



GSM/GPRS/GPS Tracker **GV200**

@Tracker Air Interface Firmware Update

Revision: 1.00



Contents

Contents	1
0. Revision History	2
1. Scope.....	3
2. Message.....	4
2.1. Command And Acknowledgement	4
2.1.1. Start the firmware update	4
2.1.2. Stop the firmware update	5
2.1.3. Acknowledgement.....	5
2.2. Report.....	7
2.2.1. Update Confirmation.....	7
2.2.2. Package Downloading.....	7
2.2.3. Firmware Updating	8
3. Firmware Update Process	9
3.1. Initiation of the update process	9
3.2. Confirmation of the update process.....	9
3.3. Downloading of the update package	9
3.4. Updating of the firmware	10
3.5. An example of successful updating	11

Queclink
Confidential

0. Revision History

Revision	Date	Author	Description of change
1.00	2011-03-03	Hendry PAN	Initial

Queclink
Confidential

1. Scope

This document describes the firmware update over the air for GV200. This enables to update the firmware of GV200 remotely without having to require the end users to bring their device to the service centre. Thus the service provider of GV200 could conveniently push new features or bug fix to the end users and promote the customer experience.

During the firmware updating, three equipments are involved:

- ✧ the terminal: GV200 whose firmware is to be updated
- ✧ the backend server: the server which remotely controls the terminal and receives report from the terminal
- ✧ the file server: the server which hosts the packages of the update

The file server and the backend server could host on the same machine.

This document describes the process of the firmware update and the necessary message exchanged during the updating, while below information are not covered:

- ✧ The timing and the strategy that the backend server to initiate the updating.
- ✧ The deployment method of the update package.
- ✧ How to setup a file server.
- ✧ The communication between the backend server and the file server.

2. Message

2.1. Command And Acknowledgement

The command **AT+GTUPD** is used to start and stop the firmware update remotely.

2.1.1. Start the firmware update

To start the firmware update, the backend server sends **AT+GTUPD (sub:0)** command to the device. Upon this command, the device is informed of where to download the update package and how to download the package.

➤ **Start: AT+GTUPD=**

Example:			
AT+GTUPD=gv200,0,0,10,0,,http://fota.queclink.com/gv200_0201_0205.bin,,,,,0001\$			
Parameter	Length(byte)	Range/Format	Default value
password	4~6	'0'-'9','a'-'z','A'-'Z'	Gv200
sub-command	1	0	
max download retry	1	0 – 3	0
download timeout	2	10 – 30 min	10
download protocol	1	0	0
download user name	<=6	'0'-'9','a'-'z','A'-'Z'	
download password	<=6	'0'-'9','a'-'z','A'-'Z'	
download URL	100	legal URL	
reserved	0		
reserved	0		
reserved	0		
reserved	0		
serial number	4	0000-FFFF	
tail character	1	\$	\$

- ✧ <password>: the valid character of password is '0'-'9', 'a'-'z', 'A'-'Z'. The default value is "gv200".
- ✧ <sub-command>: sub-command of AT+GTUPD, 0 means to start the firmware update.
- ✧ <max download retry>: specifies the maximum time of retrying to download the update package upon downloading failure.
- ✧ <download timeout>: specifies the expiration timeout of one single downloading. If the downloading expires, it is considered to be failure.
- ✧ <download protocol>: the protocol used to download the package. Only HTTP is supported now. Set to 0.
- ✧ <download user name>: if the file server uses authentication, specifies the user name here.
- ✧ <download password>: if the file server uses authentication, specifies the password here.

- ✧ <download URL>: specifies the URL to download the package.
- ✧ <reserved>: reserved for future extension.
- ✧ <serial number>: As the command reference, the exact serial number will be sent back to the platform in ACK. It is in hexadecimal format. It should begin from 0000 and increases by 1 every time. It should roll back after “FFFF”.
- ✧ <tail character>: A character to indicate the end of the command. Must be “\$”.

2.1.2. Stop the firmware update

Before the device finishes downloading the update package, the backend server could use the **AT+GTUPD (sub:1)** command to cancel the current firmware updating. If the package is downloaded successful, this command is ignored.

➤ **Stop: AT+GTUPD=**

Example: AT+GTUPD=gv200,1,,,,,0001\$			
Parameter	Length(byte)	Range/Format	Default value
password	4~6	'0'-'9','a'-'z','A'-'Z'	gv200
sub-command	1	1	
reserved	0		
reserved	0		
reserved	0		
reserved	0		
serial number	4	0000-FFFF	
tail character	1	\$	\$

- ✧ <sub-command>: sub-command of AT+GTUPD, 1 means to cancel the current firmware update process.

2.1.3. Acknowledgement

The acknowledgement message of **AT+GTUPD** command:

➤ **+ACK:GTUPD,**

Example: +ACK:GTUPD,040100,135790246811220,,0001,20090101000002,11F0\$			
Parameter	Length(byte)	Range/Format	Default
protocol version	6	XX0000 – XXFFFF, X ∈ {'A' – 'Z', '0' – '9'}	
unique ID	15	IMEI	
device name	<=10	'0'-'9','a'-'z','A'-'Z'	
serial number	4	0000-FFFF	
send time	14	YYYYMMDDHHMMSS	
count number	4	0000-FFFF	
tail character	1	\$	\$

- ✧ <Protocol Version>: The protocol version that the terminal conforms to. The first two

characters XX point out the device type. **04** means GV200. The middle two characters point out the main version number and the last two characters point out the minimum version number. And both of the main version and the minimum version are hex digital. For example, **020A** means version 2.10

- ✧ *<unique ID>*: The terminal's IMEI.
- ✧ *<device name>*: An ASCII string for the name of the device.
- ✧ *<serial number>*: The *<serial number>* in the **AT+GTUPD** command.
- ✧ *<send time>*: The terminal local time to send the message.
- ✧ *<count number>*: The self-increasing count number will be put into every acknowledgment message. The count is beginning from 0000 and increases by 1 every time. It will roll back after "FFFF".

Queclink
Confidential

2.2. Report

During the firmware updating process, the device reports its status to the backend server by message **+RESP:GTUPD** upon different phases, including the update confirmation information, package downloading information and firmware updating information.

2.2.1. Update Confirmation

The device sends update confirmation information to the backend server if:

- ✧ the update command is confirmed by the device.
- ✧ the update command is refused by the device.
- ✧ the update process is canceled by the backend server.
- ✧ the update command is refused because the battery is low.

➤ Confirmation: **+RESP:GTUPD**,

Example: +RESP:GTUPD,040100,135790246811220,,100,,20090101000000,11F0\$			
Parameter	Length(byte)	Range/Format	Default
protocol version	6	XX0000 – XXXFFF, X ∈ {'A' – 'Z', '0' – '9'}	
unique ID	15	IMEI	
device name	<=10	'0'-'9','a'-'z','A'-'Z'	
code	3	100 101 102 103	
reserved	0		
send time	14	YYYYMMDDHHMMSS	
count number	4	0000-FFFF	
tail character	1	\$	\$

- ✧ `<code>`: the confirmation information
 - 100: the update command is confirmed by the device.
 - 101: the update command is refused by the device.
 - 102: the update process is canceled by the backend server.
 - 103: the update process is refused because the battery is low.

2.2.2. Package Downloading

The device sends package downloading information to the backend server if:

- ✧ the device starts to download the package
- ✧ the device finishes downloading the package successfully
- ✧ the device fails to download the package

➤ Downloading: **+RESP:GTUPD**,

Example:

+RESP:GTUPD,040100,135790246811220,,200,1,20090101000000,11F0\$			
Parameter	Length(byte)	Range/Format	Default
protocol version	6	XX0000 – XXFFFF, X ∈ {'A' – 'Z', '0' – '9'}	
unique ID	15	IMEI	
device name	<=10	'0'-'9','a'-'z','A'-'Z'	
code	3	200 201 202	
download times	1	1 2 3 4	
send time	14	YYYYMMDDHHMMSS	
count number	4	0000-FFFF	
tail character	1	\$	\$

- ✧ <code>: the downloading information
 - 200: the device starts to download the package
 - 201: the device finishes downloading the package successfully
 - 202: the device fails to download the package
- ✧ <download times>: the count number of the package downloading

2.2.3. Firmware Updating

The device sends firmware updating information to the backend server if:

- ✧ the device starts to update the firmware
- ✧ the device finishes update the firmware successfully
- ✧ the device fails to update the firmware
- ✧ the update process does not start because the battery is low

➤ Updating: +RESP:GTUPD,

Example:			
+RESP:GTUPD,040100,135790246811220,,300,,20090101000000,11F0\$			
Parameter	Length(byte)	Range/Format	Default
protocol version	6	XX0000 – XXFFFF, X ∈ {'A' – 'Z', '0' – '9'}	
unique ID	15	IMEI	
device name	<=10	'0'-'9','a'-'z','A'-'Z'	
code	3	300 301 302 303	
reserved	0		
send time	14	YYYYMMDDHHMMSS	
count number	4	0000-FFFF	
tail character	1	\$	\$

- ✧ <code>: the updating information
 - 300: the device starts to update the firmware
 - 301: the device finishes updating the firmware successfully
 - 302: the device fails to update the firmware
 - 303: the update process does not start because the battery is low

3. Firmware Update Process

The firmware update process includes four steps.

3.1. Initiation of the update process

The backend server sends the **AT+GTUPD (sub:0)** command to the device to initiate the update process. Along with this command, the backend server sends necessary information for the device to start the update process.

It's the backend server's duty to decide when and how to initiate the firmware update process to all the devices it controls. As the response messages collector and the controller, the backend server has all the information it needs to start an update process including the current firmware versions of the devices it controls (retrieved with the **AT+GTRTO** command), the version of the latest available firmware and the location of the proper update packages.

3.2. Confirmation of the update process

Upon the **AT+GTUPD (sub:0)** command, the device will first check the current battery capacity. If the battery capacity can not support the update process, it will report **+RESP:GTUPD (code: 103)** to notify the backend server that the update process is to be aborted because of low battery. If the battery capacity is ample, the device will send **+RESP:GTUPD** with confirmation information to the backend server. Then the update process proceeds to the next step.

If the update command is confirmed, the device will turn into a non-interactive mode. That is, the end user can no longer make phone call, all incoming call are rejected automatically until the update process finishes. At the meantime, the device will ignore all the command received from the backend server if it is not related to the update process. Also the device will stop all the reports that are not related to the update process.

3.3. Downloading of the update package

If the update command is confirmed, the device will use the information sent by the backend server to download the update package. If the downloading fails, it will retry the specified times. If all attempts fail, the updating process is aborted and the device will automatically reboot to go back to the normal working mode. If the downloading successes, the update process proceeds to the next step. Either way, the device will send **+RESP:GTUPD** with downloading information to the backend server.

Before the package is downloaded, the backend server could send **AT+GTUPD (sub:1)** command to cancel the current update process. This is the only chance to abort during the update process.

3.4. Updating of the firmware

After downloading the package successfully, the device will check the battery capacity again. If the battery can not support the update process, the device will report **+RESP:GTUPD (code: 303)** to notify the backend server that the update process is to be aborted because of low battery. If the battery capacity is ample, the device will send **+RESP:GTUPD (code:300)** to the backend server to inform of the start of the updating. Then it uses the update package to update the firmware. After the updating, success or failure, the device will reboot automatically. After the device boots up, it sends **+RESP:GTUPD** with updating information to the backend server and works as usual.

Queclink
Confidential

3.5. An example of successful updating

