Updating Firmware for TXx2-W and EXSx2-W Modules

Application Note 16

80783NT12058A - Rev. 0
2023-8-10
Released
Confidential

Technical Documentation

Contents

Tables

Figures

1 Applicability Table

Table 1: **Applicability table**

Products
Cinterion® TX62-W Release
Cinterion® TX62-W-B Release
Cinterion® TX82-W Release
Cinterion® EXS62-W as of Release 01.100
Cinterion® EXS82-W as of Release 01.100

2 Introduction

2.1 Scope

This Application Notes describes how to update the firmware of products listed in Section 1.

The firmware of a module is stored in its Flash System. Firmware updates (including incremental firmware updates/patches for EXSx2-W and TXx2-W products) are released as binary and executable files. They have the following syntax:

File name	Example
Full update (standard):	
<product>_<revision>_<arn>.usf	exs82-w_rev01.001_arn01.000.00.usf
<product>_<revision>_<arn>_gwinswup.exe	exs82-w_rev01.001_arn01.000.00_gwinswup.exe
Incremental update (delta patches):	
<product>_<revision-old>_<arn-old>_to_<revision-new>_<arn-new>.usf	exs82-w_rev01.001_arn01.000.00_to_rev01.001_arn01.000.06.usf
<product>_<revision-old>_<arn-old>_to_<revision-new>_<arn-new>_gwinswup.exe	exs82-w_rev01.001_arn01.000.00_to_rev01.001_arn01.000.06_gwinswup.exe

Legend:

<product>	Product variant or family name, e.g., EXSx2-W or TXx2-W.
<revision> | <revision-old> | <revision-new>	Firmware's revision number. Example: rev**.*** --> rev01.001
<arn> | <arn-old> | <arn-new>	Firmware's application revision number. Example: arn**.***.** --> arn01.000.00
*.usf	The binary file is intended for download through the host application or a dedicated program developed by the customer for any operating system.
*_gwinswup.exe	Executable program named gWinSwup integrating the *.USF binary file and handling all procedures needed for updating the firmware. Program is ready for use on systems running Microsoft® Windows.

Apart from the executable program gWinSwup allowing firmware updates from systems running Windows, there exists also the possibility for firmware updates from systems running Linux (see Chapter 3). Chapter 4 describes incremental firmware updates for EXSx2-W and TXx2-W products, and Chapter 5 gives advice on how to develop a customized update program.

2.2 Audience

This document is intended for system integrators that are using the Telit EXSx2-W and TXx2-W modules in their products.

2.3 Contact Information, Support

For technical support and general questions, e-mail:

- TS-EMEA@telit.com
- TS-AMERICAS@telit.com
- TS-APAC@telit.com
- TS-SRD@telit.com
- TS-ONEEDGE@telit.com

Alternatively, use: https://www.telit.com/contact-us/

Product information and technical documents are accessible 24/7 on our website: https://www.telit.com

2.4 Conventions

Note: Provide advice and suggestions that may be useful when integrating the module.

Danger: This information MUST be followed, or catastrophic equipment failure or personal injury may occur.

ESD Risk: Notifies the user to take proper grounding precautions before handling the product.

Warning: Alerts the user on important steps about the module integration.

All dates are in ISO 8601 format, that is YYYY-MM-DD.

2.5 Related Documents

[1] Hardware Interface Description related to your Telit Cinterion product
[2] AT Command Set Specification related to your Telit Cinterion product
[3] Release Notes related to your Telit Cinterion product
[4] Application Note 17: Over-The-Air Firmware Update (FOTA)

Documents related to evaluation kits supplied by Telit Cinterion:

[5] DSB75 Support Box Hardware Interface Description
[6] Cinterion® LGA DevKit User Guide

For testing and evaluating software update procedures you can take advantage of evaluation modules mounted on an evaluation kit such as:

- DSB75 Support Board + DSB75 Adapter
- Cinterion® LGA DevKit SM + Cinterion® LGA DevKit socket SML

2.6 General Requirements and Precautions

- The firmware download is intended to install either the same or a new firmware version. Installation of an older firmware version may lead to the module no longer being operational.
- Take every precaution to avoid disruption of the firmware download. Do not bend, stress or remove any cable. In the event of failure, there would be no valid software installed - although the update process can be started again any time using "no firmware" mode.

2.7 User Defined Settings

The firmware download procedure has no impact on user defined parameters. Non-volatile user settings will be preserved.

2.8 Firmware Download Interfaces

Updates can be downloaded to the module via the ASC0 interface and USB-ACM port.

The module's interface used for the download shall connect to an appropriate port of the computer where the software update files are located, e.g.,

ASC0:

- on Windows systems COMx for the ASC0 interface (see Figure 1; sample port COM102),
- on Linux systems, e.g. Linux Ubuntu, the port named /dev/ttyS1 (or /dev/ttyUSB0, when using a USB to Serial Adapter).

USB-ACM:

- on Windows systems COMx for the ASC0 interface (see Figure 2; sample port COM27),
- on Linux systems, e.g. Linux Ubuntu, the port named /dev/ttyACM0 (or /dev/ttyACM*, if has more than one ACM USB devices connected).

Depending on the selected interface, the following lines are required:

Table 2: Required lines

Used module interface	Lines used on module side	Lines used on application side (e.g. PC)
ASC0	RXD, TXD, ON, GND	RXD, TXD, DTR connected to module's ON line, GND
USB-ACM		

ASC0:

- When downloading software over ASC0 it is recommended to have a connection between the module's ON line and the computer's DTR line. This enables the download utility to start the module in the following cases:
  - **a** at the beginning of the download if the module is still in Power Down mode,
  - **b** after successful software transfer, to restore the original baudrate of the module

if the baudrate used by download utility was not identical with the module's original baudrate (AT+IPR).

If a) and b) are not necessary, because the module is switched on before the download utility starts and because the baudrates selected for the download utility and for AT+IPR are the same, there is no need to toggle the ON line during the download process, and thus no need to connect both lines.

- On the DSB75 Support Board, the connection between ON and DTR is available for ASC0,

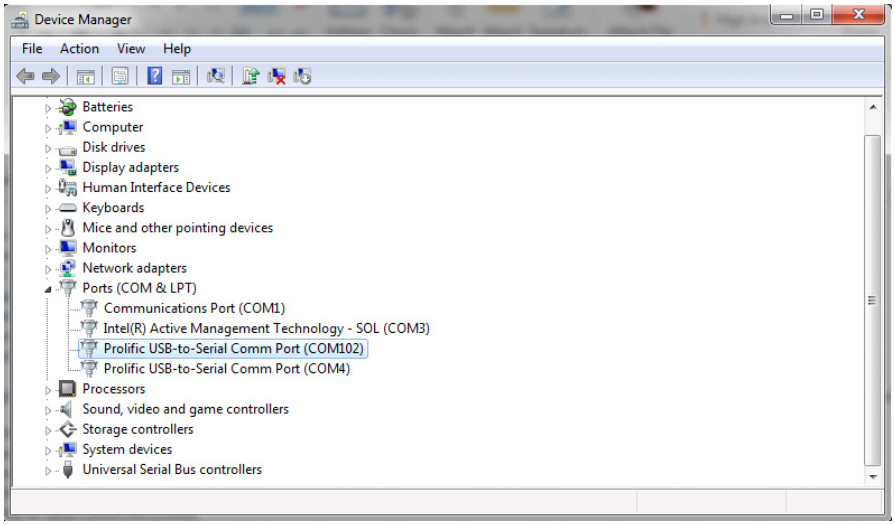


Figure 1: **Example: Device Manager with sample port COM102 assigned to ASC0**

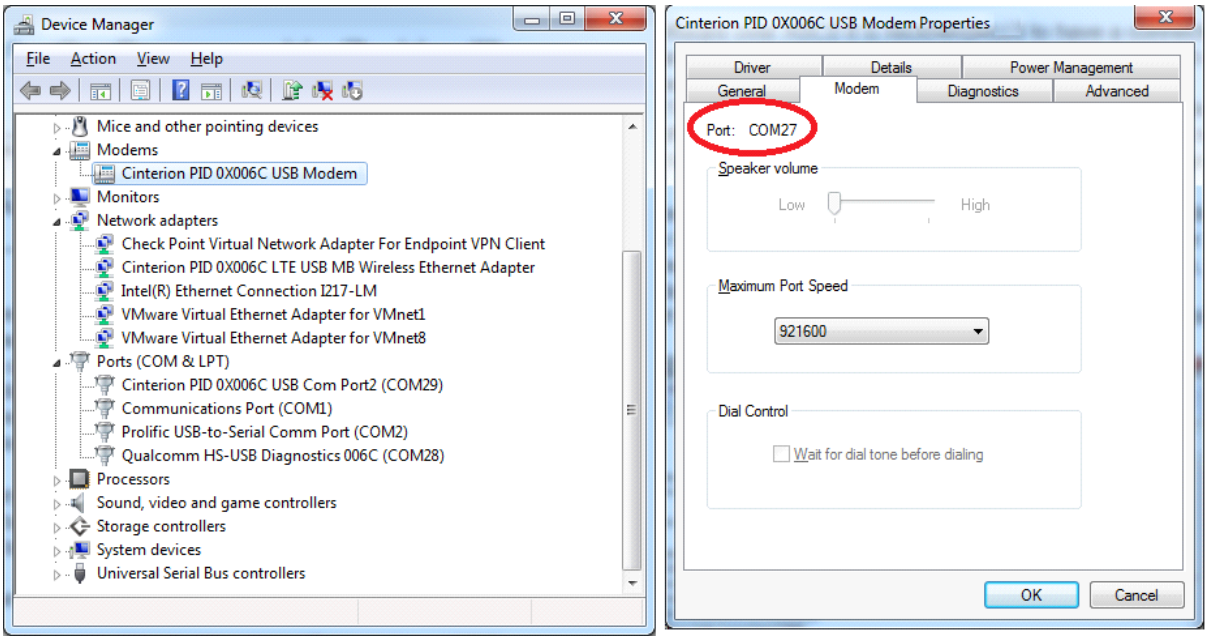


Figure 2: **Example: Device Manager with sample port COM27 assigned to USB-ACM port**

2.9 Terms and Conditions

Refer to https://www.telit.com/hardware-terms-conditions/.

2.10 Disclaimer

THE MATERIAL IN THIS DOCUMENT IS FOR INFORMATIONAL PURPOSES ONLY. TELIT CINTERION RESERVES THE RIGHT TO MAKE CHANGES TO THE PRODUCTS DESCRIBED HEREIN. THE SPECIFICATIONS IN THIS DOCUMENT ARE SUBJECT TO CHANGE AT THE DISCRETION OF TELIT CINTERION WITHOUT PRIOR NOTICE. THIS DOCUMENT IS PROVIDED ON "AS IS" BASIS ONLY AND MAY CONTAIN DEFICIENCIES OR INADEQUACIES. TELIT CINTERION DOES NOT ASSUME ANY LIABILITY FOR INFORMATION PROVIDED IN THE DOCUMENT OR ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT DESCRIBED HEREIN.

TELIT CINTERION GRANTS A NON-EXCLUSIVE RIGHT TO USE THE DOCUMENT. THE RECIPIENT SHALL NOT COPY, MODIFY, DISCLOSE, OR REPRODUCE THE DOCUMENT EXCEPT AS SPECIFICALLY AUTHORIZED BY TELIT CINTERION.

TELIT CINTERION AND THE TELIT CINTERION LOGO, ARE TRADEMARKS OF TELIT CINTERION AND ARE REGISTERED IN CERTAIN COUNTRIES. ALL OTHER REGISTERED TRADEMARKS OR TRADEMARKS MENTIONED IN THIS DOCUMENT ARE THE PROPERTY OF THEIR RESPECTIVE OWNERS AND ARE EXPRESSLY RESERVED BY TELIT CINTERION (AND ITS LICENSORS).

3 Firmware Update Tools

This chapter introduces the two update tools gWinSwup (see Section 3.1) and gLinSwup (see Section 3.2) available for firmware updates (or incremental firmware updates). These tools basically automate the following steps required to update the module's firmware:

1. Switch on module
2. Host uses AT^SFDL command to trigger the firmware update
3. Module responds
4. Host begins to send *.usf file data
5. Module updates firmware and gives host response
6. Repeat step 3 and step 4 (if necessary).
7. Done

After the update procedure is complete, verify that the software update was successful. The current firmware version can for instance be queried with ATI1.

3.1 gWinSwup

gWinSwup comes as a ready-to-use program for Windows 7, Windows 8, and Windows 10 designed to transfer the firmware from a Windows computer to the module's FFS. The gWinSwup executable includes all procedural steps listed above in Chapter 3.

The source code for gWinSwup can be supplied on request.

The following steps are required to complete the firmware download:

- If the module's ON line and the computer's DTR line are connected the download can be started no matter whether the module is switched on or off.
- Close any application connected to the module.
- To start the program run the **_gwinswup.exe* file and select the COM port the module is connected to.
- Select the baud rate. Higher baud rates should be set only if supported by the computer's COM port. gWinSwup supports the following baud rates: 115200, 230400, 460800, 921600 Baud.

  The baud rate for the firmware download can be set regardless of the module's baud rate configured with AT+IPR. Thus, you can take advantage of a higher baud rate (if supported by the computer's COM port). If the baud rate selected for the download differs from the AT+IPR setting, gWinSwup briefly shows the module's original setting before starting the download at the selected baud rate. After successful transfer the original AT+IPR setting of the module is valid again.

  If the download aborts prematurely and, after the next attempt, completes successfully, the original baud rate cannot be restored. To operate the module from the ASC0 interface you may need to set AT+IPR again. See download abort on ASC0 in Section 3.1.1.

- Click the *START* button. The installation will take a couple of minutes.
- After the download has completed the module remains switched on and is ready to operate.
- Close gWinSwup.

NOTE:

1. If USB COM port is connected during module reset, WINDOWS will refuse to enumerate the USB port after a reset. Close the SWUP app or disconnect any Terminal program before manually resetting the module.
2. No need to setup baudrate for USB-ACM COM-port. If USB-ACM port is selected from gWinSwup app, the baudrate will be marked in grey and not selectable.
3. Module's USB driver has to be correctly installed on windows before using USB interface for SWUP. i.e. can input AT command from the USB-ACM COM-port.

The gWinSwup download procedure is shown below in Figure 3.

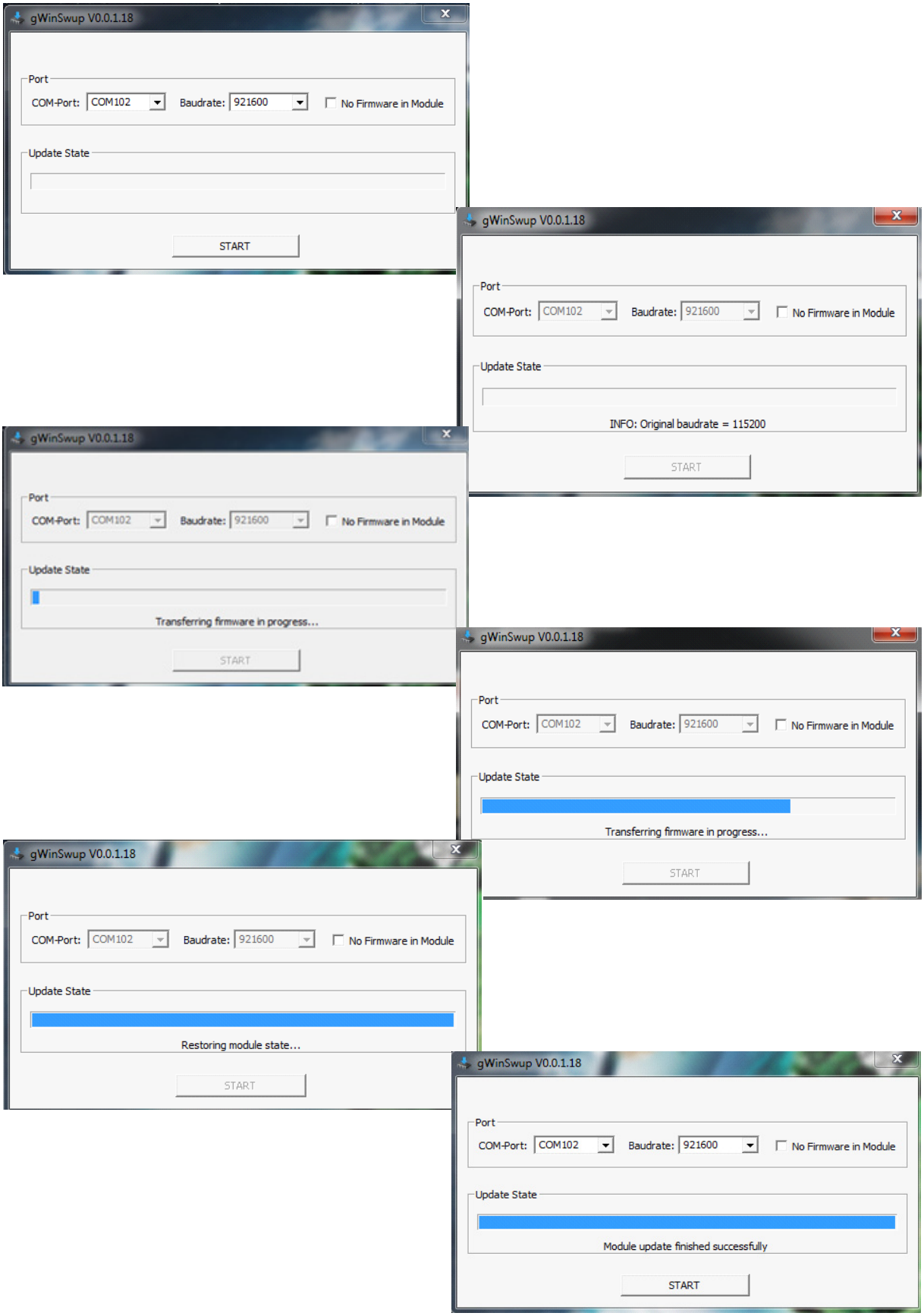
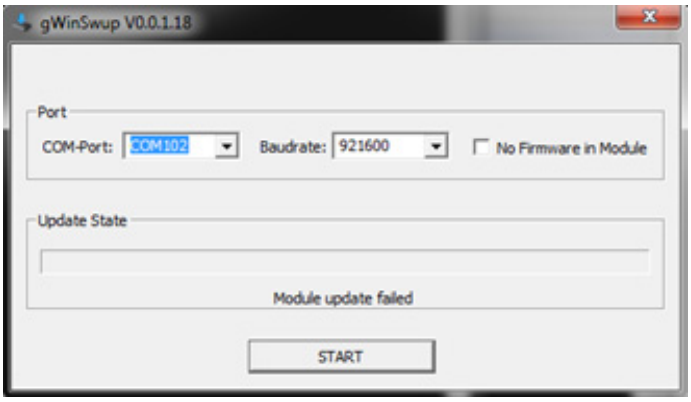
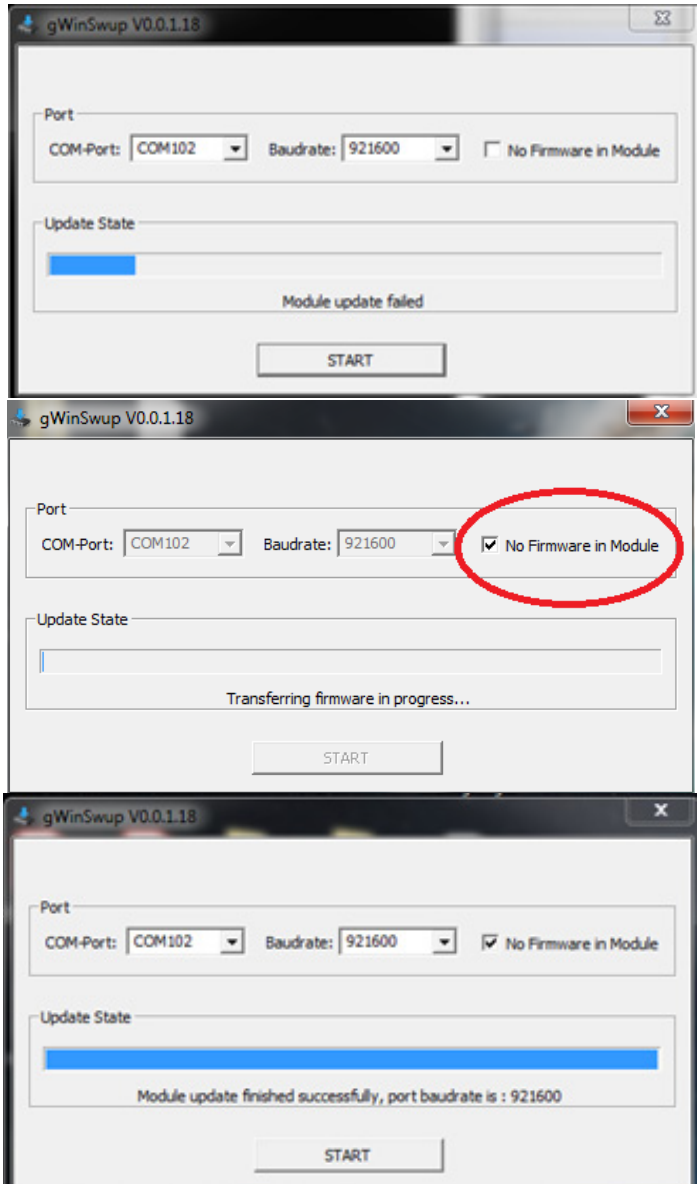


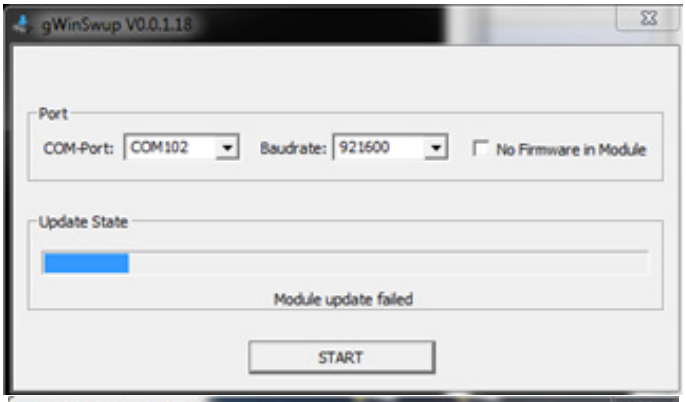
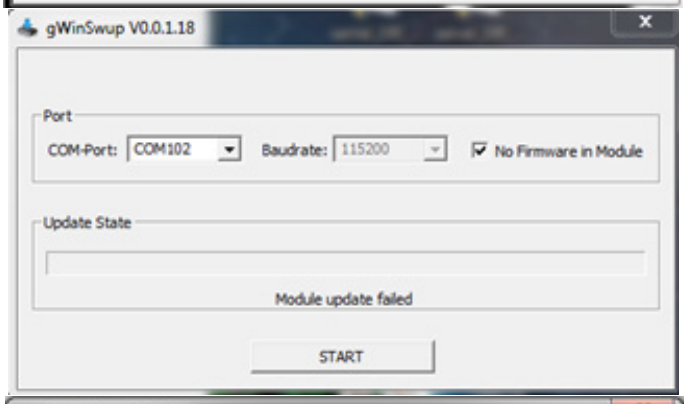
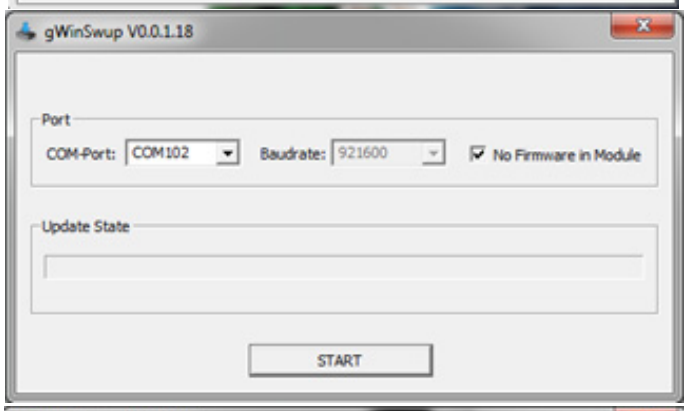
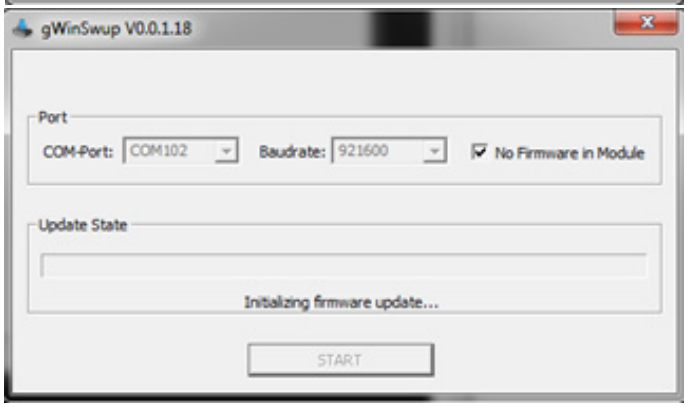
Figure 3: gWinSwup download via ASC0

3.1.1 Troubleshooting gWinSwup

Table 3: **Troubleshooting gWinSwup problems**

gWinSwup dialog box	Meaning
	*Download failure on ASC0:* a) The COM port is not free for download. Probably, the host application or a Terminal program is still open. b) The baudrate selected with gWinSwup is higher than the maximum rate supported by the computer's COM port. *Download failure on USB-ACM:* a) The COM port is not free for download. Probably, the host application or a Terminal program is still open. b) The USB driver is not OK and/or USB port not correctly enumerated.

gWinSwup dialog box	Meaning
	*Download abort:* This error message occurs if the download was aborted during firmware update, e.g. due to switching off the module or disconnecting the power supply. The module is no longer responding to for example AT commands. *Solution:* • Restart the module • Open gWinSwup and • Check the option *"No Firmware in Module"* • Click the *START* button. • The download process will be performed as described in Chapter 3. • In this case, after the software update, the original baudrate can not be restored, which will be prompted at the gWinSwup window. The original baudrate can be restored by sending "AT +IPR".

gWinSwup dialog box	Meaning
   	*Failed in No Firmware Mode:* The reason is that you have changed the baudrate in No Firmware Mode compared to previous Normal Firmware update. Normally if the firmware update is interrupted, the same baudrate must be kept to do firmware update in No Firmware mode. *Solution:* • Restart the module. • Open gWinSwup. • Change Baudrate to before-interrupt value • Check "No Firmware in Module" • Click the START button.

3.2 gLinSwup

gLinSwup is a program intended to be used as a tool for downloading firmware from host systems running Linux. The gLinSwup source code is provided at the discretion of Telit Cinterion DIS AIS under a proprietary "SOURCE CODE LICENSE AGREEMENT".

The gLinSwup source code can be compiled or used as example for developing a customized download utility under Linux. The source code handles all steps and AT commands required to update the module‘s firmware as listed above in Chapter 3. Appropriate time periods and delays are set to efficiently manage all tasks. The source code also includes detailed comments on how to handle all necessary procedures.

4 Incremental Firmware Update

In order to perform a firmware upgrade, it is not invariably necessary to update a module's complete firmware. Instead, a firmware upgrade may sometimes only comprise certain parts of a module's firmware - to for instance implement software patches.

Please note that incremental firmware patches are provided at the discretion of Telit Cinterion DIS AIS as part of Telit Cinterion's maintenance strategy. Original base firmware and suitable incremental target firmware patches are customer-specifically aligned, i.e., incremental firmware patches can be specifically prepared between two appropriate firmware versions.

The basic concept, prerequisites, and steps required for such an incremental firmware update however, are exactly the same as for a complete firmware update, except that the *.usf binary files made available for an incremental software update are as a rule smaller, and that only parts of the software are actually updated/patched in the end.

Therefore, for an incremental firmware update, please follow the explanations and steps as given for a full firmware update in Chapter 3.

Also, please ensure that the firmware update/patch file is actually intended for the module's current base firmware version.

If gWinSwup returns "ERROR: Download failed" or "ERROR: Upload failed" before the actual incremental firmware update has started, i.e., before the status bar starts to show the download progress, the program most probably tries to upload a wrong patch to the module. In such a case the module will still function with its existing firmware version.

IMPORTANT: Once the incremental patch procedure is started there is no possibility to cancel and automatically roll-back to the original firmware version! This means that an interrupted incremental firmware download must be repeated with exactly the same patch in order to successfully complete the firmware update, or a standard firmware update needs to be performed to roll-back to the original firmware version. For both cases it is required to follow the troubleshooting solution described for a download abort in Section 3.1.1.

5 Developing Firmware Download Utility

This chapter gives advice on how to develop a utility dedicated for downloading the firmware *.USF file into the module's Flash System. The download interface is the AT command instance at ASC0.

Designed according to these guidelines, the gWinSwup or gLinSwup programs supplied by Telit Cinterion DIS AIS are examples of a ready-to-use download program which integrates the firmware *.USF file and meets the requirements specified below.

5.1 Format of the *. USF Firmware File

The firmware *.USF file for your product is supplied by Telit Cinterion. The file consists of records to be transferred one by one from the download utility to the module.

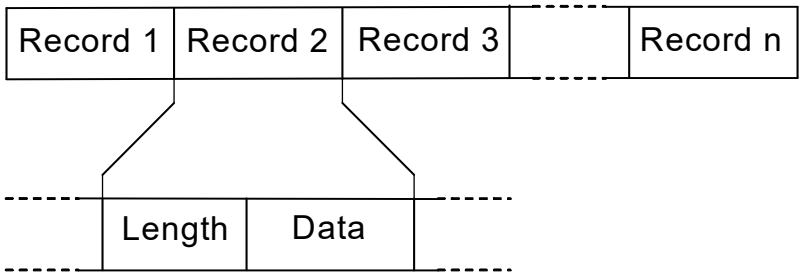


Figure 4: Format of *.USF firmware file

Length: Byte 1 and 2 of each record indicate the number of the following data bytes. LSB (least significant byte) first.

Data: 3rd to nth byte

5.2 Requirements for the Firmware Download Utility

The download utility must be designed to meet the requirements of the download process specified in the download protocol. Figure 5 shows a flow chart representing the required steps. A programming example is given in Section 5.3. For further detail on the AT^SFDL command see also [2].

First of all, to start the download process, the download utility has to send the "AT^SFDL" command to the module. Then the module enters the firmware download mode and sends the start pattern to the download utility. Upon receipt of the start pattern ("ANSWER_OK" in the example below) the download utility must start to send the first record.

Optionally, we recommend that the buffer of the serial interface be purged before the download utility starts to send the first record.

After each record the download utility has to wait for an acknowledgment delivered by the module. If this is "OK", the next record can be transferred. "BUSY" means that the module is still processing the record last received - in this case, the transfer must be halted until the next "OK" acknowledgment is received. A checksum verifies the correct transmission of each record. If a download error is detected, the module will send a "RETRY" message.

The download utility must be capable to evaluate the answer patterns received from the module and to correctly transmit the data.

Once all records are successfully received, the AT^SFDL command returns an "OK" message, and the update process is finished.

Table 4: Answer patterns required for generic download utility

Answer pattern	Hex value	Function
`#define ANSWER_OK`	`0x01`	After calling AT^SFDL, "`ANSWER_OK`" notifies the download utility that the first record can be sent. After each successfully received record, "`ANSWER_OK`" notifies the download utility that the next data record can be transferred.
`#define ANSWER_RETRY`	`0x02`	Notifies that a failure was detected after verifying the checksum. The download utility is requested to send the last record again and wait until this record is successfully received and acknowledged by "`ANSWER_OK`".
`#define ANSWER_FATAL`	`0x03`	Download failed at all. Possible causes: The firmware you are trying to download is not intended for this module type, or you are trying to install an older version. The currently installed software is not affected.
`#define ANSWER_BUSY`	`0x04`	The module is still processing the record received. The transfer from the download utility must be halted until the module sends the next "`ANSWER_OK`". For example, "`ANSWER_BUSY`" may appear when the flash is erased on the module. This procedure usually takes a couple of seconds. In this case, "`ANSWER_BUSY`" is no error.

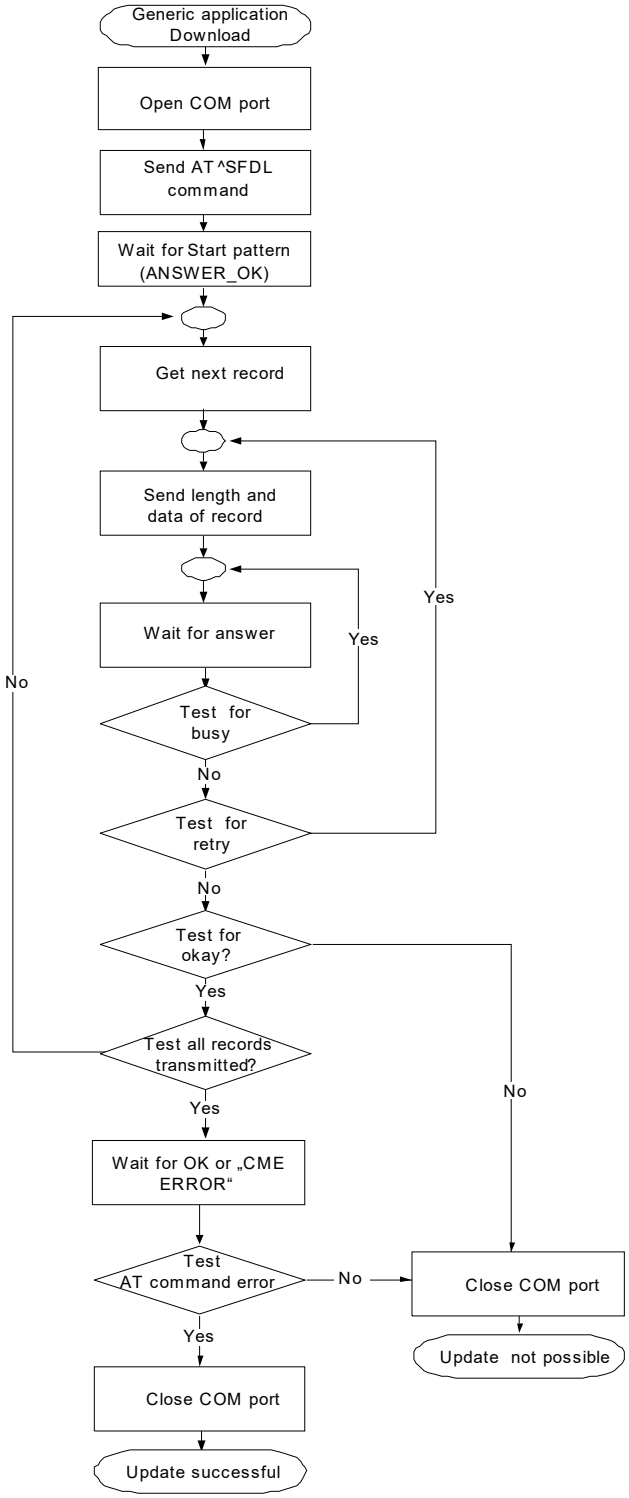


Figure 5: Download protocol

5.3 Programming Example

```
void DoUpdate (void)
{
   BYTE bAnswer ;
   BYTE bRecordData[MAX_RECORD_SIZE] ;
   WORD wRecordLen ;

   /* Send AT command */
   WriteToModule ((BYTE *) "AT^SFDL\r", 8) ;

   /* Wait for start pattern */
   do
   {
   ReadFromModule (&bAnswer, 1) ;
   }
   while (ANSWER_OK != bAnswer) ;

   do
   { /* Read record from file */
     wRecordLen = MAX_RECORD_SIZE ;
     FileReadRecord (&wRecordLen, bRecordData) ;

     do
     {  /* Send length of record */
        WriteToModule ((BYTE *) &wRecordLen, sizeof (WORD)) ;

        /* Send data of record */
        WriteToModule (bRecordData, wRecordLen) ;

        do
        {/* Wait for answer */
        ReadFromModule (&bAnswer, 1) ;
        }
        while (ANSWER_BUSY == bAnswer) ;
     }
     while (ANSWER_RETRY == bAnswer) ;
   }
   while (!FileEnd () && ANSWER_OK == bAnswer) ;

   /* Wait for "OK" result code*/
   if (!ReadResultLineFromModule ())
     /* Error */
     return ;
}
```

6 Document History

New document: "Updating Firmware for TXx2-W and EXSx2-W Modules", Version 02

Preceding document: "Updating Firmware for TXx2-W and EXSx2-W Modules", Version 01

Chapter	What is new
---	Supported products updated.

New document: "Updating Firmware for TXx2-W and EXSx2-W Modules", Version 01

Chapter	What is new
---	Initial document setup.

© Telit 2023 All rights reserved.