NEC

User's Manual

Data FLASH Converter

Conversion tool for data flash memory image generation

Document No. U19003EE2V0UM00
Date Published March 2010

© NEC Electronics (Europe) GmbH

- **The information in this document is current as of March, 2010. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".
  The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.
  "Standard": Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots.
  "Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).
  "Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)
(1) "NEC Electronics" as used in this statement means NEC Electronics Corporation and also includes its majority-owned subsidiaries.
(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

M8E 02. 11-1

Revision History

Date	Revision	Chapter	Description
24-10-2007	V1.00	---	First release
01-03-2010	V2.00	---	Second release, updated entire manual to describe UX6 FLASH Technology support in combination with the V850 EEPROM emulation library EEL (Type T05).

Table of Contents

1. Introduction 7

1.1 Abbreviations 7
1.2 Main features of Data FLASH Converter 8
1.3 Trademarks 8

2. Input files / formats 9

2.1 Input file combinations 9
2.2 Program Code for Data FLASH or Code FLASH 10
2.3 EEPROM emulation data 10
2.3.1 EEPROM XML file format for V850E1/ES 11
2.3.2 EEPROM XML file format for V850E2 13

3. Output file / formats 15

4. Software Installation 16

5. Data FLASH converter, GUI 20

5.1 Toolbar 22
5.2 The Menu 22
5.2.1 File menu 23
(1) Hex editor / viewer 23
(2) Open Device Parameter File 28
(3) Set Data FLASH area 29
(4) Open Data FLASH EEPROM file 31
(5) Open Data FLASH program file 33
(6) Open Code FLASH program file 35
(7) Generate merged Data FLASH / Code FLASH file 37
(8) Generate Data FLASH file 38
5.3 Process window 40

6. Data FLASH converter, command line interface 41

6.1 Command line options 42
6.2 Combination of command line options 43
6.2.1 Generation of merged Data FLASH / Code FLASH file 44
(1) Generation of output file including Code FLASH, Data FLASH program and EEPROM emulation data . 44
(2) Generation of output file including Code FLASH program and EEPROM emulation data 44
(3) Generation of output file including Code FLASH program and Data FLASH program data 44
6.2.2 Generation of Data FLASH file 46
(1) Generation of output file including Data FLASH program and EEPROM emulation data 46
(2) Generation of output file including Data FLASH program data 46
(3) Generation of output file including EEPROM emulation data 46
6.3 Return Codes 49

7. Error Messages 50

List of Figures

Figure 1: EEPROM emulation data, XML file format for V850E1/ES....11
Figure 2: EEPROM emulation data, XML file format for V850E2 13
Figure 3: Install preparation....16
Figure 4: Welcome screen 16
Figure 5: License Agreement window 17
Figure 6: Choose Destination Folder window 17
Figure 7: Ready to Install window 18
Figure 8: Installation progress window....18
Figure 9: Installation completion window 19
Figure 10: Program folder after installation 19
Figure 11: Data FLASH Converter main window 20
Figure 12: Toolbar Buttons....22
Figure 13: File menu....23
Figure 14: Hex Editor, open file dialog 24
Figure 15: Hex Editor, main window....25
Figure 16: Hex Editor, Save as dialog....26
Figure 17: Hex Editor, format of saved Data FLASH file....27
Figure 18: Open Device Parameter File....28
Figure 19: Set Data FLASH, start address....29
Figure 20: Set Data FLASH, size 29
Figure 21: Config Window, Data FLASH area box....30
Figure 22: Open Data FLASH EEPROM file dialog 31
Figure 23: Config Window, Data FLASH EEPROM file box 32
Figure 24: Address map window, Data FLASH memory box (1) 32
Figure 25: Open Data FLASH program file dialog 33
Figure 26: Config Window, Data FLASH Program file box 33
Figure 27: Address map window, Data FLASH memory box (2) 34
Figure 28: Open Code FLASH program file dialog 35
Figure 29: Config Window, Code FLASH Program file box 35
Figure 30: Address map window, Code FLASH memory box....36
Figure 31: Save merged Data FLASH / Code FLASH file dialog....37
Figure 32: Save Data FLASH file dialog....38
Figure 33: Process window 40
Figure 34: Data FLASH Converter, command line shell 41
Figure 35: Dual output generation for Data FLASH memory image (V850E1/ES) 47
Figure 36: Output generation for Data FLASH memory image (V850E2) 48
Figure 37: Error message, dialog box 50
Figure 38: Error message, command shell window 50

List of Tables

Table 1: Supported combinations of input files 9
Table 2: Main window composition 21
Table 3: Hex Editor, navigation and input keys 26
Table 4: Command line options 42
Table 5: Error Messages 1/2 51
Table 6: Error Messages 2/2 52

1. Introduction

The Data FLASH Converter is a windows based tool that generates a Data FLASH memory image from EEPROM emulation data and/or from a program code file that is mapped to the Data FLASH area of a V850 microcontroller.

Additionally the Data FLASH Converter is able to generate a merged output file containing EEPROM emulation data, program code for Data FLASH and program code for Code FLASH. This output file can be read by the standard NEC FLASH programmer PG-FP5 and can be used for FLASH programming of a V850 based device containing Code FLASH and Data FLASH.

Apart from the FLASH programmer support the Data FLASH Converter can generate a data flash memory image that can be read by the standard debugger tool chain for V850. This makes it possible to pre-initialize the Data FLASH memory also during the debugging phase.

1.1 Abbreviations

Abbreviations	***Description***
EEL	EEPROM Emulation Library
FAL	FLASH Access Library (FLASH access layer)
FDL	FLASH Data Library (Data FLASH access layer)
FSL	FLASH Self-programming Library (accessing code FLASH)
FW	Firmware
FCB	FLASH Control Block
XML	e***X***tensible ***M***arkup ***L***anguage

1.2 Main features of Data FLASH Converter

- Generates a Data FLASH memory image based on EEPROM emulation data represented by an XML file.
- Generates a Data FLASH memory image for the following V850 based devices:

  V850E1/ES: UX4 and MF2 FLASH Technology in combination with the V850 EEPROM emulation library EEElib (previous Type)

  V850E2: UX6 FLASH Technology in combination with the V850 EEPROM emulation library EEL (Type T05)

- Merges EEPROM emulation data and program data into one Data FLASH memory image.
- Generates a Data FLASH memory image including ID-tag information from a program file mapped to the Data FLASH area. Supported input file formats are Intel-hex and Motorola S-record. For Intel-hex the extended linear address record and extended segment address record formats are supported. With regard to the Motorola-S record type, the S2 and S3 record formats are supported for program code mapped to the Data FLASH memory.
- Merges EEPROM emulation data and program data into one Data FLASH memory image.
- Merges Code FLASH programs and Data FLASH memory images into one output file.
- Generates two output files for FLASH programming and debugging purpose.
- Supports two output file formats: Intel-hex and Motorola-S record.
- A graphical user interface is provided as well as a command line interface for batch mode support or automatic output file generation.

1.3 Trademarks

Microsoft and Windows are registered trademarks of Microsoft Corporation. All other product names are trademarks or registered trademarks of their respective owners.

2. Input files / formats

2.1 Input file combinations

The Data FLASH Converter is able to handle the following combination of input files, for the generation of a single Data FLASH memory image or the generation of a merged Code FLASH / Data FLASH file.

	Supported combinations of input files	***Supported file formats***
1	CodeFLASH program file + DataFLASH program file + EEPROM file	HEX, SREC, XML
2	CodeFLASH program file + DataFLASH program file	HEX, SREC
3	CodeFLASH program file + EEPROM file	HEX, SREC, XML
4	DataFLASH program file + EEPROM file	HEX, SREC, XML
5	DataFLASH program file	HEX, SREC
6	EEPROM file	XML

Table 1: Supported combinations of input files

2.2 Program Code for Data FLASH or Code FLASH

The program code has to be available in an Intel-hex or Motorola-S record format. For Intel-hex the extended linear address record and extended segment address record formats are supported. With regard to the Motorola-S record type, the S2 and S3 record formats are supported for program code mapped to the Data FLASH memory. For program files mapped to the Code FLASH the S1, S2 and S3 record formats are supported.

2.3 EEPROM emulation data

To specify the EEPROM emulation data the Data FLASH converter supports the standard XML file format (e***X***tensible ***M***arkup ***L***anguage) as input format.

The EEPROM emulation data can be specified for V850E1/ES as well for the V850E2 based microcontrollers:

V850E1/ES: UX4 and MF2 FLASH Technology in combination with the V850 EEPROM emulation library EEElib (previous Type)

V850E2: UX6 FLASH Technology in combination with the V850 EEPROM emulation library EEL (Type T05)

2.3.1 EEPROM XML file format for V850E1/ES

The XML input format and the corresponding tags for V850E1/ES based microcontrollers is shown below.

```
<?xml version="1.0"?>
<DataFlash>
        <AdministrativeSection>
                <SectionSize>0x0800</SectionSize>
                <Offset>0x0000</Offset>
        </ AdministrativeSection >
        <Record>
                <ID>0x80</ID>
                <Length>4</Length>
                <Data>0x20, 0x30, 0x40, 0x50</Data>
        </Record>
        <Record>
                <ID>0x81</ID>
                <Length>8</Length>
                <Data>
                        0x20, 0x30, 0x31, 0x32,
                        0x40, 0x40, 0x41, 0x42
                </Data>
        </Record>
</ DataFlash >
```

Figure 1: EEPROM emulation data, XML file format for V850E1/ES

The predefined tags are:

DataFlash	Defining the start of the definition file.
AdminstrativeSection	This section is describing the physical parameters that are needed for the file generation.
SectionSize	Size of one section that is used for the EEPROM emulation, e.g. 0x800 (2 kByte), 0x1000 (4 kByte), 0x2000 (8 kByte), 0x4000 (16 kByte)…
Offset	The offset where the section starts in the Data FLASH. It is always related to the start address of the Data FLASH. The offset can have a size of 2 kByte * n.
Record	Defining the start of a ID-L record.
ID	Identifier for the data record (16bit). Format is fixed to four digits in hex.
Length	Length definition of the data record (16bit). Format is fixed to four digits in hex. The minimum data length is equal to four bytes. The length can be specified in 4 Byte *n steps.
Data	Any hexadecimal or decimal value. The line length is not fixed.

Comments have to be added as specified in XML definition starting with ‘<!-- ’ and ending with ‘ -->’

The values defining the ID, length or data information must have the following ‘C’ like notation:

- Leading ‘0x’ denotes a hex value.
- Leading zeros within a hex value are interpreted by the Data FLASH Converter and do also determine the width of a Data value.

Example:

0x00	→	data width = 1 Byte
0x0012	→	data width = 2 Byte
0x00000023	→	data width = 4 Byte

- All other notation is interpreted as decimal value

Values have to be separated by either space ‘ ‘ or comma ‘,’.

2.3.2 EEPROM XML file format for V850E2

The XML input format and the corresponding tags for V850E2 based microcontrollers is shown below.

```
<?xml version="1.0"?>
<DataFlash>
        <AdministrativeSection>
                <FAL_CONFIG_DATAFLASH_SIZE>16</FAL_CONFIG_DATAFLASH_SIZE>
                <FAL_CONFIG_BLOCK_SIZE>0x800</FAL_CONFIG_BLOCK_SIZE>
                <EEL_CONFIG_BLOCK_START>2</EEL_CONFIG_BLOCK_START>
                <EEL_CONFIG_BLOCK_CNT>6</EEL_CONFIG_BLOCK_CNT>
                <EEL_CONFIG_DF_BASE_ADDRESS>0x2000000</EEL_CONFIG_DF_BASE_ADDRESS>
                <EEL_CONFIG_BLOCK_CNT_REFRESH_THRESHOLD>0x02
                </EEL_CONFIG_BLOCK_CNT_REFRESH_THRESHOLD>
                <EEL_CONFIG_ERASE_SUSPEND_THRESHOLD>10
                </EEL_CONFIG_ERASE_SUSPEND_THRESHOLD>
        </AdministrativeSection >
        <Record>
                <ID>0x80</ID>
                <IDName>Name1</IDName>
                <Length>2</Length>
                <Data>0x20 0x30</Data>
        </Record>
        <Record>
                <ID>0x10</ID>
                <IDName>Name2</IDName>
                <Length>5</Length>
                <Data>0x20 0x30 0x31 0x32 0x33</Data>
        </Record>
</DataFlash >
```

Figure 2: EEPROM emulation data, XML file format for V850E2

The predefined tags are:

DataFlash	Defining the start of the definition file.
AdminstrativeSection	This section is describing the physical parameters that are needed for the file generation.
FAL_CONFIG_DATAFLASH_SIZE	Number of Data FLASH blocks, accessible by the FAL. Typical number of available Data FLASH blocks.
FAL_CONFIG_BLOCK_SIZE	Data FLASH block size in bytes.
EEL_CONFIG_BLOCK_START	1st block of the EEL pool.
EEL_CONFIG_BLOCK_CNT	Number of blocks for the EEL pool.
EEL_CONFIG_DF_BASE_ADDRESS	Data FLASH base address.
EEL_CONFIG_BLOCK_CNT_REFRESH_THRESHOLD	Threshold for minimum number of prepared blocks.
EEL_CONFIG_ERASE_SUSPEND_THRESHOLD	Threshold for erase suspend counter. Exceeding this results in warning.

Record	Defining the start of a ID-L record.
ID	Identifier for the data record (16bit). Format is fixed to four digits in hex.
IDName	Defining the name off the record.
Length	Length definition of the data record.
Data	Any hexadecimal or decimal value. The line length is not fixed.

Comments have to be added as specified in XML definition starting with ‘<!-- ’ and ending with ‘ -->’

The values defining the ID, length or data information must have the following ‘C’ like notation:

- Leading ‘0x’ denotes a hex value.
- Leading zeros within a hex value are interpreted by the Data FLASH Converter and do also determine the width of a Data value.

Example:

0x00	→	data width = 1 Byte
0x0012	→	data width = 2 Byte
0x00000023	→	data width = 4 Byte

- All other notation is interpreted as decimal value

Values have to be separated by either space ‘ ‘ or comma ‘,’.

3. Output file / formats

The output format of the Data FLASH converter is always the same as the chosen input file format for program code mapped to the Data FLASH as well program code mapped to the Code FLASH. In case a Data FLASH image should be created from EEPROM emulation data exclusively, the output format can be chosen between Intel-hex and Motorola-S record.

4. Software Installation

The Data FLASH Converter requires Windows 2000, Windows XP or Windows Vista operating system installed on your personal computer. To start the installation, please run the “setup.exe” installation program. During initialization phase of the setup program a progress bar is shown.

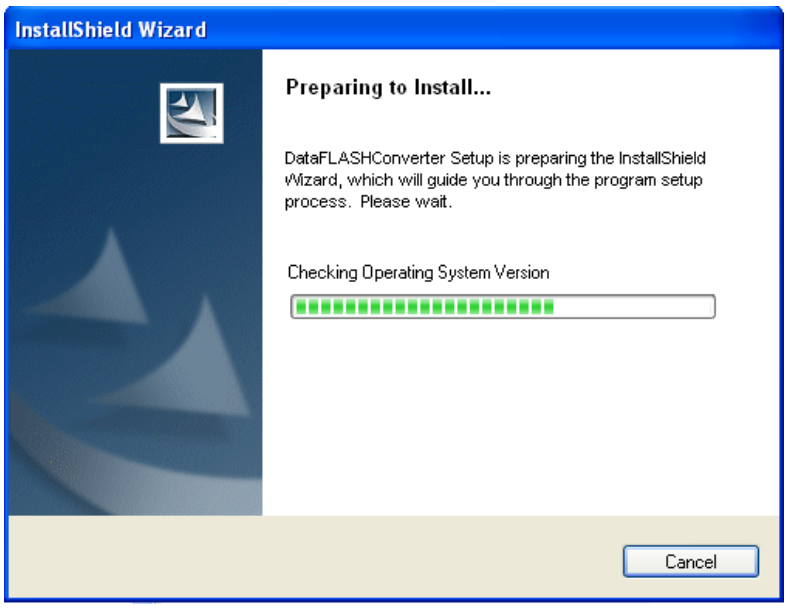

Figure 3: Install preparation

After the setup program has finished initialization a welcome screen appears. Click the ***Next*** button to continue the installation.

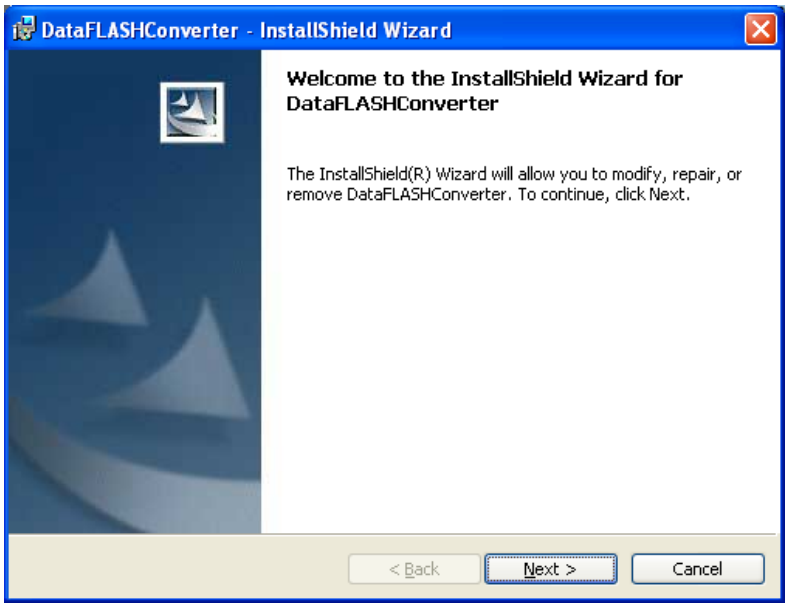

Figure 4: Welcome screen

In the next step of installation, the License Agreement window appears. Please read the Software Program License Agreement carefully. To continue the installation accept the License agreement and click the ***Next*** button. If you not accepted the license agreement the installation procedure exits.

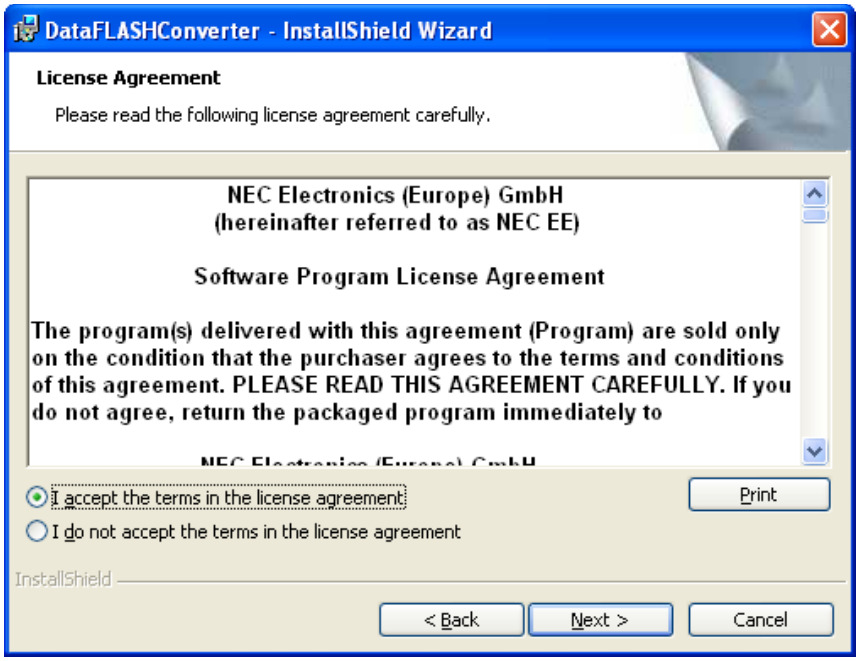

Figure 5: License Agreement window

The Destination Folder window appears. You might exit the installation by clicking the ***Cancel*** button. You can click the ***Change*** button to be able to change the installation destination path. Press the ***Next*** button to proceed with the installation.

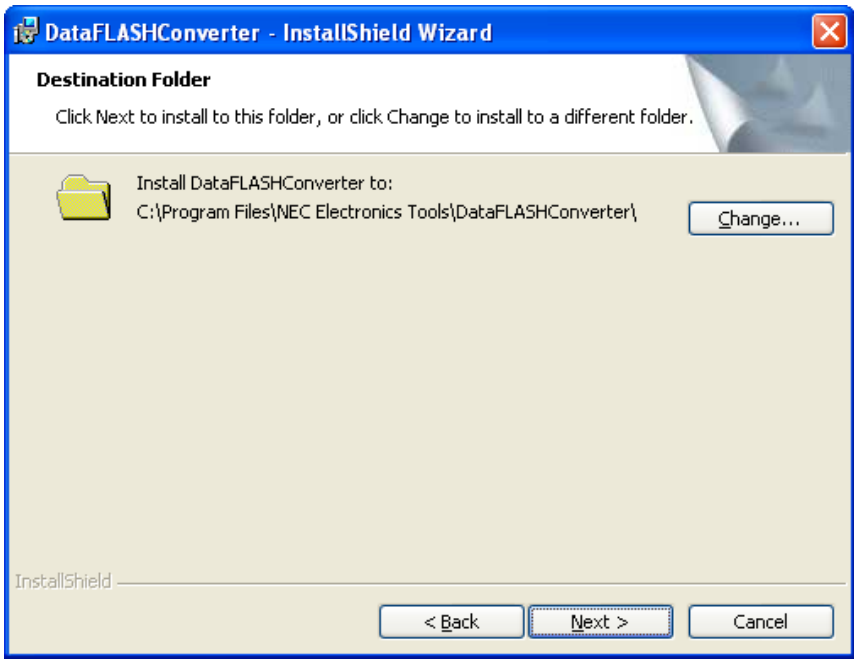

Figure 6: Choose Destination Folder window

After everything has been setup for installation, press the ***Install*** button to start the copy process and to finalize the installation.

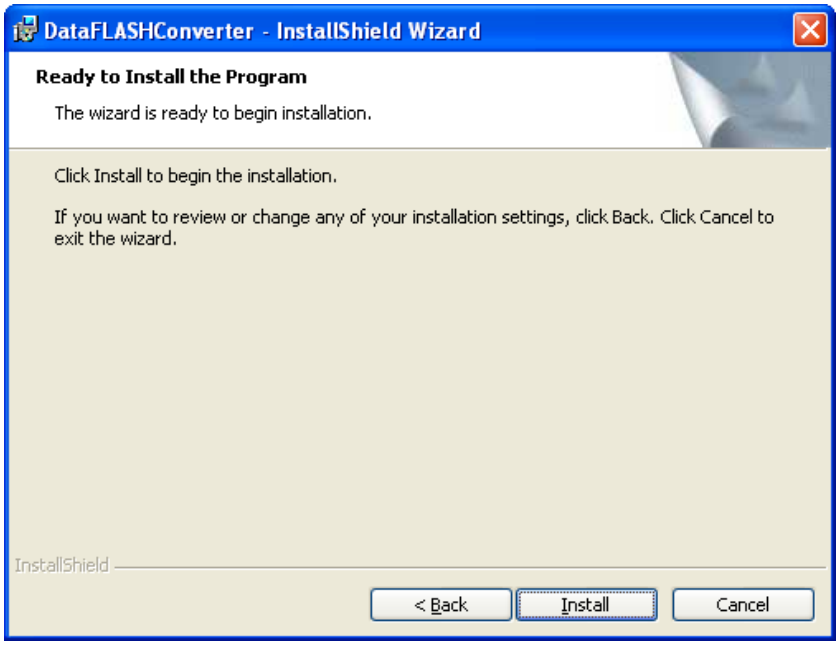

Figure 7: Ready to Install window

On the installation progress window you can follow the installation. You can always abort the installation by pressing the ***Cancel*** button.

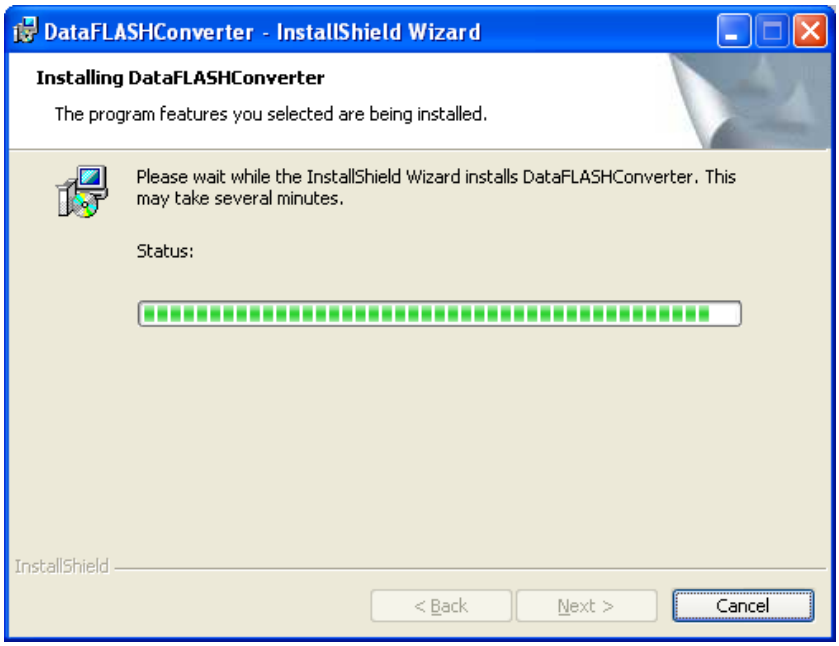

Figure 8: Installation progress window

When the installation is complete, the following message window appears.

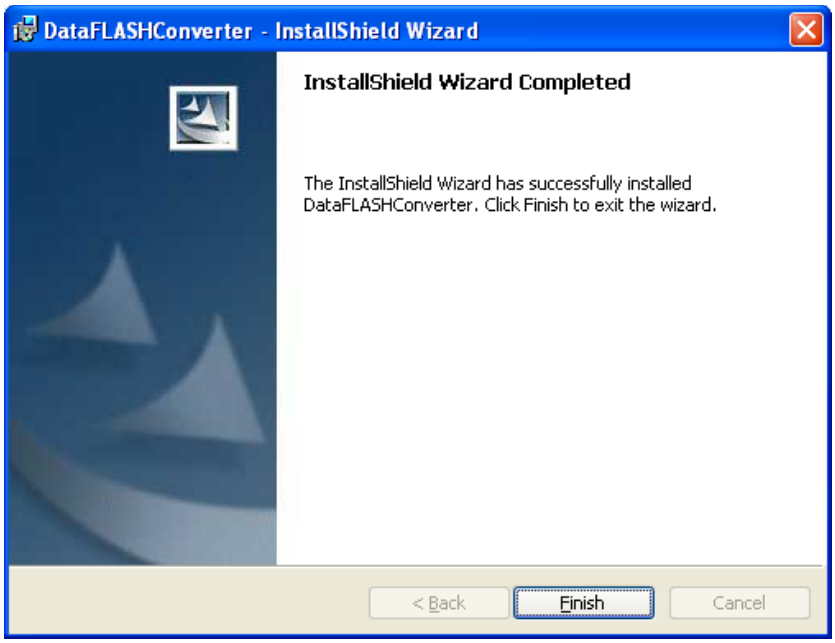

Figure 9: Installation completion window

The program has been installed successfully and an entry has been made into the program menu to start the Data FLASH Converter software.

The installation process is complete now and the setup utility has installed a new program folder which holds the Data FLASH Converter GUI Software and a Data FLASH Converter uninstall icon which allows you to remove the Data FLASH Converter software from your personal computer.

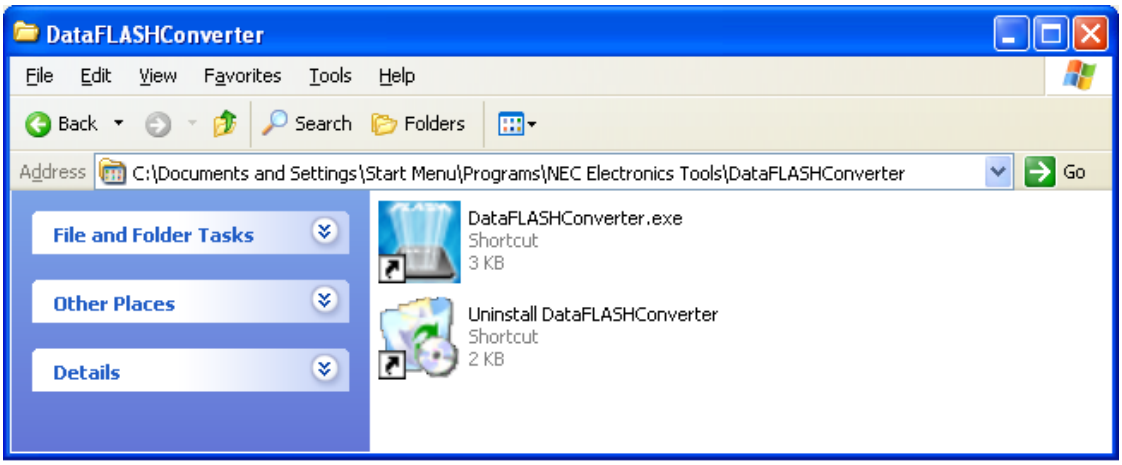

Figure 10: Program folder after installation

5. Data FLASH converter, GUI

The graphical user interface of the Data FLASH Converter allows an easy and an intuitive configuration and generation of Data FLASH memory images. When starting the Data FLASH Converter GUI a screen similar to the following one will appear. Some commands and functions may be unavailable when the GUI is started for the first time, depending on the specified device parameter file, Data FLASH area definition or the loaded program files or loaded EEPROM file.

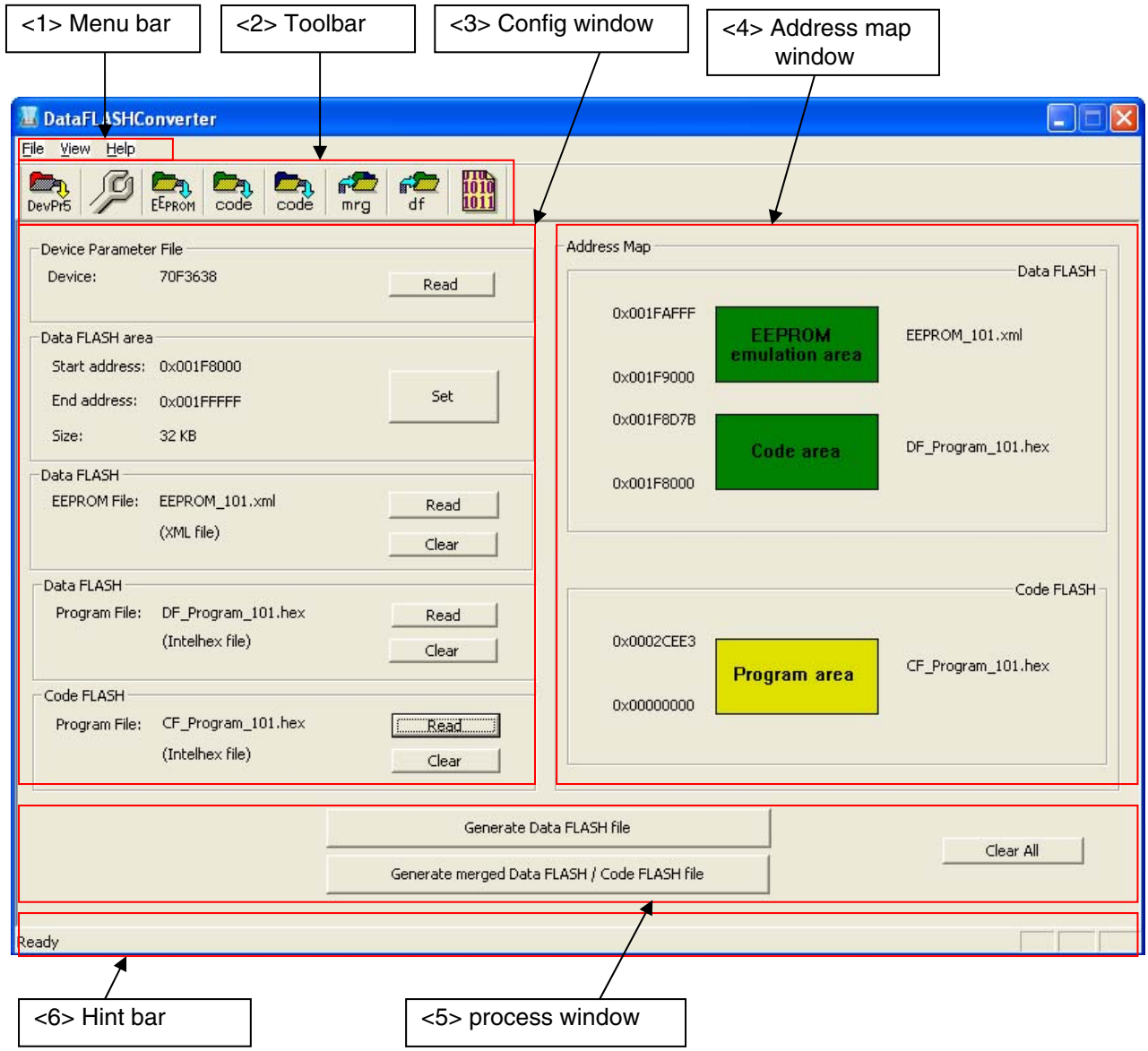

Figure 11: Data FLASH Converter main window

The main window consists of the following areas.

	Name	***Displayed items***
<1>	Menu bar	Menu items executable by the Data FLASH Converter
<2>	Toolbar	Frequently used commands, as buttons
<3>	Config window	Configuration settings
<4>	Address map window	Address map information for Code FLASH and Data FLASH memory images
<5>	Process window	Controls to start conversion, merging or reset of all configuration settings
<6>	Hint bar	Hints for commands and toolbar

Table 2: Main window composition

5.1 Toolbar

The toolbar contains buttons to configure the Data FLASH Converter and to start all key functions quickly. The following buttons and corresponding functions are offered by the Data FLASH converter.

	Open Device Parameter File
	Setup Data FLASH area
	Open EEPROM emulation data
	Open program file for Data FLASH
	Open program file for Code FLASH
	Generate merged Data / Code FLASH file
	Generate Data FLASH file
	Open HEX editor / viewer

Figure 12: Toolbar Buttons

5.2 The Menu

Analogue to the Toolbar the main menu of the Data Flash Converter allows the configuration and execution of all converter functions.

5.2.1 File menu

The ***File*** menu allows starting a HEX file editor/viewer, setting the Data FLASH area, loading a program file for Data FLASH or respectively Code FLASH and loading an EEPROM emulation file in various formats. Additional the generation of a Data FLASH file or the output of a merged Code FLASH / Data FLASH file can be controlled via the file menu.

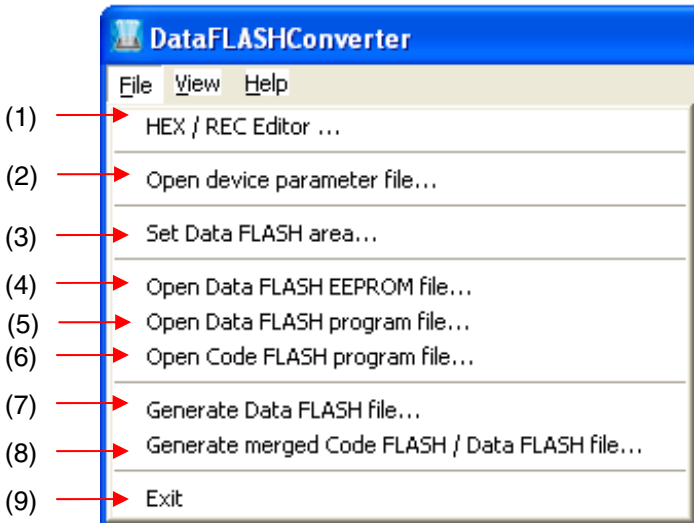

Figure 13: File menu

(1) Hex editor / viewer

The ***HEX Editor*** menu allows to edit / display a program file in Intel HEX file format or Motorola
S-Record format. A HEX editor window opens and in a file open dialog the file to be edited can be specified. Additional to this a generated Data FLASH file or a merged Code/Data FLASH file can be displayed or edited by the HEX editor. To use the HEX editor the Data FLASH area must be specified first. The display of the Data FLASH contents and ID-tag information is always based on the setting of the Data FLASH area.

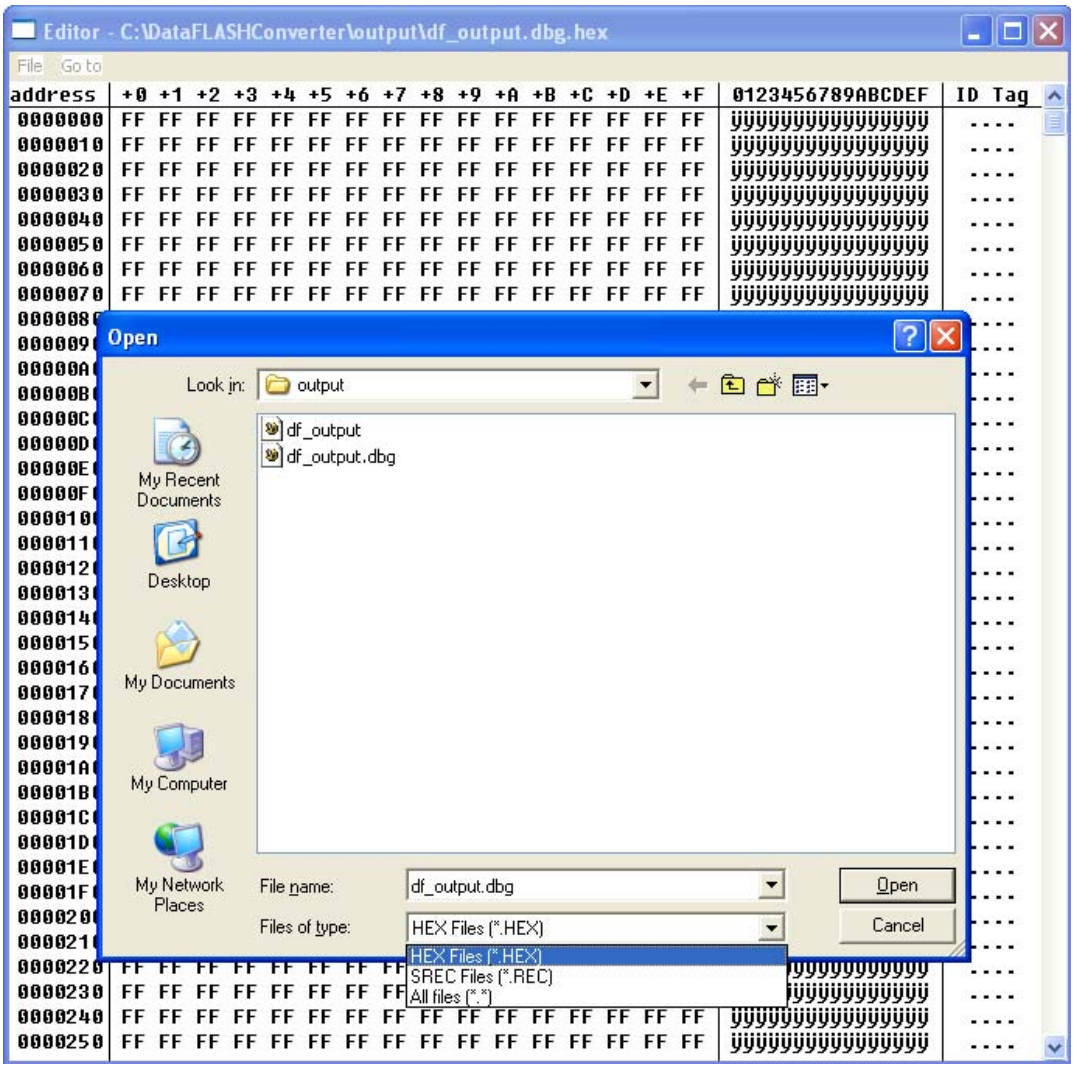

Figure 14: Hex Editor, open file dialog

HEX Files (*.HEX), SREC Files (*.REC) or All files (*.*) may be selected from the Files of types list box. After selecting a file to be opened, the file selected in the HEX Editor main window is loaded. At this time, whether the file is of the Intel HEX format or Motorola-S record format is automatically recognized. When loading is finished, the HEX Editor main window is opened.

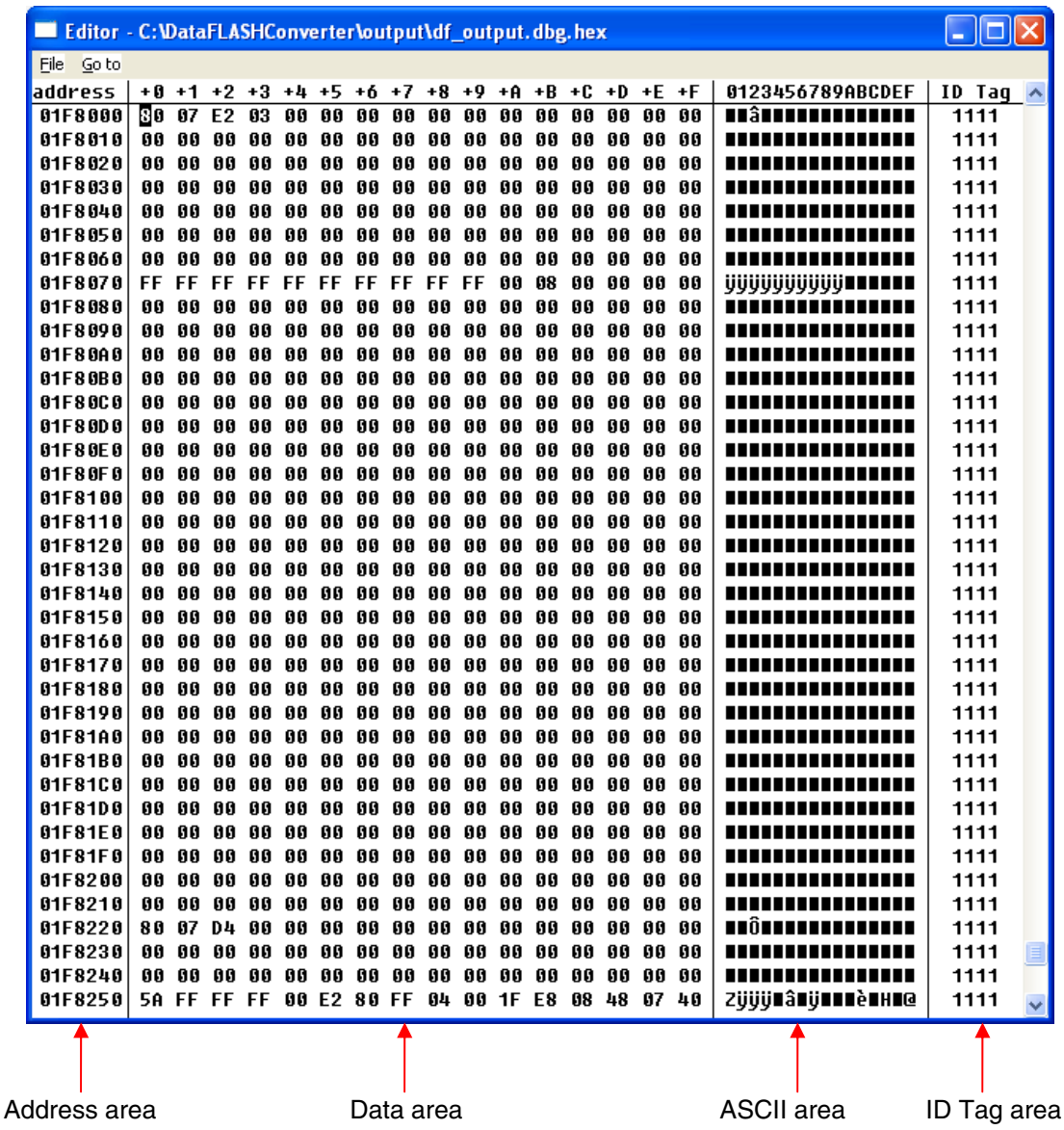

Figure 15: Hex Editor, main window

The displayed file contents can be modified by placing the cursor in the data display area in the HEX Editor main window. Data input via the keyboard is accepted for all shown memory locations.

The ID Tag area is the area used with the target device supporting Data FLASH. The ID Tag information is available only for the address space of the Data FLASH area specified within the Data FLASH Converter. Input 0 or 1 for the ID Tag area. If 0 is input, then 00000000h will be set, and if 1 is input, FFFFFFFFh will be used as the ID Tag. For details on Data FLASH and ID Tags, refer to the user's manual of the target device.

The HEX Editor only accepts hexadecimal data, i.e., numbers 0 to 9 and letters A to F. All other data will be rejected.

The ASCII representation, if any, is shown in the ASCII display area. This area is for reference only, so no data can be input.

Use the scroll bar to move another address range into the visible area of the HEX Editor.
The address space that can be viewed in the HEX Editor window is up to 4 MB + 64 KB (Data FLASH space).

The following keys can be input using the keyboard.

Key	***Function***
0-9, A-F	Data input
→	Move cursor in right direction
←	Move cursor in left direction
↑	Move cursor in up direction
↓	Move cursor in down direction
Tab	Tab Move cursor to next input field (address + 1)

Table 3: Hex Editor, navigation and input keys

If any changes have been made to the file, [Save] and [Save As] in the [File] menu in the HEX Editor main window becomes available for saving the modified data.

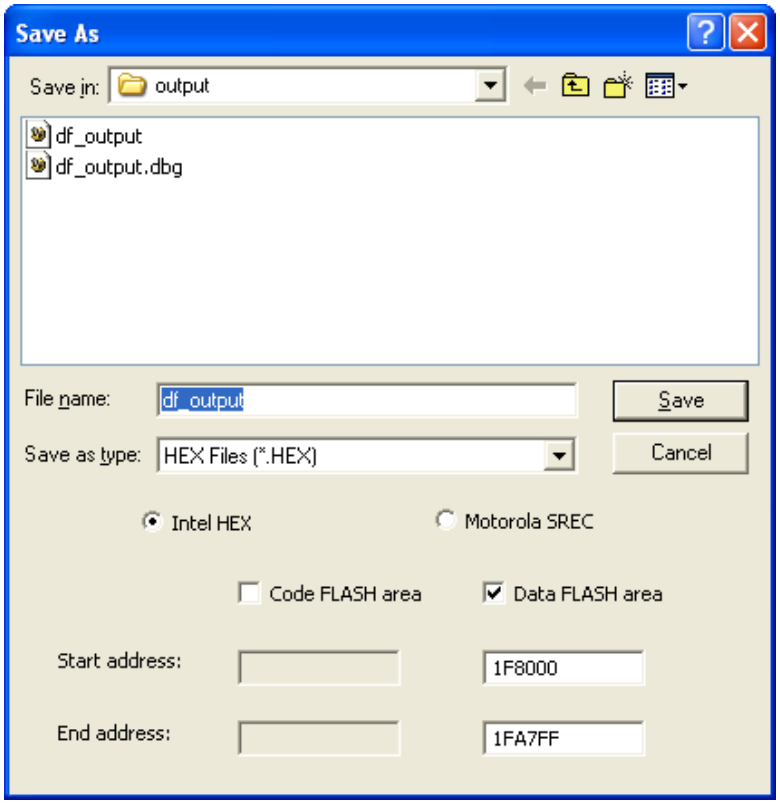

Figure 16: Hex Editor, Save as dialog

Besides the file name and folder location, a start address and an end address for the new file can be selected in the [Save As] dialog box. The original start address and end address are displayed by default. Select the file radio button for the format in which to save the data. Select the same format as that of the loaded file.

The relationship between the HEX Editor and saved File for the Data FLASH can bee seen in the figure below:

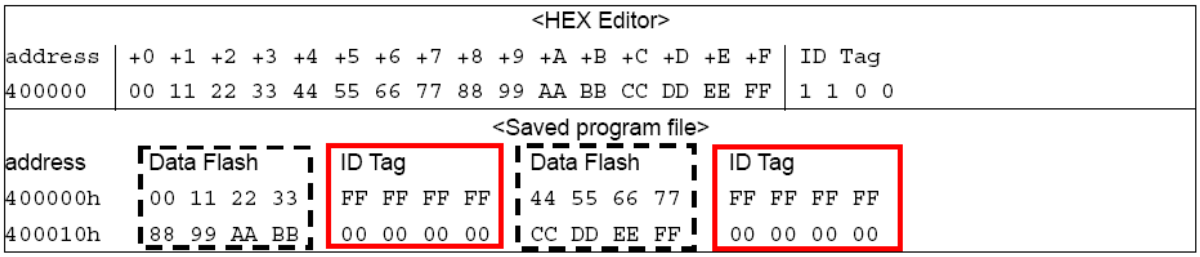

Figure 17: Hex Editor, format of saved Data FLASH file

(2) Open Device Parameter File

The ***Open Device Parameter File*** menu allows to setup the corresponding PG-FP5 FLASH programmer device parameter file. Before further settings and operations can be made within the Data FLASH Converter GUI the corresponding "*.pr5" file has to be specified.

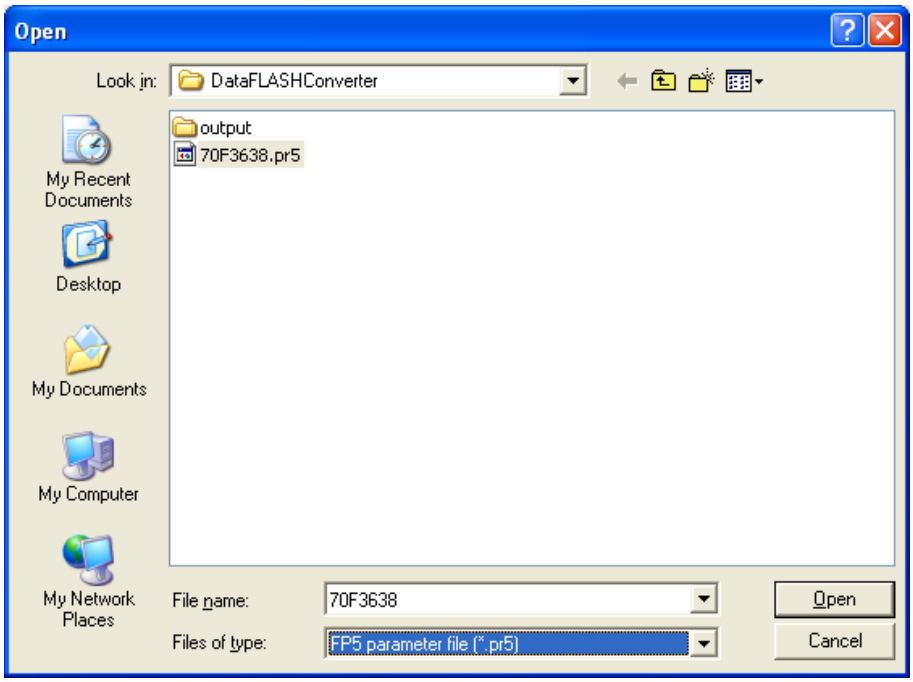

Figure 18: Open Device Parameter File

V850E2:

For V850E2 based devices the start address, end address and size of the Data FLASH area is defined by the device parameter file. In case of V850E2 type microcontrollers the start address of the Data FLASH is fixed to address 0x2000000. According to this, the Data FLASH area can not be changed within the "Set Data FLASH area" menu.

V850E1/ES:

For V850E1/ES based devices the start address of the Data FLASH area is not fixed. It can be changed device depended to different chip select areas. Within the Setup Data FLASH dialog the start address of the Data FLASH must be specified. The entered start address should be equal to the physical start address of the Data FLASH area of the corresponding V850E1/ES device. The size of the Data FLASH can be specified by the pull-down menu. Supported sizes are 32 Kbyte and 64 Kbyte.

(3) Set Data FLASH area

The ***Set Data FLASH area*** menu allows to setup the start address and the size of the Data FLASH area. This menu is only available for V850E1/ES based devices. For V850E2 type based devices the Data FLASH area is fixed to address 0x2000000 and can not be changed. By selecting this menu item the following dialog window appears and offers the possibility to specify the Data FLASH mapping.

Figure 19: Set Data FLASH, start address

Within the Setup Data FLASH dialog the start address of the Data FLASH must be specified. Only hexadecimal input values are permissible. The entered start address should be equal to the physical start address of the Data FLASH area of the corresponding V850E1/ES device. The size of the Data FLASH can be specified by the pull-down menu. Supported sizes are 32 Kbyte and 64 Kbyte.

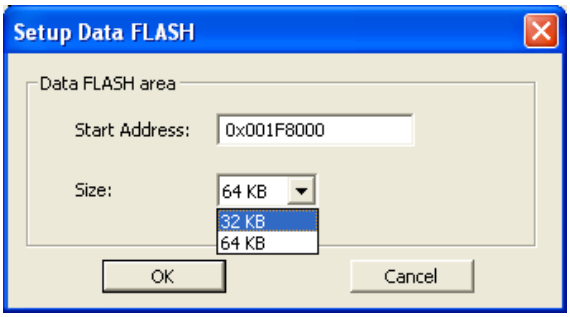

Figure 20: Set Data FLASH, size

By closing the dialog pressing the <OK> button, the specified memory area is displayed in the Data FLASH area box of the main window.

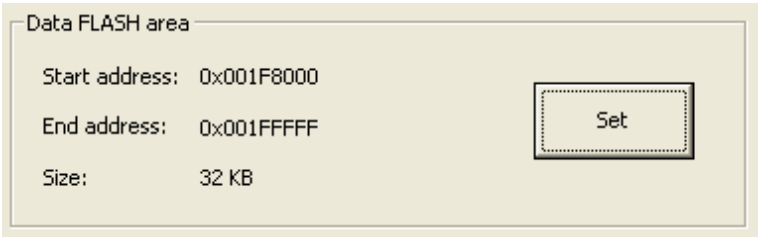

Figure 21: Config Window, Data FLASH area box

Additional to the file menu and toolbox button the Data FLASH area can be specified by pressing the <Set> button within the Data FLASH area box.

(4) Open Data FLASH EEPROM file

The ***Open Data FLASH EEPROM file*** menu allows selecting and loading an EEPROM emulation data file by the Data FLASH Converter. After downloading the EEPROM emulation data the corresponding memory area within the Data FLASH is shown in the address map window.

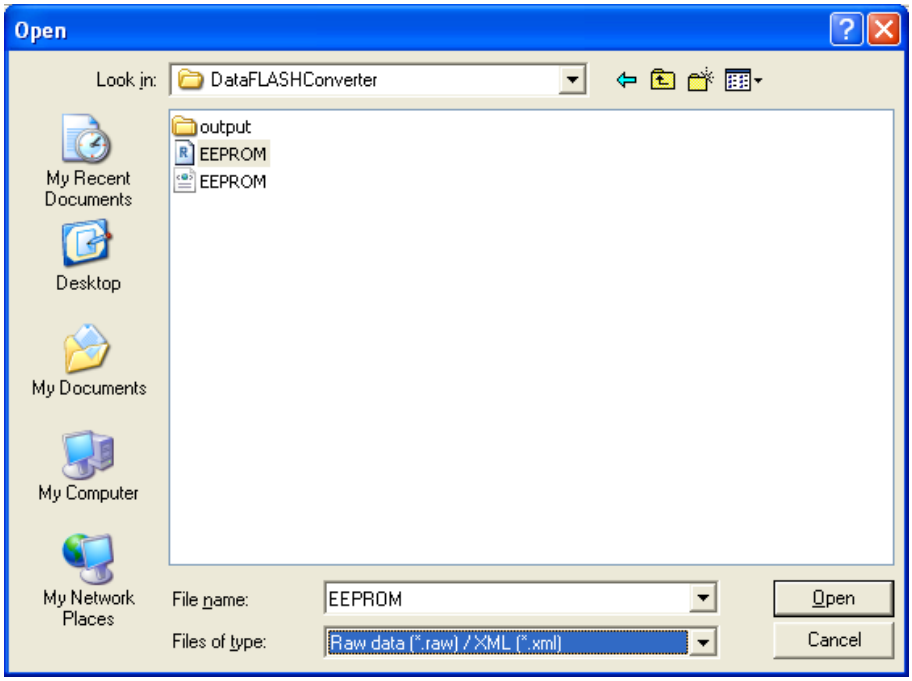

Figure 22: Open Data FLASH EEPROM file dialog

The most recently used directory a file has been loaded from will be offered in this file open menu. EEPROM emulation data can be loaded in the XML file format.

After EEPROM emulation data is loaded the corresponding file and file format is displayed in the Config window of the Data FLASH converter.

Figure 23: Config Window, Data FLASH EEPROM file box

Additional to the file menu and toolbox button the EEPROM emulation data can be loaded by pressing the <Read> button within the Config window of the Data FLASH Converter. The<Clear> button offers the possibility to delete a current selected file.

To get information about the needed Data FLASH memory for the EEPROM emulation data, the mapping is shown in the address map window of the Data FLASH converter.

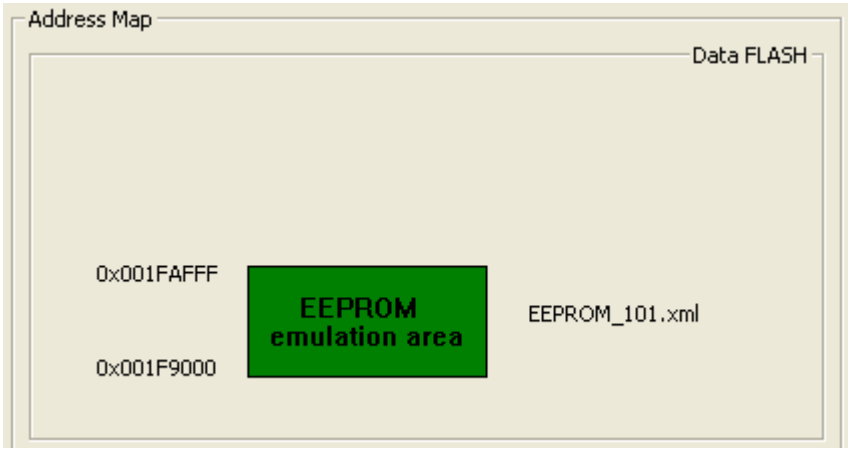

Figure 24: Address map window, Data FLASH memory box (1)

(5) Open Data FLASH program file

The ***Open Data FLASH program file*** menu allows selecting and loading a program file that is linked to be executed from the Data FLASH memory. After downloading the program file the corresponding memory area where the program is placed within the Data FLASH is shown in the address map window.

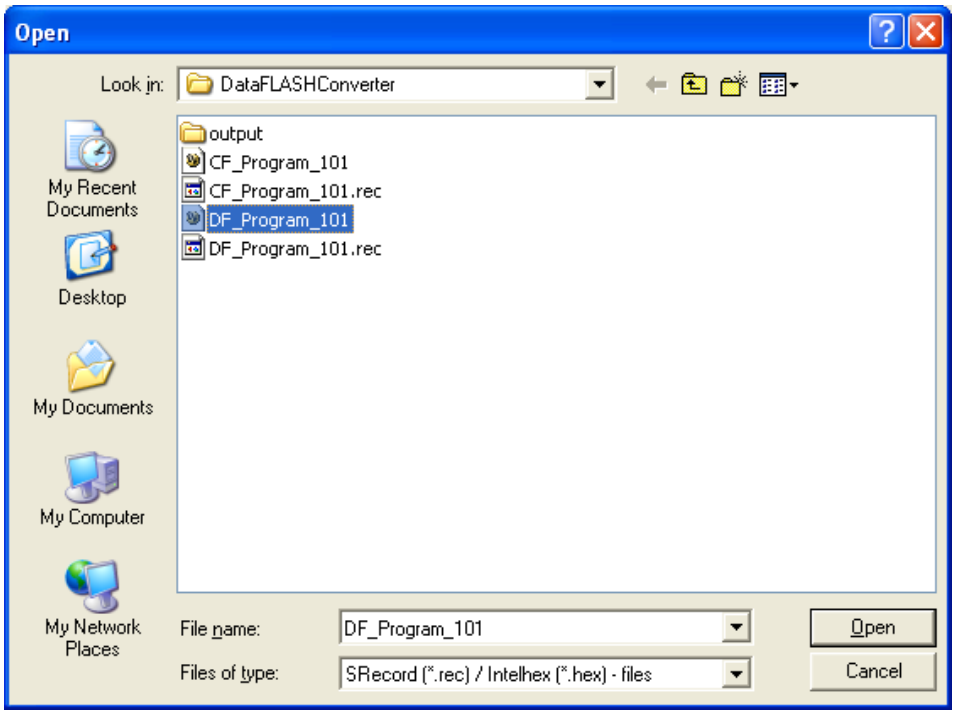

Figure 25: Open Data FLASH program file dialog

The program code has to be available in an Intel-hex or Motorola-S record format. For Intel-hex the extended linear address record and extended segment address record formats are supported. With regard to the Motorola-S record type, the S2 and S3 record formats are supported for program code mapped to the Data FLASH memory.

After the program code is loaded the corresponding file and file format is displayed in the Config window of the Data FLASH converter.

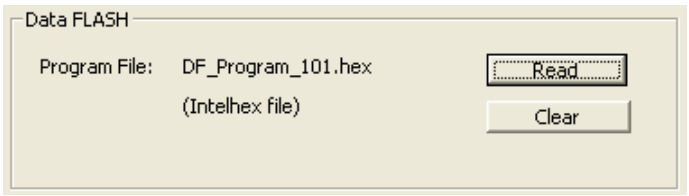

Figure 26: Config Window, Data FLASH Program file box

Additional to the file menu and toolbox button the program file can be loaded by pressing the <Read> button within the Config window of the Data FLASH Converter. The<Clear> button offers the possibility to delete a current selected file.

To get information about the address map of the program file, the mapping is shown in the address map window of the Data FLASH converter.

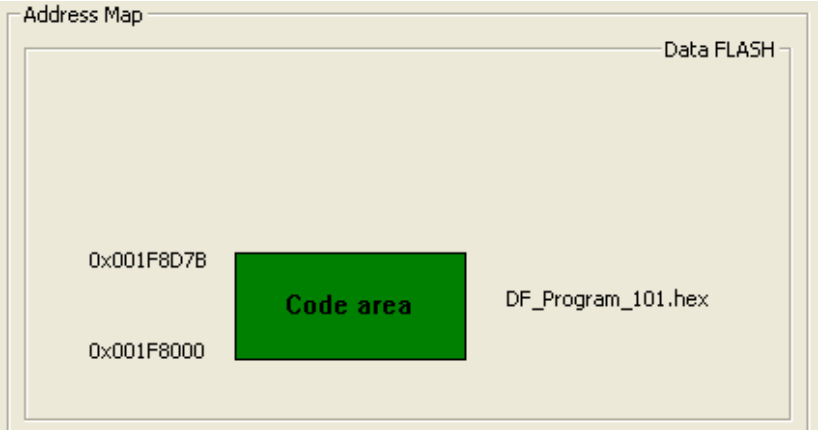

Figure 27: Address map window, Data FLASH memory box (2)

(6) Open Code FLASH program file

The ***Open Code FLASH program file*** menu allows selecting and loading a program file that is linked to be executed from the Code FLASH memory. After downloading the program file the corresponding memory area where the program is placed within the Code FLASH is shown in the address map window.

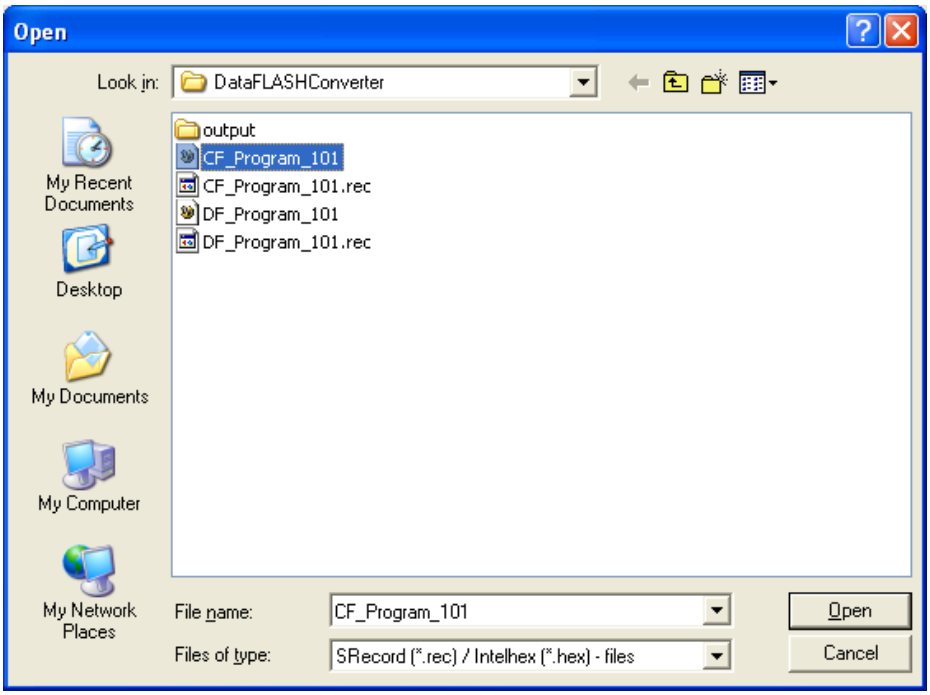

Figure 28: Open Code FLASH program file dialog

The program code has to be available in an Intel-hex or Motorola-S record format. For Intel-hex the extended linear address record and extended segment address record formats are supported. With regard to the Motorola-S record type, the S1, S2 and S3 record formats are supported for program files mapped to the Code FLASH.

After the program code is loaded the corresponding file and file format is displayed in the Config window of the Data FLASH converter.

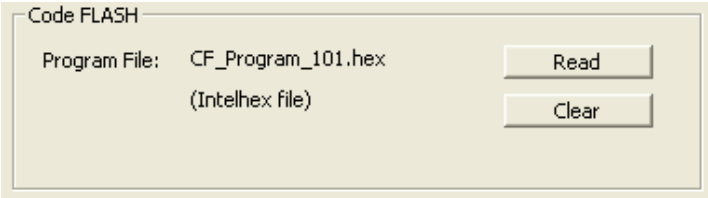

Figure 29: Config Window, Code FLASH Program file box

Additional to the file menu and toolbox button the program file can be loaded by pressing the <Read> button within the Config window of the Data FLASH Converter. The<Clear> button offers the possibility to delete a current selected file.

To get information about the address map of the program file, the mapping is shown in the address map window of the Data FLASH converter.

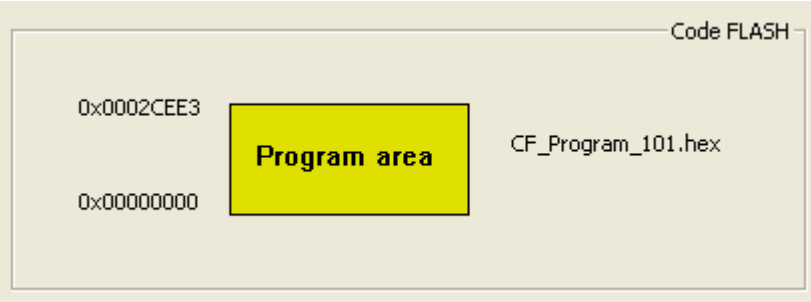

Figure 30: Address map window, Code FLASH memory box

(7) Generate merged Data FLASH / Code FLASH file

The ***Generate merged Data FLASH / Code FLASH file*** menu allows to generate a merged output file including Data FLASH and Code FLASH information. The file format of the generated file depends on the format of the loaded program file for the Data FLASH respectively Code FLASH.
By executing this command a file save dialog opens and offers the possible to specify the name and location of the output file to be generated.

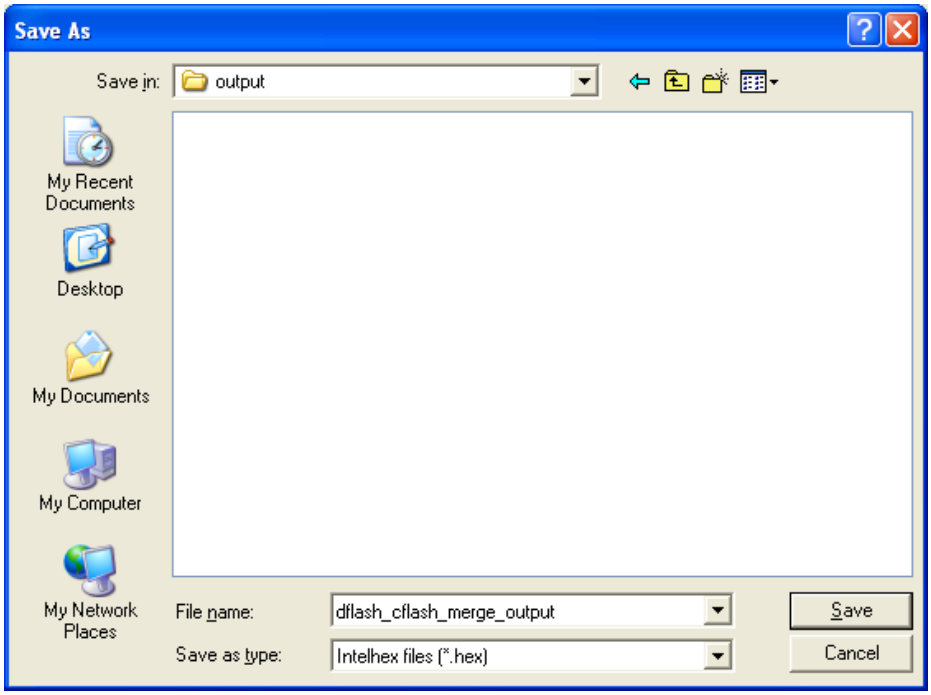

Figure 31: Save merged Data FLASH / Code FLASH file dialog

This merged output file can be read by the standard NEC FLASH programmer PG-FP5 and can be used for FLASH programming of a V850 based device containing Code FLASH and Data FLASH.

Note for V850E1/ES: The NEC FLASH programmer uses the serial programming mode for programming the Data FLASH memory of a V850E1/ES device. Within this mode the Data FLASH memory is accessed for reading and writing via the FLASH Control Block macro (FCB) of the corresponding V850E1/ES device. By writing or reading via the FCB the start address of the Data FLASH memory is always 0x0400000 regardless of the used V850E1/ES device. This address differs to the physical mapping of Data FLASH within normal mode, where the Data FLASH memory is accessed via the instruction fetch or data bus of the V850E1/ES device.

Note: A merged Data FLASH / Code FLASH file can not be used for debugging purpose.

(8) Generate Data FLASH file

The ***Generate Data FLASH file*** menu allows to generate a output file including Data FLASH information only. In case EEPROM emulation data and program code for Data FLASH are specified together a single output file is generate representing the complete Data FLASH memory image. Within this case, the file format of the generated output file depends on the format of the loaded Data FLASH program file. By exclusively converting EEPROM emulation data the format of the output file can be chosen between Intel-hex and Motorola-S record.
By executing the ***Generate Data FLASH file*** command a file save dialog opens and offers the possible to specify the name and location of the output file to be generated.

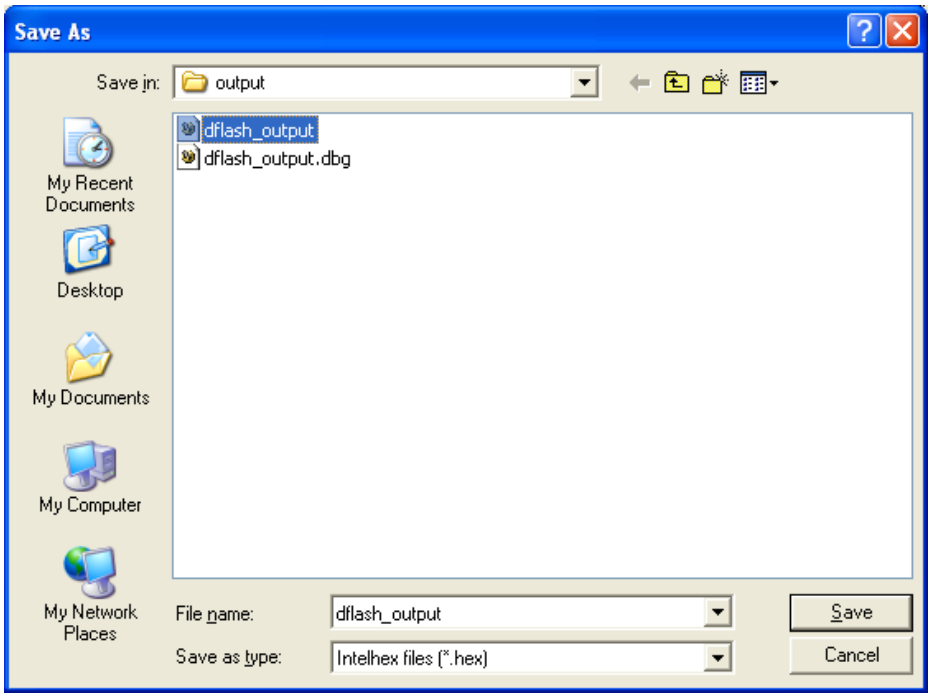

Figure 32: Save Data FLASH file dialog

V850E1/ES:

By generating a Data FLASH file two separate output files are generated, one for FLASH programming and the other one for debugging purpose. The file for debugging purpose includes the marking "**.dbg**" within the file name and is mapped to the Data FLASH memory area specified by the user. The file for FLASH programming purpose has no additional marking within its name and is mapped to address 0x0400000. Please refer also to Figure 35.

Background: The NEC FLASH programmer uses the serial programming mode for programming the Data FLASH memory of a V850E1/ES device. Within this mode the Data FLASH memory is accessed for reading and writing via the FLASH Control Block macro (FCB) of the corresponding V850E1/ES device. By writing or reading via the FCB the start address of the Data FLASH memory is always 0x0400000 regardless of the used V850E1/ES device. This address differs to the physical mapping of Data FLASH used within normal mode or expected during debugging, where the Data FLASH memory is accessed via the instruction fetch or data bus of the V850E1/ES device.

V850E2:

By generating a Data FLASH file one output file will be generated, which can be used for FLASH programming and for debugging purpose. For V850E2 type based devices the Data FLASH area is fixed to address 0x2000000. Additionally to this, two header files for the EEL will be generated. The name of the header files is fixed to "*EEL_Descriptor.h" and* "*FDL_Descriptor.h".*

EEL_Descriptor.h = Header file for the EEPROM Emulation Library for NEC V850 devices
FDL_Descriptor.h = Header file for the Data Flash Access Library for NEC V850 devices

5.3 Process window

The process window offers additional to the file menu and the toolbar the possibility to generate a Data FLASH file or a merged Code FLASH / Data FLASH file.

Supplementary, the complete Data FLASH converter settings can be cleared by using the <Clear All> button.

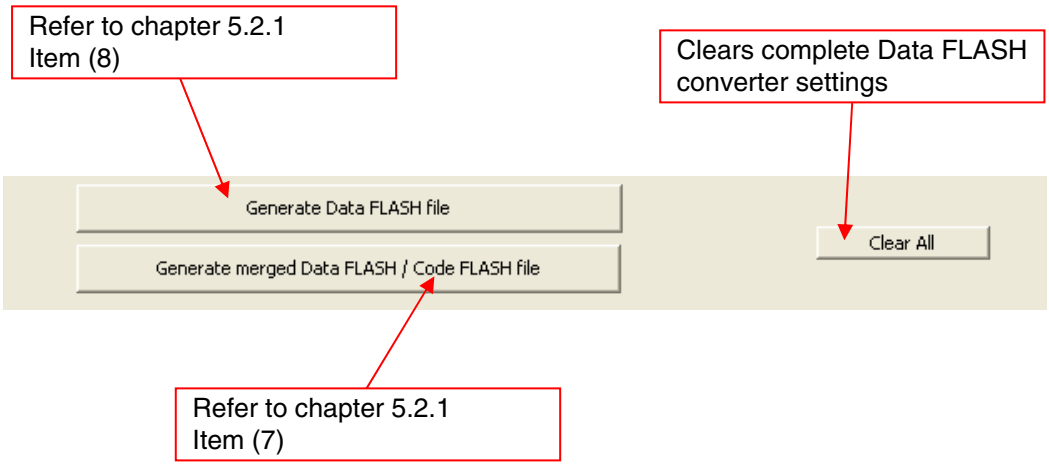

Figure 33: Process window

6. Data FLASH converter, command line interface

Beside the graphical user interface the Data FLASH converter provides a command line interface to support batch mode operation or automatic output file generation. The syntax for the command line interface is as follows:

DataFLASHConverter -[data flash mapping] <*options*> -[input files] <*filename*>
-[output file] <*filename*>

By executing the Data FLASH converter without any options from the command line the graphical user interface is started.

By doing the conversion via the command line interface a separate command shell window will open, showing all necessary information, the input options / settings, the conversion processing and finally the conversion result.

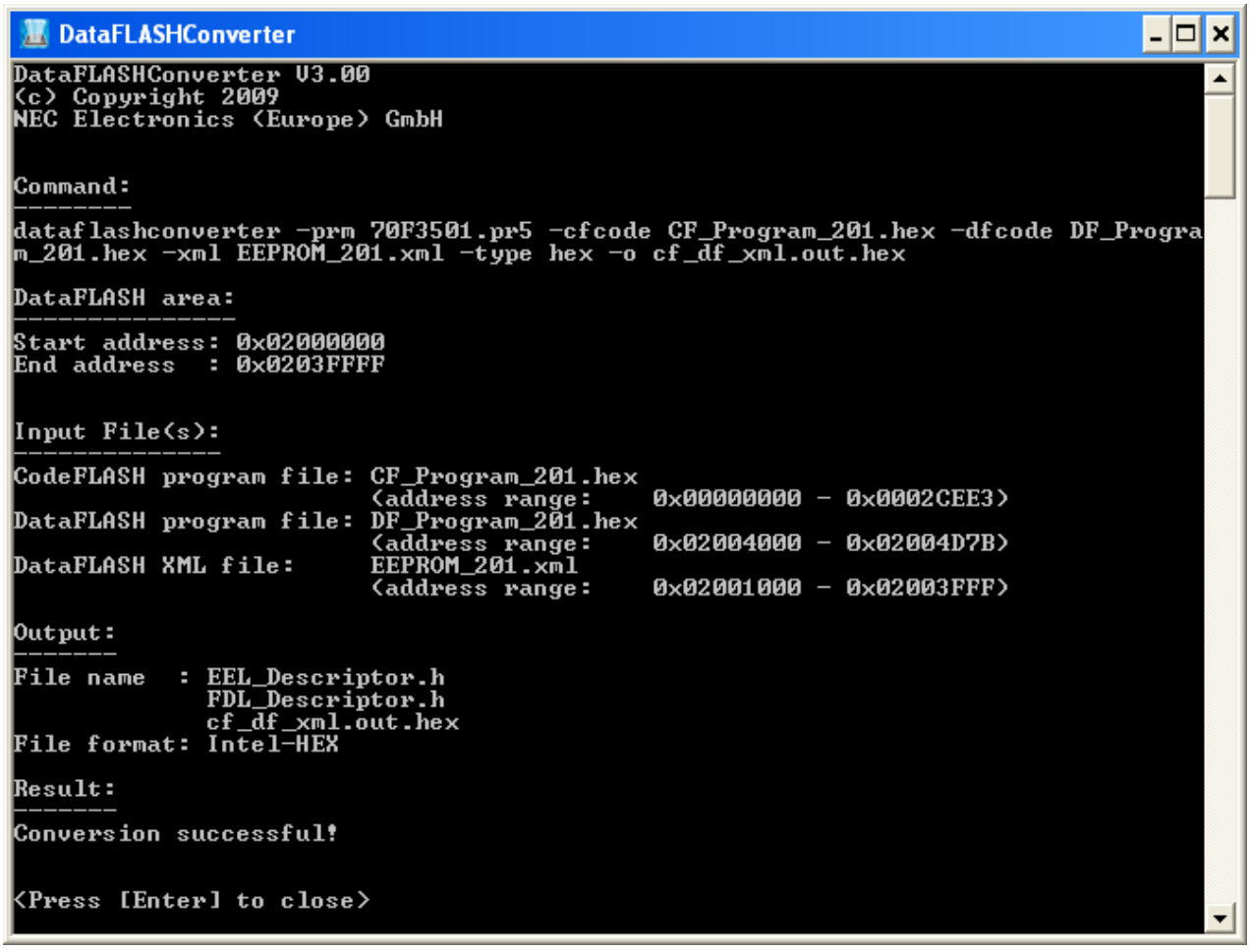

Figure 34: Data FLASH Converter, command line shell

6.1 Command line options

The following command line options are supported by the Data FLASH Converter:

Command	*Definition*
-prm <file>	specifies the device parameter file “*.pr5”
-dfbase <address>	specifies the physical start address of the Data FLASH memory. The address has be specified in hexadecimal notation, i.e. 0x001F8000
-dfsize < 32KB / 64KB >	specifies the size of the Data FLASH memory. Supported sizes are 32 kByte and 64 kByte
-xml <file>	specifies the Data FLASH EEPROM file (XML format)
-type <srec / hex>	specifies the input format for the program code files and the output file format
-o <file>	specifies the output file
-cfcode <file>	specifies the program file for the Code FLASH
-dfcode <file>	specifies the program file for the Data FLASH
-help	displays help information (this command list)
-log <file>	specifies the log file to be created. By using this option an separate log file is created showing all necessary information, the input options / settings, the memory mapping and finally the conversion result.
-nodisplay	specifies that the converter should run in silent mode, no additional command shell window will be opened. Please use this mode within an automated generation or make flow.

Table 4: Command line options

6.2 Combination of command line options

The following combinations of command line options are supported by the Data FLASH converter. They are subdivided in command line options for V850E1/ES and V850E2 based microcontrollers:

V850E1/ES: For each different operation it is a must to specify the device parameter file (command "**-prm** *<file>*"), Data FLASH mapping (commands "**-dfbase** *<address>*" and "**-dfsize** *<size>*"), the file type (command "**-type** *<type>*") and the output file (command "**-o** *<file>*"). For a clear arrangement they will not be referenced in the combinations below. The order on how the options have to be specified within the command line is not prescribed.

-prm *<file>* **-dfbase** *<address>* **-dfsize** *<size>* **-type** *<type>* … **-o** *<file>*

V850E2: For each different operation it is a must to specify the device parameter file (command "**-prm** *<file>*"), the file type (command "**-type** *<type>*") and the output file (command "**-o** *<file>*"). For V850E2 type based devices the Data FLASH area is fixed to address 0x2000000, therefore the address and size has not to be specified. For a clear arrangement they will not be referenced in the combinations below. The order on how the options have to be specified within the command line is not prescribed.

-prm *<file>* **-type** *<type>* … **-o** *<file>*

The following options for the output file generation are common for the V850E1/ES and V850E2 type based microcontrollers.

6.2.1 Generation of merged Data FLASH / Code FLASH file

Generates a merged output file including Data FLASH and Code FLASH information. The file format of the generated file depends on the format of the loaded program file for the Data FLASH respectively Code FLASH and must fit to the specified file type.

(1) Generation of output file including Code FLASH, Data FLASH program and EEPROM emulation data

… -cfcode *<file>* **-dfcode** *<file>* **-xml** *<file> …*

(2) Generation of output file including Code FLASH program and EEPROM emulation data

… -cfcode *<file>* **-xml** *<file> …*

(3) Generation of output file including Code FLASH program and Data FLASH program data

… -cfcode *<file>* **-dfcode** *<file> …*

This merged output file can be read by the standard NEC FLASH programmer PG-FP5 and can be used for FLASH programming of a V850 based device containing Code FLASH and Data FLASH.

Note for V850E1/ES: The NEC FLASH programmer uses the serial programming mode for programming the Data FLASH memory of a V850E1/ES device. Within this mode the Data FLASH memory is accessed for reading and writing via the FLASH Control Block macro (FCB) of the corresponding V850E1/ES device. By writing or reading via the FCB the start address of the Data FLASH memory is always 0x0400000 regardless of the used V850E1/ES device. This address differs to the physical mapping of Data FLASH within normal mode, where the Data FLASH memory is accessed via the standard instruction fetch or data bus of the V850E1/ES device.

Note: A merged Data FLASH / Code FLASH file can not be used for debugging purpose.

6.2.2 Generation of Data FLASH file

Generates a output file including Data FLASH information only. In case EEPROM emulation data and program code for Data FLASH are specified together a single output file is generate representing the complete Data FLASH memory image. Within this case, the file format of the generated output file depends on the format of the loaded Data FLASH program file and must fit to specified file type. By exclusively converting EEPROM emulation data the format of the output file can be chosen between Intel-hex and Motorola-S record by using the file type option.

(1) Generation of output file including Data FLASH program and EEPROM emulation data

… -dfcode *<file>* **-xml** *<file> …*

(2) Generation of output file including Data FLASH program data

… -dfcode *<file>* **…**

(3) Generation of output file including EEPROM emulation data

… -xml *<file> …*

V850E1/ES:

By generating a Data FLASH file two separate output files were generated, one for FLASH programming and the other one for debugging purpose. The file for debugging purpose includes the marking "**.dbg**" within the file name specified by the "**-o**" option and is mapped to the Data FLASH memory area specified by the user. The file for FLASH programming purpose has no additional marking within its name and is mapped to address 0x0400000.

V850E2:

By generating a Data FLASH file one output file will be generated, which can be used for FLASH programming and for debugging purpose. For V850E2 type based devices the Data FLASH area is fixed to address 0x2000000. Additionally to this, two header files for the EEL will be generated. The name of the header files is fixed to "*EEL_Descriptor.h" and* "*FDL_Descriptor.h".*

EEL_Descriptor.h = *Header file for the* EEPROM Emulation Library for NEC V850 devices
FDL_Descriptor.h = *Header file for the* Data Flash Access Library for NEC V850 devices

Example for V850E1/ES:

```
DataFlashConverter -prm 70F3638.pr5 -dfbase 0x01F8000
                   -dfsize 32kb -type hex
                   -xml EEPROM.xml -o dataflashmem.hex
```

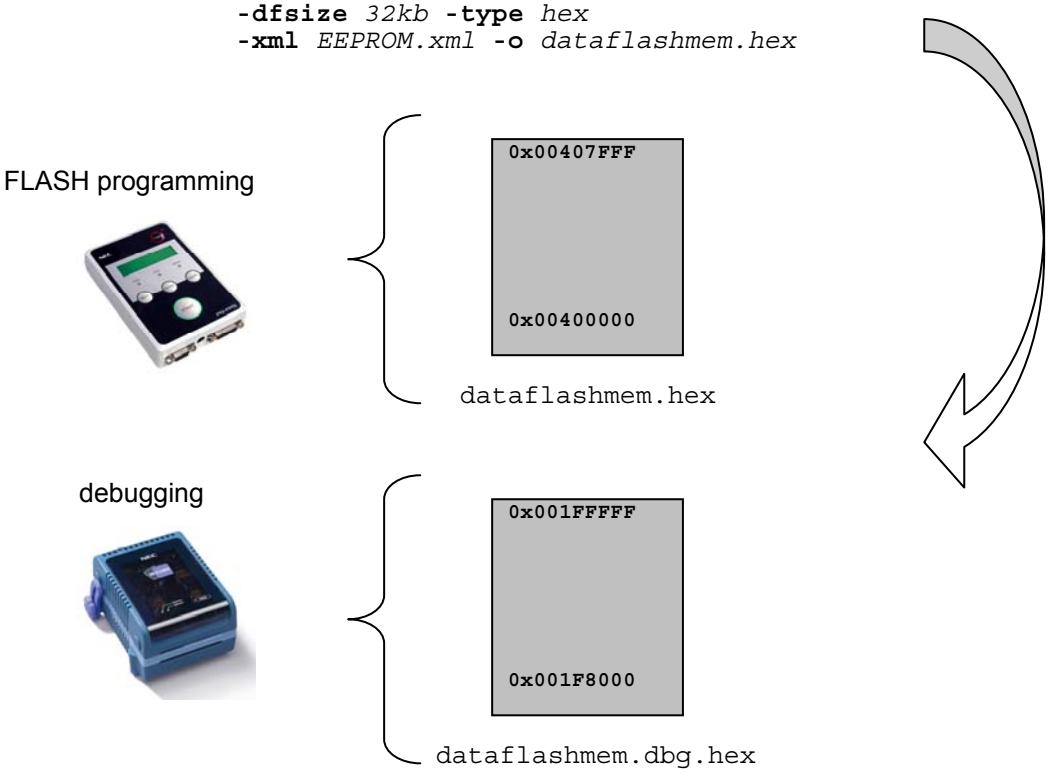

Figure 35: Dual output generation for Data FLASH memory image (V850E1/ES)

Background: The NEC FLASH programmer uses the serial programming mode for programming the Data FLASH memory of a V850E1/ES device. Within this mode the Data FLASH memory is accessed for reading and writing via the FLASH Control Block macro (FCB) of the corresponding V850E1/ES device. By writing or reading via the FCB the start address of the Data FLASH memory is always 0x0400000 regardless of the used V850E1/ES device. This address differs to the physical mapping of Data FLASH used within normal mode or expected during debugging, where the Data FLASH memory is accessed via the standard instruction fetch or data bus of the V850E1/ES device.

Example for V850E2:

```
DataFlashConverter -prm 70F3537.pr5 -type hex
                   -xml EEPROM.xml -o dataflashmem.hex
```

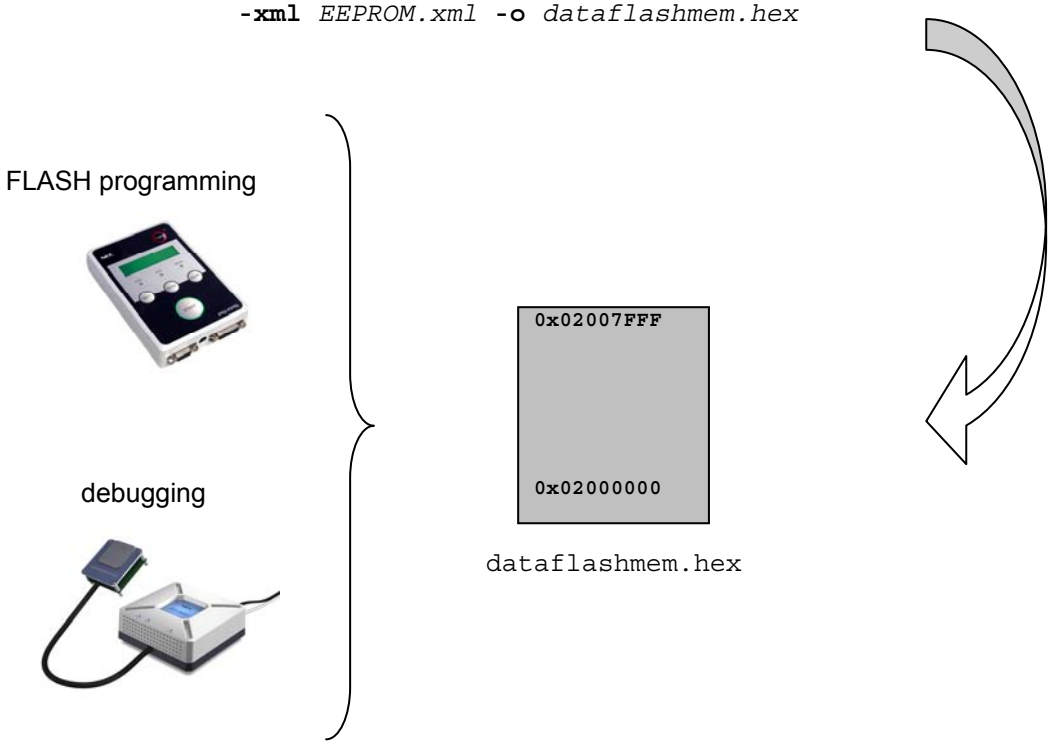

Figure 36: Output generation for Data FLASH memory image (V850E2)

By generating a Data FLASH file one output file will be generated, which can be used for FLASH programming and for debugging purpose. For V850E2 type based devices the Data FLASH area is fixed to address 0x2000000. Additionally to this, two header files for the EEL will be generated. The name of the header files is fixed to "*EEL_Descriptor.h" and* "*FDL_Descriptor.h".*

EEL_Descriptor.h *= Header file for the* EEPROM Emulation Library for NEC V850 devices
FDL_Descriptor.h *= Header file for the* Data Flash Access Library for NEC V850 devices

6.3 Return Codes

For using the Data FLASH converter within an automated generation or make flow, different return codes are generated depending on the conversion result. In case a conversion was successful the Data FLASH converter returns the value zero. Otherwise in case of errors during conversion the Data FLASH converter returns the corresponding error number. A detailed list of all error numbers can be found in chapter 7 of this document.

7. Error Messages

Error messages will be output in the error dialog boxes when using the graphical user interface of Data FLASH Converter.

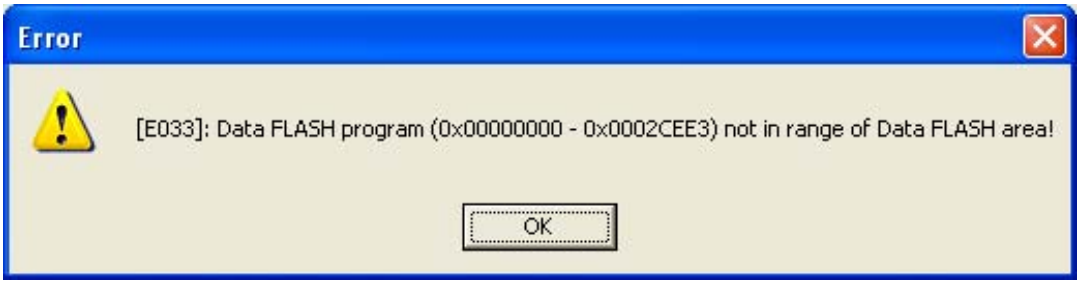

Figure 37: Error message, dialog box

In case by using the command line interface the error message is displayed in the command shell window. Additional the error message is written to the corresponding log file if specified (option **"-log** *<file>*").

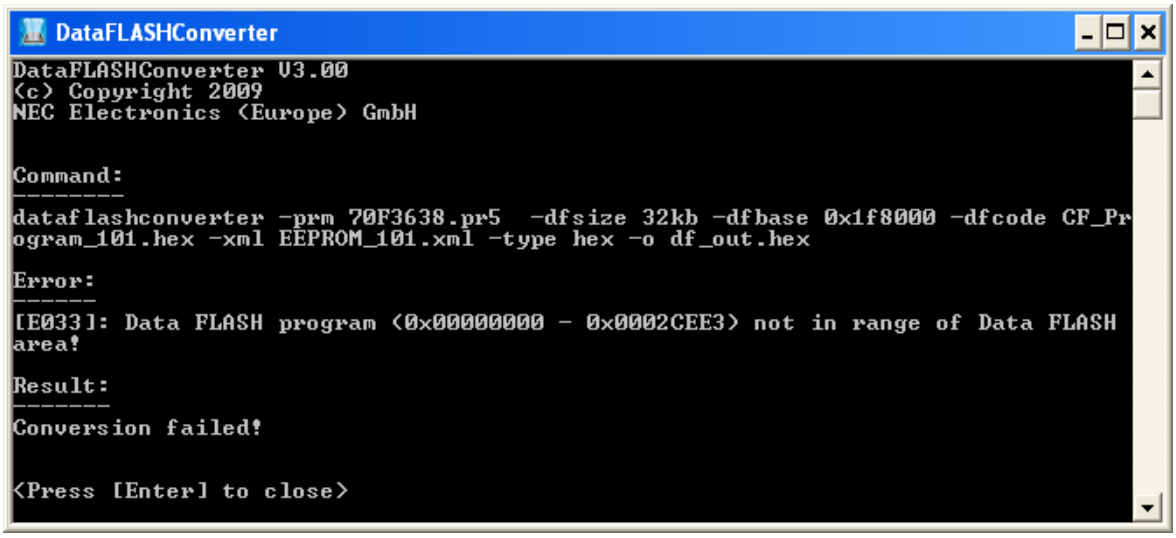

Figure 38: Error message, command shell window

Number	*Message / Description*
[E001]	Could not create output file!
[E002]	Could not open Data FLASH EEPROM file!
[E003]	Could not open Code FLASH Program file!
[E004]	Could not open Data FLASH Program file!
[E005]	Illegal combination of command line options!
[E006]	Illegal command line option detected!
[E007]	Data FLASH EEPROM file. Address is not in range of Data FLASH area!
[E008]	Please specify correct file type (SREC/HEX)!
[E009]	Data FLASH program file is not of type HEX!
[E010]	Data FLASH program file is not of type SREC!
[E011]	Code FLASH program file is not of type HEX!
[E012]	Code FLASH program file is not of type SREC!
[E013]	Data FLASH EEPROM file has invalid file format/type! Only XML supported!
[E014]	Data FLASH program file has invalid file format/type! Only SREC/HEX supported!
[E015]	Code FLASH program file has invalid file format/type! Only SREC/HEX supported!
[E016]	Data FLASH EEPROM file. Length validation error! Data length must be a size 4 * n. (with n > 0!) Affected record ID: <id>
[E017]	Data FLASH EEPROM file. Data length mismatch error! Given length and data length do not match. Affected record ID: <id>
[E018]	Data FLASH EEPROM file. RecordID <id> was used more than one time, error!
[E019]	Data FLASH EEPROM file. Section size validation error! Section size must be a size of (2048 * 2 pow(n)). Total size is: <size>
[E020]	Data FLASH EEPROM file. Section size is bigger than Data FLASH area size!
[E021]	Data FLASH EEPROM file. Missing opening bracket!
[E022]	Data FLASH EEPROM file. Missing closing bracket!
[E023]	Data FLASH EEPROM file. Wrong codeword or wrong data format found in file!
[E024]	Data FLASH EEPROM file. Missing carriage return and line feed!
[E025]	Data FLASH EEPROM file. Missing codeword or missing bracket!
[E026]	Data FLASH EEPROM file. Missing element within <AdministrativeSection> tag of XML file!
[E027]	Data FLASH EEPROM file. Missing element within <Record> tag of XML file!
[E028]	Data FLASH EEPROM file. Missing element within <DataFlash> tag of XML file!
[E029]	Data FLASH EEPROM file. Wrong tag <tag> found within XML file!
[E030]	Data FLASH EEPROM file. Error detected for tag <tag> within XML file!
[E031]	Data FLASH EEPROM file. Data size mismatch error! Data size is larger than section size. Data size: <size>, Section size: <size>
[E032]	Data FLASH EEPROM file. Offset + SectionSize*2 = <size> exceeds Data FLASH area end address!
[E033]	Data FLASH program (<start address> - <end address>) not in range of Data FLASH area!
[E034]	Code FLASH program (<start address> - <end address>) overlaps Data FLASH area!
[E035]	Data FLASH EEPROM emulation area (<start address> - <end address>) overlaps Data FLASH program (<start address> - <end address>)! Data FLASH EEPROM file could not be loaded!
[E036]	Data FLASH Program (<start address> - <end address>) overlaps Data FLASH EEPROM emulation area (<start address> - <end address>)! Data FLASH Program file could not be loaded!
[E037]	Overlap of Data FLASH Program (<start address> - <end address>) and Data FLASH EEPROM emulation area (<start address> - <end address>)!
[E038]	Data FLASH program file and Code FLASH program file must have the same file type, HEX or SREC!

Table 5: Error Messages 1/2

Number	*Message / Description*
[E039]	Data FLASH base address exceeds maximum size. Please specify a base address less then 0xF0000000!
[E040]	Wrong Data FLASH base address specified. Please insert Data FLASH base address as hexadecimal value! (for instance, Base Address: 0x001F8000)
[E041]	Data FLASH area does not start at 2KByte aligned address!
[E042]	Data FLASH Offset does not fit to 2KByte boundary!
[E043]	Wrong Data FLASH size specified. Supported sizes are 32KB and 64KB!
[E044]	Data FLASH program file does not start at 4Byte aligned address!
[E045]	Could not open template file template_eel_descriptor.txt! Create EEL_Descriptor.h according internal template file.
[E046]	Could not open template file template_fdl_descriptor.txt! Create FDL_Descriptor.h according internal template file.
[E047]	Could not open the parameter file or the specified parameter file is wrong.
[E048]	Illegal ID found in XML file!
[E049]	Data FLASH EEPROM file. Mismatch block size in XML file!
[E050]	Data FLASH EEPROM file. Wrong <EEL_CONFIG_BLOCK_START> tag in XML file!
[E051]	Data FLASH EEPROM file. Wrong <EEL_CONFIG_BLOCK_CNT> tag in XML file!
[E052]	Data FLASH EEPROM file. Wrong <EEL_CONFIG_DF_BASE_ADDRESS> tag in XML file!
[E053]	Data FLASH EEPROM file. Data size overflow in XML file!
[E054]	Data FLASH EEPROM file. Data length should not be larger than valid block space!
[E055]	Data FLASH EEPROM file. XML setting will cause overflow data flash size defined in device parameter file!
[E056]	Data FLASH EEPROM file. Wrong <EEL_CONFIG_BLOCK_CNT_REFRESH_THRESHOLD> tag in XML file!
[E057]	Data FLASH EEPROM file has invalid file format type! Only XML supported!

Table 6: Error Messages 2/2

[Memo]

For further information, please contact:

NEC Electronics Corporation
1753, Shimonumabe, Nakahara-ku,
Kawasaki, Kanagawa 211-8668,
Japan
Tel: 044-435-5111
http://www.necel.com/

[America]

NEC Electronics America, Inc.
2880 Scott Blvd.
Santa Clara, CA 95050-2554, U.S.A.
Tel: 408-588-6000
800-366-9782
http://www.am.necel.com/

[Europe]

NEC Electronics (Europe) GmbH
Arcadiastrasse 10
40472 Düsseldorf, Germany
Tel: 0211-65030
http://www.eu.necel.com/

Hanover Office
Podbielskistrasse 166 B
30177 Hannover
Tel: 0 511 33 40 2-0

Munich Office
Werner-Eckert-Strasse 9
81829 München
Tel: 0 89 92 10 03-0

Stuttgart Office
Industriestrasse 3
70565 Stuttgart
Tel: 0 711 99 01 0-0

United Kingdom Branch
Cygnus House, Sunrise Parkway
Linford Wood, Milton Keynes
MK14 6NP, U.K.
Tel: 01908-691-133

Succursale Française
9, rue Paul Dautier, B.P. 52
78142 Velizy-Villacoublay Cédex
France
Tel: 01-3067-5800

Sucursal en España
Juan Esplandiu, 15
28007 Madrid, Spain
Tel: 091-504-2787

Tyskland Filial
Täby Centrum
Entrance S (7th floor)
18322 Täby, Sweden
Tel: 08 638 72 00

Filiale Italiana
Via Fabio Filzi, 25/A
20124 Milano, Italy
Tel: 02-667541

Branch The Netherlands
Steijgerweg 6
5616 HS Eindhoven
The Netherlands
Tel: 040 265 40 10

[Asia & Oceania]

NEC Electronics (China) Co., Ltd
7th Floor, Quantum Plaza, No. 27 ZhiChunLu Haidian
District, Beijing 100083, P.R.China
Tel: 010-8235-1155
http://www.cn.necel.com/

Shanghai Branch
Room 2509-2510, Bank of China Tower,
200 Yincheng Road Central,
Pudong New Area, Shanghai, P.R.China P.C:200120
Tel:021-5888-5400
http://www.cn.necel.com/

Shenzhen Branch
Unit 01, 39/F, Excellence Times Square Building,
No. 4068 Yi Tian Road, Futian District, Shenzhen,
P.R.China P.C:518048
Tel:0755-8282-9800
http://www.cn.necel.com/

NEC Electronics Hong Kong Ltd.
Unit 1601-1613, 16/F., Tower 2, Grand Century Place,
193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: 2886-9318
http://www.hk.necel.com/

NEC Electronics Taiwan Ltd.
7F, No. 363 Fu Shing North Road
Taipei, Taiwan, R. O. C.
Tel: 02-8175-9600
http://www.tw.necel.com/

NEC Electronics Singapore Pte. Ltd.
238A Thomson Road,
#12-08 Novena Square,
Singapore 307684
Tel: 6253-8311
http://www.sg.necel.com/

NEC Electronics Korea Ltd.
11F., Samik Lavied'or Bldg., 720-2,
Yeoksam-Dong, Kangnam-Ku,
Seoul, 135-080, Korea
Tel: 02-558-3737
http://www.kr.necel.com/

G0706