L26-DR&L26-P&L26-T&L89&LC98S

Firmware Upgrade Guide

GNSS Module Series

Version: 1.0

Date: 2020-09-12

Status: Released

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

Quectel Wireless Solutions Co., Ltd.
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 office. For more information, please visit: http://www.quectel.com/support/sales.htm.

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/technical.htm or email to support@quectel.com.

GENERAL NOTES

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

DISCLAIMER

WHILE QUECTEL HAS MADE EFFORTS TO ENSURE THAT THE FUNCTIONS AND FEATURES UNDER DEVELOPMENT ARE FREE FROM ERRORS, IT IS POSSIBLE THAT THESE FUNCTIONS AND FEATURES COULD CONTAIN ERRORS, INACCURACIES AND OMISSIONS. UNLESS OTHERWISE PROVIDED BY VALID AGREEMENT, QUECTEL MAKES NO WARRANTIES OF ANY KIND, IMPLIED OR EXPRESS, WITH RESPECT TO THE USE OF FEATURES AND FUNCTIONS UNDER DEVELOPMENT. TO THE MAXIMUM EXTENT PERMITTED BY LAW, QUECTEL EXCLUDES ALL LIABILITY FOR ANY LOSS OR DAMAGE SUFFERED IN CONNECTION WITH THE USE OF THE FUNCTIONS AND FEATURES UNDER DEVELOPMENT, REGARDLESS OF WHETHER SUCH LOSS OR DAMAGE MAY HAVE BEEN FORESEEABLE.

COPYRIGHT

About the Document

Revision History

Version	Date	Author	Description
1.0	2020-09-12	Berton PENG/ Ai HONG	Initial

Contents

Table Index

Figure Index

1 Introduction

This document list the needed steps required to apply the firmware upgrade function for the following Quectel GNSS modules.

- L26-DR
- L26-P
- L26-T
- L89
- LC98S

Following the procedures after the module is connected to a host, users will upload a new firmware to these GNSS modules over UART. The connection is illustrated by the figure below.

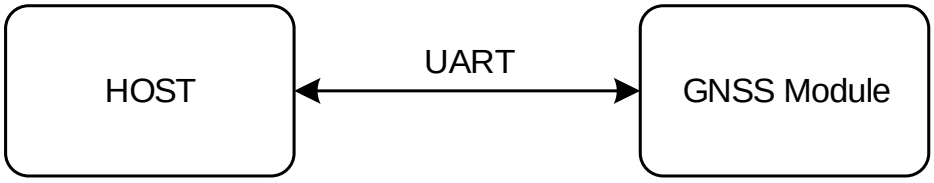

Figure 1: Firmware Upgrade Connection

2 Upgrade Procedure

2.1. Overview

2.1.1. Firmware Upgrade in Normal Mode

Please refer to the flowchart of upgrade in normal mode below.

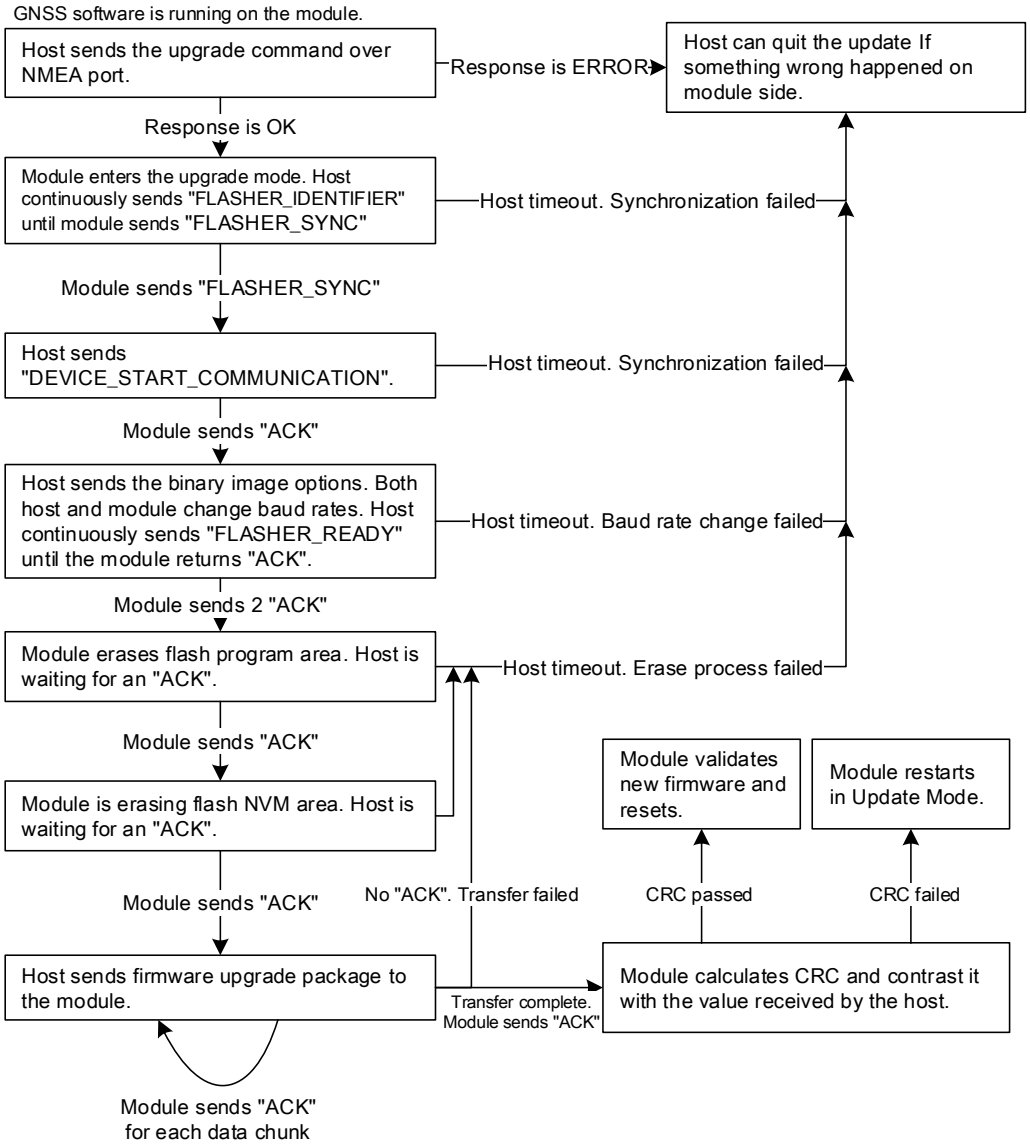

Figure 2: Firmware Upgrade in Normal Mode

2.1.2. Firmware Upgrade in Recovery Mode

Recovery mode is used to upgrade the firmware without using UART interface. Upgrade starts with host continuously sending "FLASHER_IDENTIFIER" word while the module is under reset; when the module exits from reset, it starts synchronization and sends "FLASHER_SYNC" word when ready. From now on, upgrade proceeds exactly like normal mode. A flowchart of upgrade in recovery mode is shown below.

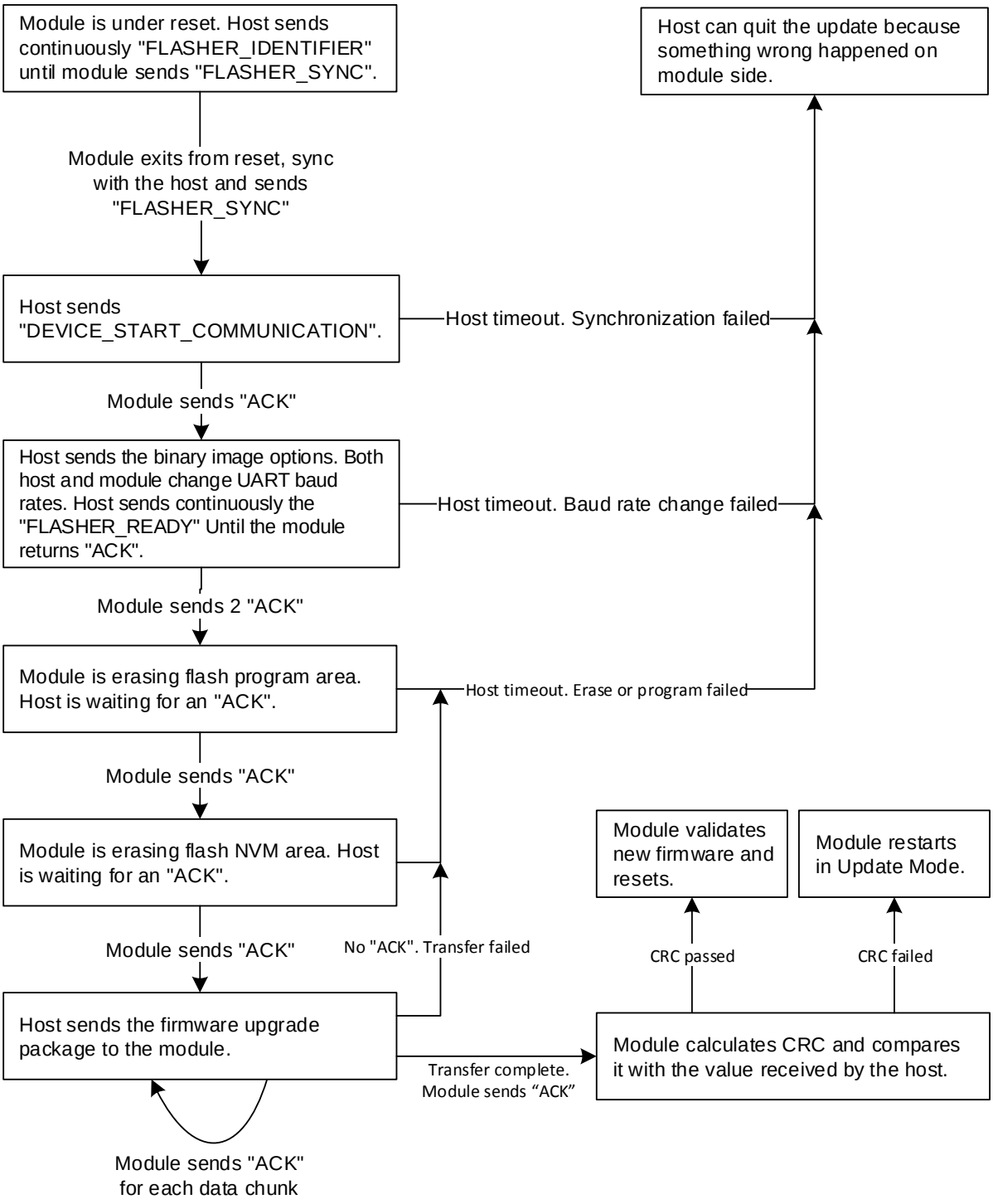

Figure 3: Firmware Upgrade in Recovery Mode

2.2. Procedure

The following subsections list the upgrade procedure steps.

All the constants used in this document are listed in the following table.

Table 1: Constants List

Constants Name	Constants Value
FLASHER_IDENTIFIER	0xBCD501F4
FLASHER_SYNC	0x83984073
DEVICE_START_COMMUNICATION	0xA3
FLASHER_READY	0x4A
ACK	0xCC
NAK	0xDD
NVM_FLASH_OFFSET	0x00100000
NVM_FLASH_ERASE_SIZE	0x00100000

NOTE

Multibyte constants are sent/received in little-endian mode.

2.2.1. Send FW Upgrade Command

After the module is connected to a host, the FW upgrade command **$PSTMFWUPGRADE** starts the firmware upgrade procedure. The FW upgrade command is described as follows:

Synopsis:

```
$PSTMFWUPGRADE*<checksum><CR><LF>
```

Arguments:

Parameter	Format	Description
checksum	Hexadecimal, 2 digits	Checksum of the message bytes between but not including the “$” and “*” characters

Results:

- If successful, the module responds with the following message, indicating that the module invalidates current firmware and switches into Firmware Upgrade Mode until a new firmware is downloaded and validated:

  ```
  $PSTMFWUPGRADEOK*<checksum><CR><LF>
  ```

- In case of error, the following message will be returned by the module, indicating that the host can quit the upgrade when error occurs on the module side:

  ```
  $PSTMFWUPGRADEERROR*<checksum><CR><LF>
  ```

NOTES

1. Commands are transmitted in string format via UART.
2. The string ***<checksum>** is optional when users input commands.

2.2.2. Synchronize

As soon as the module receives the FW upgrade command, GNSS firmware will reset the hardware and enter Firmware Upgrade Mode. Then, the software running on the host can start to synchronize with the module. In this phase the host continuously sends “FLASHER_IDENTIFIER” word; at the same time, it checks for a response from the module. In order to complete synchronization phase, the module sends back the “FLASHER_SYNC” word.

Host software uses a timeout of 3 seconds. If no response is received in this time interval, the host software will return with timeout error.

As soon as the host receives “FLASHER_SYNC”, it will send “DEVICE_START_COMMUNICATION” word and wait for "ACK" response from the module.

NOTE

The baud rate of the host should be set with the same baud rate that will be configured in the binary image options for the module.

2.2.3. Write Binary Image Options

Just after the synchronization with the module, the host must send the binary image options, which are packed inside a structure shown below:

```
//Binary image options
struct ImageOptions
{
byte eraseNVM;
byte programOnly;
byte reserved;
byte baudRate;
int firmwareSize;
uint firmwareCRC;
int nvmAddressOffset;
int nvmSize;
};
```

Where:

Field	Type	Description
eraseNVM	byte	The possible value is true or false. If true, the NVM flash memory area will be erased during upgrade.
programOnly	byte	The possible value is true or false. If true, firmware will be flashed without erasing flash memory area.
reserved	byte	Must be 0
baudRate	byte	The new UART baud rate used to download the firmware. Below there are the possible values: 0 = 57600 bps 1 = 115200 bps 2 = 230400 bps 3 = 460800 bps 4 = 921600 bps
firmwareSize	int	The file size of firmware binary image.
firmwareCRC	uint	The CRC32 code calculated first on the field of image file size (in little-endian format) and then over entire firmware image payload.
nvmAddressOffset	int	Set to "NVM_FLASH_OFFSET"
nvmSize	int	Set to "NVM_FLASH_ERASE_SIZE"

2.2.4. Erase Program Area and Flash NVM Area

If ***programOnly*** field is set to false, the host software must wait for an "ACK" response from the module as confirmation that flash program area has been erased.

If ***eraseNVM*** field is set to true, the host software will wait for an "ACK" response from the module as confirmation that the flash NVM area has been erased.

In either case the host software can use a timeout of 30 seconds; if no response is received in this time interval, host software can return with timeout error.

2.2.5. Send Firmware Package

At this point the host will send the firmware package to the module for the upgrade inside the module's flash. The software running on the host must split the firmware package (as binary image data) into n chunks of 16 KB and size of the last chunk must be equal to the remaining bytes number. Each data chunk will be acknowledged with "ACK" response from the module.

2.2.6. CRC

At this point the module bootloader software performs a CRC error check on the image data received by the host, and if the check is passed, an "ACK" response will be sent back to the host and the new downloaded firmware will be regarded as validated. Otherwise, if the check failed, an "NAK" response will be sent. In both cases the module resets itself. The code for CRC32 algorithm is shown as follows.

```
//CRC32 algorithm.
const uint32_t crc32_tab[] =
{
    0x00000000, 0x77073096, 0xee0e612c, 0x990951ba, 0x076dc419,
    0x706af48f, 0xe963a535, 0x9e6495a3, 0x0edb8832, 0x79dcb8a4,
    0xe0d5e91e, 0x97d2d988, 0x09b64c2b, 0x7eb17cbd, 0xe7b82d07,
    0x90bf1d91, 0x1db71064, 0x6ab020f2, 0xf3b97148, 0x84be41de,
    0x1adad47d, 0x6ddde4eb, 0xf4d4b551, 0x83d385c7, 0x136c9856,
    0x646ba8c0, 0xfd62f97a, 0x8a65c9ec, 0x14015c4f, 0x63066cd9,
    0xfa0f3d63, 0x8d080df5, 0x3b6e20c8, 0x4c69105e, 0xd56041e4,
    0xa2677172, 0x3c03e4d1, 0x4b04d447, 0xd20d85fd, 0xa50ab56b,
    0x35b5a8fa, 0x42b2986c, 0xdbbbc9d6, 0xacbcf940, 0x32d86ce3,
    0x45df5c75, 0xdcd60dcf, 0xabd13d59, 0x26d930ac, 0x51de003a,
    0xc8d75180, 0xbfd06116, 0x21b4f4b5, 0x56b3c423, 0xcfba9599,
    0xb8bda50f, 0x2802b89e, 0x5f058808, 0xc60cd9b2, 0xb10be924,
    0x2f6f7c87, 0x58684c11, 0xc1611dab, 0xb6662d3d, 0x76dc4190,
    0x01db7106, 0x98d220bc, 0xefd5102a, 0x71b18589, 0x06b6b51f,
    0x9fbfe4a5, 0xe8b8d433, 0x7807c9a2, 0x0f00f934, 0x9609a88e,
```

```
    0xe10e9818, 0x7f6a0dbb, 0x086d3d2d, 0x91646c97, 0xe6635c01,
    0x6b6b51f4, 0x1c6c6162, 0x856530d8, 0xf262004e, 0x6c0695ed,
    0x1b01a57b, 0x8208f4c1, 0xf50fc457, 0x65b0d9c6, 0x12b7e950,
    0x8bbeb8ea, 0xfcb9887c, 0x62dd1ddf, 0x15da2d49, 0x8cd37cf3,
    0xfbd44c65, 0x4db26158, 0x3ab551ce, 0xa3bc0074, 0xd4bb30e2,
    0x4adfa541, 0x3dd895d7, 0xa4d1c46d, 0xd3d6f4fb, 0x4369e96a,
    0x346ed9fc, 0xad678846, 0xda60b8d0, 0x44042d73, 0x33031de5,
    0xaa0a4c5f, 0xdd0d7cc9, 0x5005713c, 0x270241aa, 0xbe0b1010,
    0xc90c2086, 0x5768b525, 0x206f85b3, 0xb966d409, 0xce61e49f,
    0x5edef90e, 0x29d9c998, 0xb0d09822, 0xc7d7a8b4, 0x59b33d17,
    0x2eb40d81, 0xb7bd5c3b, 0xc0ba6cad, 0xedb88320, 0x9abfb3b6,
    0x03b6e20c, 0x74b1d29a, 0xead54739, 0x9dd277af, 0x04db2615,
    0x73dc1683, 0xe3630b12, 0x94643b84, 0x0d6d6a3e, 0x7a6a5aa8,
    0xe40ecf0b, 0x9309ff9d, 0x0a00ae27, 0x7d079eb1, 0xf00f9344,
    0x8708a3d2, 0x1e01f268, 0x6906c2fe, 0xf762575d, 0x806567cb,
    0x196c3671, 0x6e6b06e7, 0xfed41b76, 0x89d32be0, 0x10da7a5a,
    0x67dd4acc, 0xf9b9df6f, 0x8ebeeff9, 0x17b7be43, 0x60b08ed5,
    0xd6d6a3e8, 0xa1d1937e, 0x38d8c2c4, 0x4fdff252, 0xd1bb67f1,
    0xa6bc5767, 0x3fb506dd, 0x48b2364b, 0xd80d2bda, 0xaf0a1b4c,
    0x36034af6, 0x41047a60, 0xdf60efc3, 0xa867df55, 0x316e8eef,
    0x4669be79, 0xcb61b38c, 0xbc66831a, 0x256fd2a0, 0x5268e236,
    0xcc0c7795, 0xbb0b4703, 0x220216b9, 0x5505262f, 0xc5ba3bbe,
    0xb2bd0b28, 0x2bb45a92, 0x5cb36a04, 0xc2d7ffa7, 0xb5d0cf31,
    0x2cd99e8b, 0x5bdeae1d, 0x9b64c2b0, 0xec63f226, 0x756aa39c,
    0x026d930a, 0x9c0906a9, 0xeb0e363f, 0x72076785, 0x05005713,
    0x95bf4a82, 0xe2b87a14, 0x7bb12bae, 0x0cb61b38, 0x92d28e9b,
    0xe5d5be0d, 0x7cdcefb7, 0x0bdbdf21, 0x86d3d2d4, 0xf1d4e242,
    0x68ddb3f8, 0x1fda836e, 0x81be16cd, 0xf6b9265b, 0x6fb077e1,
    0x18b74777, 0x88085ae6, 0xff0f6a70, 0x66063bca, 0x11010b5c,
    0x8f659eff, 0xf862ae69, 0x616bffd3, 0x166ccf45, 0xa00ae278,
    0xd70dd2ee, 0x4e048354, 0x3903b3c2, 0xa7672661, 0xd06016f7,
    0x4969474d, 0x3e6e77db, 0xaed16a4a, 0xd9d65adc, 0x40df0b66,
    0x37d83bf0, 0xa9bcae53, 0xdebb9ec5, 0x47b2cf7f, 0x30b5ffe9,
    0xbdbdf21c, 0xcabac28a, 0x53b39330, 0x24b4a3a6, 0xbad03605,
    0xcdd70693, 0x54de5729, 0x23d967bf, 0xb3667a2e, 0xc4614ab8,
    0x5d681b02, 0x2a6f2b94, 0xb40bbe37, 0xc30c8ea1, 0x5a05df1b,
    0x2d02ef8d
};

uint32_t Ql_Check_CRC32(uint32_t Val, const unsigned char *Data, int32_t Offset, int32_t Length)
{
    uint32_t result = Val ^ 0xFFFFFFFF;
    uint32_t i = 0;
```

```
    for (i = Offset; i < (Offset + Length); i++)
    {
        result = crc32_tab[(result ^ Data[i]) & 0xFF] ^ (result >> 8);
    }

    return (result ^ 0xFFFFFFFF);
}
```

2.2.7. Firmware Upgrading

After the CRC is passed, reset the module. During power-on cycle, the module will output the information of the firmware version. You can also use the command **$PSTMGETPAR,1500<CR><LF>** to get the firmware version information. This information is used to verify whether the module has been successfully upgraded.

3 Appendix A References

Table 2: Terms and Abbreviations

Abbreviation	Description
CRC	Cyclic Redundancy Check
FW	Firmware
GNSS	Global Navigation Satellite System
NVM	Non-volatile memory
PC	Personal Computer
UART	Universal Asynchronous Receiver/Transmitter