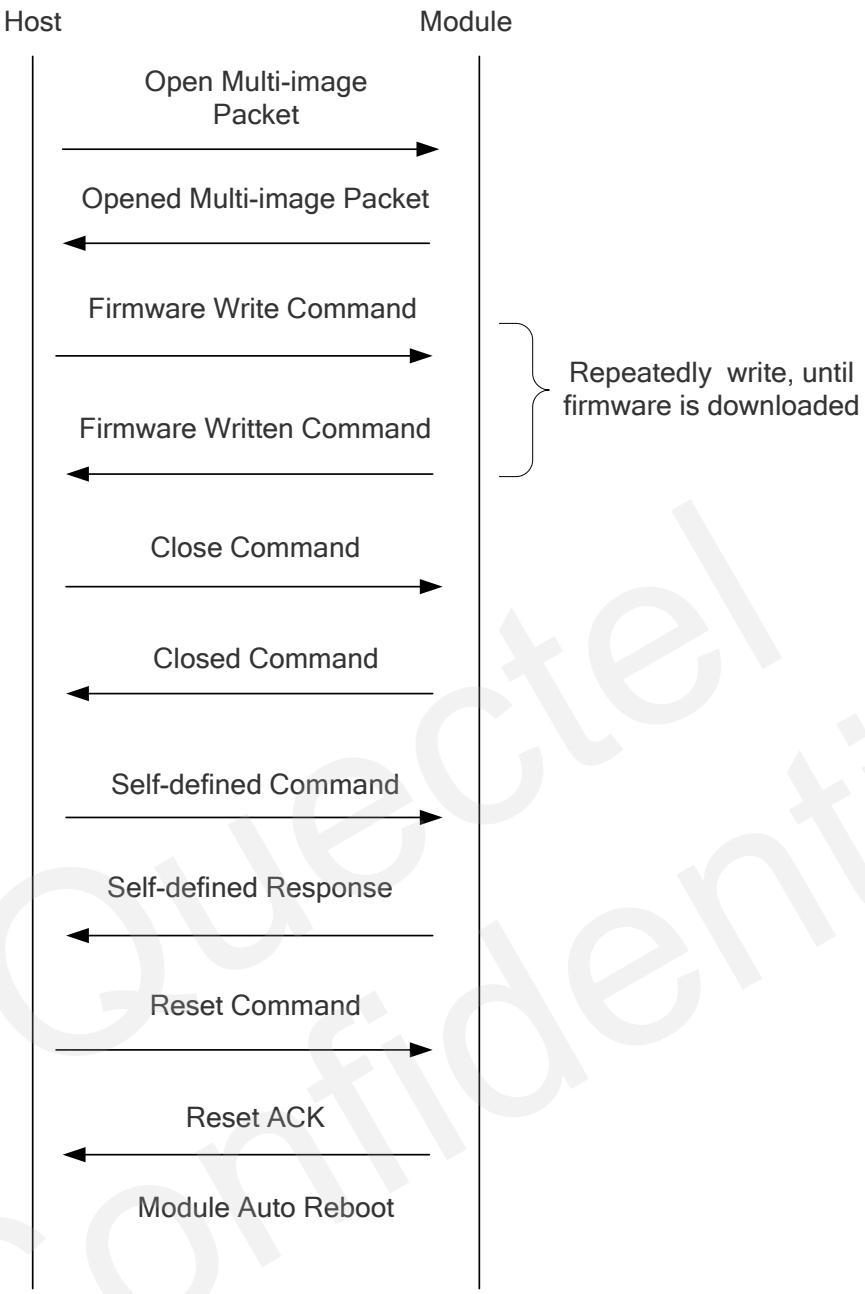

Figure 5: Download New Firmware

Step 1: Host sends “Open Multi-image command” to module.

Step 2: If module receives “Open Multi-image command”, then it will reply “Opened Multi-image response”.

Step 3: If host receives “Opened Multi-image response”, then host should send “Firmware Write command” to download the AMSS file.

Step 4: If module receives “Firmware Write command”, then it will reply “Firmware Written response”.

Step 5: If host receives “Firmware Written commend”, then host should send “Close command”.

Step 6: If module receives “Close command”, it will reply “Closed command”.

Step 7: If host receives “Closed command“, then host should send “Self-defined command”.

Step 8: If module receives “Self-defined command”, then it will reply “Self-defined response”.
Step 9: If host receives “Self-defined response”, then host should send “Reset command”.
Step 10: If module receives “Reset command”, then it will reply “Reset ACK”.

Finally, the module will automatically reboot and the whole upgrade process is completed.

2.2. Packet Format

The download protocol is compatible with Asyn-HDLC. About Async-HDLC, please refer to the Internet RFC-1331. All information transferred by this protocol is organized into packets which are contained in frames. In each frame, packets are composed of a header and body. The header contains control information, and the body contains the data. The body is simply the 8-bit data to be processed according to the command. The control information is small, just contains a command and its parameters.

NOTE

The Appendix A Reference lists all the command packets.

Information sent over the link is sent into packets contained in frames. All fields are multiple 8 bits, 1 octet. The commands and their parameters are detailed in the following section.

- The frame header is 7E (hexadecimal).
- The FCS is a 16-bit CRC.
- The flag octet is 7E (hexadecimal).
- The escape octet is 7D (hexadecimal).
- All multi-octet fields in the packet are transmitted with the most significant byte first. Most significant byte, followed by second most significant, and so on. For example, to transmit the value 256, the first sent byte would be 0x01, followed by 0x00.
- The CRC is transmitted with the least significant byte first.

The packet format is illustrated in following figure:

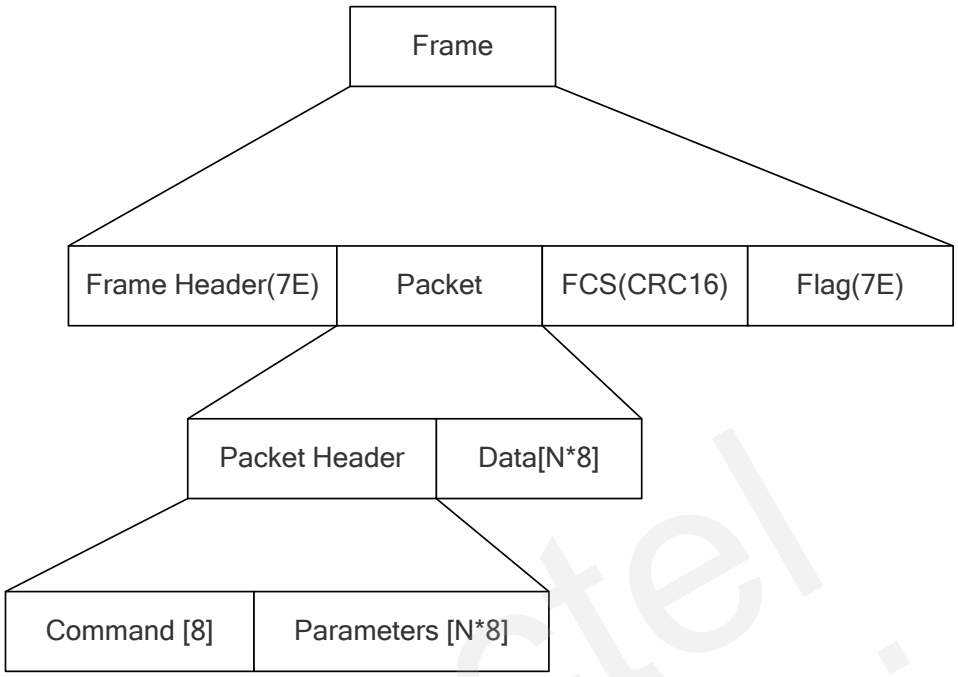

Figure 6: Packet Format

3 AT+QDL Introduction

Host sends “AT+QDL=<baudrate>” to module. When module replies “OK”, and “AT+QDL” command is executed successfully, module will automatically reboot and enter PRG status.

The “AT+QDL” command can set the baud rate of the serial port.

AT+QDL	
Set Command **AT+QDL=<baudrate>**	Response **OK**

NOTE

The baud rate setting is invalid under the USB. Currently, the supported baud rate includes 9600, 19200, 38400, 57600, 115200, 230400, 460800 and 921600.

4 Example

The example shows how to upgrade firmware in Linux, and the kernel version is 3.2.44.

The upgrade process is via ttyUSB2 port (AT port). The upgrade firmware file used is "UC20EQAR02A01V01M1024.bin" of which size is about 19MB.

Execute the following command in terminal:

```
./ UC20_linux_upgrade –f UC20EQAR02A01V01M1024.bin –p ttyUSB2
```

Table 1: Command Parameters

Parameters	Mandatory or Optional	Description
-f<firmware package filename>	Mandatory	<firmware package filename> is the name of the firmware package
-p<port>	Optional	<port> is the serial port

The following figure describes the whole upgrade process in Linux.

```
root@kent:~/workspace/update/Debug# ./update -f UC20EQAR02A01V01M1024.bin -p ttyUSB2
UC20 upgrade tool, Mon Aug 26 16:15:03 2013

Log name is UPGRADE20130826161503.log
File name is UC20EQAR02A01V01M1024.bin
Upgrade file version is UC20EQAR02A01V01M1024
Module Status Detection
In AT Command Status
Old version is  UC20EQAR02A01M1024
In PRG Status
Send NPRG6695.hex
progress : 100% finished

Start to firmware download
DBL downloading...
progress : 100% finished
FSBL downloading...
progress : 100% finished
OSBL downloading...
progress : 100% finished
AMSS downloading...
progress : 100% finished
restart...

UC20 upgraded successfully, Mon Aug 26 16:17:04 2013
```

Figure 7: Upgrade In Linux OS

5 Appendix A Reference

Table 2: No-op Command

Field	Length(bits)	Description
Command(0x06)	8	Command identifier code – The host shall set this field to 0x06.

Table 3: Query Status Command

Field	Length(bits)	Description
Command (0x0C)	1	Command identifier code – The host shall set this field to 0x0C

Table 4: Query Status Response

Field	Length(bits)	Description
Date	N	Include “OSBL” string – Normal download mode Include “PBL” string – Emerge download mode No response – Send AT+QDL=<baudrate>

Table 5: ACK Response

Field	Length(bits)	Description
Command(0x02)	8	Command identifier code – The host shall set this field to 0x02

Table 6: Parameters Request Command

Field	Length(bits)	Description
Command(0x07)	8	Command identifier code – The host shall set this field to 0x07

Table 7: Parameters Response

Field	Length(bits)	Description
Command(0x08)	8	Command identifier code – The host shall set this field to 0x08

Table 8: Write 32-bit Command

Field	Length(bits)	Description
Command(0x0F)	8	Command identifier code – The host shall set this field to 0x0F
Address	32	Write address – The host shall set this field to the 32-bit address of the first byte of the data to be written.
Length	16	Data length – The host shall set this field to the number of bytes of data to be written to memory.
Data	8*length	Write data – The host shall set this field to the data to be written to memory. The bytes shall be included in order of increasing memory address.

Table 9: Go Command

Field	Length(bits)	Description
Command(0x05)	8	Command identifier code – The host shall set this field to 0x05
Segment	16	Code segment address – The host shall set this field to the value to be loaded into the code segment register.
Offset	16	Code offset address – The host shall set this field to the value to be loaded into the instruction pointer register.

Table 10: Hello Command

Field	Octets	Description	Value Release 1.00
Command (0x01)	1	Command identifier code	Host sets this field to 0x01
Magic number (“fast download protocol	32	Protocol magic identifying number	Host sets this field to “fast download protocol host”

host")			
Version number	1	Version number of this protocol implementation	Host sets this field to indicate the maximum version of this protocol that the host supports; value for this field is 0x04
Compatible version	1	Lowest compatible version	Host sets this field to indicate the lowest version of the protocol that it supports; value for this field is 0x02
Feature bits	N	Data for the feature mask	Host sets these bits to indicate the negotiated set of features requested to be used

Table 11: Hello Response

Field	Octets	Description	Value
Command (0x02)	1	Command identifier code	Target sets this field to 0x02
Magic number ("fast download protocol targ")	32	Protocol magic identifying number	Target sets this field to "fast download protocol targ"
Version number	1	Version number of this protocol implementation	Target sets this field to indicate the maximum version of this protocol that the target supports
Compatible version	1	Lowest compatible version	Target sets this field to indicate the lowest version of the protocol that it supports
Maximum (preferred) block size	4	Maximum data size acceptable for a write command	Target sets this field to indicate the size of the largest data payload in a stream write command that is acceptable; host is responsible to attempt to send the largest commands possible (to reduce overhead) while being limited by this value
Base address of	4	Lowest address that is mapped into	Target uses this field to

Flash		the Flash part	report the location of the Flash part
Flash ID length	1	Length of the Flash identifier string	—
Flash identifier	N*Flash ID Length	String identifying the Flash device	Target is free to use this string for informational purposes, but the host should not attempt to control the flow of execution of code based on this string; target always returns a nonzero length ASCII string in this field
Window size	2	Sliding window size	Target sets this field to the number of unprocessed commands that can be accommodated in the command buffer; host is responsible to not overrun this limit; if the host does overrun this limit, target is free to ignore any commands sent during the overrun
Number of sectors	2	Number of erase sectors in the part	Target sets this field to report the number of sectors in the identified Flash part
Sector sizes	4*number of sectors	Size of each sector	For every sector reported in the previous field, the target indicates the size of each sector; list is in order from low address at end of the part to high address at end of the part
Feature bits	N	Data for the feature mask	Target sets these bits to indicate the negotiated set of features that are used

Table 12: Security Mode

Field	Octets	Description	Value
Command (0x17)	1	Command identifier code	Host sets this field to 0x17
Mode	1	Security mode	0x0 – Nontrusted 0x1 – Trusted

Table 13: Security Mode Response

Field	Octets	Description	Value
Command (0x18)	1	Command identifier code	Target sets this field to 0x18

Table 14: Partition Table Command

Field	Octets	Description	Value
Command (0x19)	1	Command identifier code	Host sets this field to 0x19
Override	1	Override existing table parameter	0x0 – No override 0x1 – Override existing table
Partition table	N	Data payload	Partition table to be used (maximum length 512 bytes)

Table 15: Partition Table response

Field	Octets	Description	Value
Command (0x1A)	1	Command identifier code	Target sets this field to 0x1A
Status	1	Status value	0x0 – Partition table accepted 0x1 – Partition table differs, override is accepted 0x2 – Partition table format is not recognized, and does not accept override 0x3 – Erase operation

			failed

Table 16: Close Command

Field	Octets	Description	Value
Command (0x15)	1	Command identifier code	Host sets this field to 0x15

Table 17: Closed Command

Field	Octets	Description	Value
Command (0x16)	1	Command identifier code	Target sets this field to 0x16

Table 18: Open Multi-image Command

Field	Octets	Description	Value
Command (0x1B)	1	Command identifier code	Host sets this field to 0x1b
Type	1	Which Image mode to open in	Refer to *Streaming_DLoad_Protocol.pdf*
Data	N	Data payload	Refer to *Streaming_DLoad_Protocol.pdf*

Table 19: Opened Multi-image Command

Field	Octets	Description	Value
Command (0x1C)	1	Command identifier code	Target sets this field to 0x1C
Status	1	Status value	0x0 – Open successful 0x1 – Payload length exceeded, failed 0x2 – No payload expected, failed 0x3 – Payload required, fail 0x4 – Block 0 write

	protected, failed

Table 20: Firmware Write Command

Field	Octets	Description	Value
Command (0x07)	1	Command identifier code	Host sets this field to 0x07
Address	4	Data address	Host sets this to the address from where to start the write
Data	N	Data payload	Data to be written

Table 21: Firmware Written Command

Field	octets	Description	Value
Command (0x08)	1	Command identifier code	Target sets this field to 0x08
Address	4	Data address	Target sets this to the address at where the data was written
Data	N	Data payload	Data to be written

Table 22: Reset Command

Field	Octets	Description	Value
Command (0x0B)	1	Command identifier code	Host sets this field to 0x0B

Table 23: Reset ACK Command

Field	Octets	Description	Value
Command (0x0C)	1	Command identifier code	Host sets this field to 0x0C