

To our customers,

---

## Old Company Name in Catalogs and Other Documents

---

On April 1<sup>st</sup>, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1<sup>st</sup>, 2010  
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

## Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas 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 Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

"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.

"High Quality": 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; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.

8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

# Renesas FLASH Development Toolkit 3.07

(for Windows® 98SE/Me, Windows NT® 4.0,  
Windows® 2000 and Windows® XP)

## User's Manual

# Renesas FLASH Microcomputer Programming System

HS6400FDIW3S

## Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (<http://www.renesas.com>)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guarantees regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
  - (1) artificial life support devices or systems
  - (2) surgical implantations
  - (3) healthcare intervention (e.g., excision, administration of medication, etc.)
  - (4) any other purposes that pose a direct threat to human lifeRenesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.

# Table of Contents

i.	Cautions.....	i
ii.	Preface.....	ii
iii.	Abbreviations.....	iii
iv.	Document Conventions.....	iv
Chapter 1 Introduction.....		1
1.1	Key Features.....	1
1.2	New Features.....	1
Chapter 2 System Overview.....		3
2.1	User Interface.....	4
2.1.1	Menu bar.....	4
2.1.2	Toolbars.....	6
2.1.3	Status Bar.....	9
2.1.4	Pop-up menus.....	10
2.2	Help.....	10
2.3	Hot Keys.....	11
Chapter 3 Using FDT.....		13
3.1	Quickstart.....	13
3.2	Background Information.....	13
3.2.1	What are the User Area, User Boot Area and Data Areas?.....	13
3.2.2	What is the difference between Boot Mode and User Mode?.....	14
3.2.3	What is the difference between User Mode and User Program Mode?.....	14
3.2.4	What are “Protocol A”, “Protocol B”, “Protocol C” and “Protocol D” Kernels?.....	14
3.2.5	What are the fcf and fdm files, can I edit them?.....	15
3.2.6	Can I use FDT if I do not have Local Machine Administrator rights for my PC?.....	15
3.2.7	How do I recompile a kernel?.....	16
3.2.8	How do I use the supplied demonstration User Mode Kernels?.....	16
3.2.9	Can I modify the User Mode kernels to work for a different clock frequency?.....	17
3.3	Starting FDT.....	17
3.3.1	Starting FDT without Security or Simple Interface Mode enabled.....	17
3.3.2	Starting FDT in Basic Simple Interface Mode.....	17
3.3.3	Starting FDT by double clicking on a workspace file.....	18
3.3.4	Starting FDT by double clicking on a data file.....	18
3.3.5	Starting FDT by double clicking on a w4f script configuration file.....	18
3.3.6	Starting FDT with Simple Interface Mode enabled.....	18
3.3.7	Starting FDT with Access Rights password security enabled.....	18
3.3.8	Starting FDT by double clicking on a .fpf packaged project file.....	18
3.3.9	Starting FDT from the command line.....	18
3.3.9.1.	Opening data files from the command line.....	18
3.3.9.2.	Opening workspace files from the command line.....	18

3.3.9.3.	Opening data files from the command line with Access Rights password security enabled.....	19
3.3.9.4.	Opening w4f Script Configuration files from the command line.....	19
3.3.9.5.	Opening w4f Script Configuration files from the command line with Access Rights password security enabled.....	19
3.3.9.6.	Starting FDT Basic Simple Interface Mode from the command line.....	19
3.4	Creating a New Workspace and Project .....	20
3.4.1	New Project Workspace .....	20
3.4.2	Choose Device and Kernel (not Generic Boot* ).....	21
3.4.3	Communications Port .....	22
3.4.4	Device Settings.....	23
3.4.5	Connection Type .....	24
3.4.6	Programming Options .....	25
3.4.7	FDM/E8Direct Pin Settings.....	25
3.4.8	Reset Pin Settings.....	27
3.5	Configuring a Project for a Generic Boot Device.....	29
3.5.1	New Project Workspace .....	29
3.5.2	Generic Boot - Choose Device and Kernel.....	30
3.5.3	Generic Boot - Communications Port.....	31
3.5.4	Generic Boot - FDM Pin Settings [FDM connection only] .....	31
3.5.5	Generic Boot - E8Direct Pin Settings [E8Direct connection only] .....	32
3.5.6	Generic Boot - Confirmation .....	34
3.5.7	Generic Boot – Select Device.....	34
3.5.8	Generic Boot – Select Clock Mode .....	35
3.5.9	Generic Boot – Setup Complete .....	36
3.5.10	Generic Boot – Device Settings.....	36
3.5.11	Generic Boot - Connection Type.....	36
3.5.12	Generic Boot – Programming Options .....	36
3.5.13	Generic Boot – Reset Pin Settings [FDM connection only].....	36
3.5.14	Generic Boot – Reinterrogate on connect option .....	36
3.5.15	Prior Generic Device .....	36
3.6	Connecting to the device .....	37
3.6.1	E8 USB Connection .....	37
3.6.2	ID Check Dialog.....	37
3.6.3	Locking .....	38
3.7	Programming the Data to the FLASH ROM.....	38
3.7.1	Programming an individual file to the User Area (normal flash area).....	38
3.7.2	Programming multiple files to the User Area (normal flash area) .....	38
3.7.3	Programming an individual file to the User Boot Area.....	38
3.7.4	Programming multiple files to the User Boot Area.....	39
3.7.5	Programming some files to the User Area and other files to the User Boot Area.....	39
3.8	Erasing Data from the FLASH ROM .....	40
3.9	Uploading the FLASH ROM Data .....	41
3.10	Performing a Blank Check.....	42
3.11	Disconnecting from the device .....	42
3.12	Saving a Workspace .....	42
3.13	Closing a Workspace.....	42
3.14	Exiting FDT.....	42

Chapter 4	Wait For Script.....	43
4.1	Wait For Script Overview.....	43
4.2	Wait For Script – New Configuration.....	43
4.2.1	Script File Page .....	43
4.2.1.1.	Description.....	44
4.2.1.2.	Script Path .....	44
4.2.1.3.	Timings .....	44
4.2.1.4.	Save.....	44
4.2.2	Log File Page.....	44
4.2.2.1.	Log File Directory .....	45
4.2.2.2.	Log File Name.....	45
4.2.2.3.	Save.....	45
4.2.3	Status File Page .....	45
4.2.3.1.	Status File Directory.....	45
4.2.3.2.	Status File Name .....	45
4.2.3.3.	Save.....	46
4.2.4	Error Handling Page.....	46
4.2.4.1.	On Script Fail .....	46
4.2.4.2.	Save.....	47
4.3	Wait For Script – Edit Configuration.....	48
4.4	Wait For Script Run Configuration.....	48
4.5	Wait For Script – Script Commands.....	48
4.5.1	Workspace.....	49
4.5.2	Connect .....	49
4.5.3	Download .....	50
4.5.4	Erase.....	50
4.5.5	BlankCheck.....	51
4.5.6	Upload.....	51
4.5.7	Checksum.....	51
4.5.8	Disconnect.....	52
4.5.9	RebuildImage .....	52
4.5.10	# Comment .....	52
4.5.11	Alert .....	53
4.5.12	Pause .....	53
4.5.13	Delete .....	53
4.5.14	Move \ Copy.....	54
4.5.15	Abort .....	54
4.6	Wait For Script – Example Script File.....	55
4.7	Wait For Script – Status File Format .....	56
Chapter 5	Access Rights.....	57
5.1	Access RightsOverview.....	57
5.2	Enabling Access Rights.....	57
5.3	Access Rights Administration .....	57
5.3.1	Disable All.....	58
5.3.2	Modify User .....	58
5.3.3	Add User .....	58
5.3.4	Delete User.....	59
5.3.5	Reset Pwd.....	59

5.3.6	Close.....	59
<b>Chapter 6 Configuring the User Interface ..... 61</b>		
6.1	Arranging Windows .....	61
6.1.1	Minimising windows .....	61
6.1.2	Tiling windows.....	61
6.1.3	Cascading windows .....	61
6.2	Locating Currently Open Windows .....	61
6.2.1	Locating a specific window .....	61
6.3	Enabling/disabling the Toolbar.....	62
6.4	Enabling/disabling the Workspace Window .....	62
6.5	Enabling/disabling the Output Window.....	62
6.6	Customising the Toolbar .....	63
6.6.1	Toolbars Tab .....	63
<b>Chapter 7 Menus ..... 65</b>		
7.1	File Menu .....	65
7.1.1	Close.....	65
7.1.2	New Workspace.....	65
7.1.3	Open Workspace.....	65
7.1.4	Save Workspace .....	65
7.1.5	Close Workspace.....	65
7.1.6	Open an S-Record.....	66
7.1.7	Save.....	66
7.1.8	Save As.....	66
7.1.9	Exit .....	66
7.2	Edit Menu .....	66
7.2.1	Undo.....	66
7.2.2	Redo .....	66
7.2.3	Cut.....	66
7.2.4	Copy .....	66
7.2.5	Paste .....	66
7.2.6	Select All.....	66
7.2.7	Find.....	66
7.2.8	Find In Files.....	67
7.2.9	Replace.....	67
7.3	View Menu .....	67
7.3.1	Workspace .....	67
7.3.2	Output.....	67
7.4	Project Menu .....	67
7.4.1	Set Current Project .....	67
7.4.2	Insert Project.....	67
7.4.3	Add Files.....	67
7.4.4	Remove Files.....	67
7.4.5	File Extensions.....	67
7.4.5.1	File Extensions Add.....	68
7.4.5.2	File Extensions Modify.....	69
7.4.6	Rebuild Image .....	69
7.4.7	Download Image .....	69



7.4.8	Field Programming.....	69
7.5	Tools Menu .....	70
7.5.1	Administration.....	70
7.5.2	Customise.....	70
7.5.3	Options.....	70
7.5.4	Format Views.....	70
7.5.5	Simple Interface.....	70
7.5.6	Wait For Script.....	70
7.5.7	Access Rights.....	70
7.5.7.1.	Access Rights – Login.....	70
7.5.7.2.	Access Rights – Change Password.....	70
7.5.7.3.	Access Rights – Administration.....	70
7.6	Window Menu .....	71
7.6.1	Cascade .....	71
7.6.2	Tile Horizontally .....	71
7.6.3	Tile Vertically.....	71
7.6.4	Arrange Icons .....	71
7.6.5	Close All.....	71
7.6.6	'file name' .....	71
7.7	Device Menu .....	71
7.7.1	Connect to Device .....	71
7.7.2	Disconnect.....	71
7.7.3	Erase FLASH blocks .....	71
7.7.4	Blank check.....	72
7.7.5	Upload Image .....	72
7.7.6	Download Active File.....	72
7.7.7	FLASH Checksum.....	72
7.7.8	Go From Address.....	72
7.7.9	Block Locking.....	72
7.7.10	Cancel Operation.....	72
7.7.11	Configure Flash Project.....	72
7.8	Help Menu.....	72
7.8.1	Help Topics .....	72
7.8.2	Technical Support.....	72
7.8.3	About.....	73
<b>Chapter 8 Windows .....</b>		<b>75</b>
8.1	Workspace window .....	75
8.1.1	Workspace.....	75
8.1.1.1.	Insert Project.....	75
8.1.1.2.	Allow Docking .....	76
8.1.1.3.	Hide.....	76
8.1.1.4.	Properties.....	76
8.1.2	Project .....	77
8.1.2.1.	Add Files.....	77
8.1.2.2.	Remove Files.....	77
8.1.2.3.	Add Folder.....	77
8.1.2.4.	Allow Docking .....	77
8.1.2.5.	Hide.....	77

8.1.2.6.	Properties.....	77
8.1.2.7.	Set as current Project.....	78
8.1.2.8.	Remove Project.....	78
8.1.2.9.	Unload Project.....	78
8.1.2.10.	Add Files From Kernel Dir.....	78
8.1.3	Device Image - Folder.....	78
8.1.3.1.	Remove Folder.....	78
8.1.3.2.	Rename Folder.....	78
8.2	Device File.....	79
8.2.1	Open 'filename'.....	79
8.2.2	Add Files.....	79
8.2.3	Remove Files.....	79
8.2.4	Allow Docking.....	79
8.2.5	Hide.....	79
8.2.6	Properties.....	80
8.2.7	Display Block usage.....	80
8.2.8	Exclude 'filename'.....	80
8.2.9	User Boot Area.....	80
8.2.10	Download File to [User Area] /[User Boot Area].....	80
8.2.11	File Checksum.....	80
8.2.12	Compare File->Device Checksum.....	80
8.2.13	Compare File->Device (Complete Device).....	81
8.2.14	Compare File->Device (File Data Only).....	81
8.3	Workspace Properties.....	81
8.4	Project Properties.....	82
8.5	Configure Flash Project (Flash Properties).....	83
8.5.1	Kernel Tab.....	83
8.5.2	Communications Tab.....	84
8.5.3	Device Tab.....	85
8.5.4	Programmer Tab.....	86
8.5.5	Modules Tab.....	88
8.6	S-Record Properties Window.....	89
8.6.1	Block Usage Tab.....	89
8.6.2	Cursor Value.....	89
8.6.3	Selection Value.....	90
8.7	Output Window.....	90
8.7.1	Copy.....	90
8.7.2	Clear Window.....	90
8.7.3	Save.....	90
8.7.4	Allow Docking.....	91
8.7.5	Hide.....	91
8.8	Hex Editor Window.....	91
8.8.1	Cut.....	92
8.8.2	Copy.....	92
8.8.3	Paste.....	92
8.8.4	Undo.....	92
8.8.5	Redo.....	92
8.8.6	Display Unit.....	93
8.8.7	Align to 8 Bytes.....	93

8.8.8	Toggle ASCII Column.....	93
8.8.9	Create Selection.....	93
8.8.10	Fill.....	93
8.8.11	Find.....	93
8.8.12	Replace.....	93
8.8.13	Properties.....	93
8.8.14	Add to Active Project.....	93
8.8.15	Data EntryNote.....	94
8.8.16	Goto Address.....	94
8.9	Customise - Toolbars.....	94
8.9.1	Show Tooltips.....	94
8.9.2	New.....	94
8.9.3	Reset.....	95
8.9.4	Toolbar name.....	95
8.9.5	OK.....	95
8.9.6	Delete.....	95
8.10	Customise – Menu.....	96
<b>Chapter 9 Simple Interface Mode.....</b>		<b>99</b>
9.1	Options Menu.....	100
9.1.1	Login.....	100
9.1.2	Batch Erase Mode.....	100
9.1.3	AutoDisconnect.....	100
9.1.4	Readback Verify.....	100
9.1.5	Request Checksum.....	100
9.1.6	Erase Device Before Program.....	100
9.1.7	Security Protection.....	100
9.1.8	Field Programming->Run Published File.....	100
9.1.9	About.....	100
9.2	Dialog Controls.....	101
9.2.1	Back to Project.....	101
9.2.2	Exit.....	101
9.2.3	Download Project image / Download File radio buttons.....	101
9.2.4	Project Image / Download File Area.....	101
9.2.5	Program Flash \ Erase.....	101
9.2.6	Disconnect.....	101
9.2.7	File Selection.....	101
<b>Chapter 10 Basic Simple Interface Mode.....</b>		<b>103</b>
10.1	Options Menu.....	104
10.1.1	Login.....	104
10.1.2	Batch Erase Mode.....	104
10.1.3	AutoDisconnect.....	104
10.1.4	Readback Verify.....	104
10.1.5	Request Checksum.....	104
10.1.6	Erase Device Before Program.....	104
10.1.7	Security Protection.....	104
10.1.8	New Settings.....	104
10.1.9	About.....	104

10.2	Dialog Controls .....	105
10.2.1	Exit .....	105
10.2.2	Download File radio button .....	105
10.2.3	Download File Area .....	105
10.2.4	Program Flash \ Erase.....	105
10.2.5	Disconnect.....	105
10.2.6	File Selection.....	105
Chapter 11 Field Programming.....		107
11.1	Options Menu .....	108
11.1.1	Login.....	108
11.1.2	Batch Erase Mode.....	108
11.1.3	AutoDisconnect .....	108
11.1.4	Readback Verify.....	108
11.1.5	Request Checksum .....	108
11.1.6	Erase Device Before Program .....	108
11.1.7	Security Protection .....	108
11.1.8	Field Programming->Run Published File .....	108
11.1.9	About.....	108
11.2	Dialog Controls .....	109
11.2.1	Back to Project .....	109
11.2.2	Exit.....	109
11.2.3	Download Project image radio button .....	109
11.2.4	Program Flash \ Erase.....	109
11.2.5	Disconnect.....	109
Chapter 12 E8 Update .....		111
Chapter 13 E8Direct.....		113
13.1	Firmware and Device ID .....	113
13.2	Hardware Differences.....	115
Chapter 14 QzROM Programming .....		117
14.1	Reserved Areas.....	117
14.2	Security Protection .....	118
14.2.1	Toolbar .....	119
14.2.2	Project .....	119
14.2.3	Simple Interface \ Basic Mode .....	120
Chapter 15 Locking.....		121
15.1	Locking Overview .....	121
15.2	Connection .....	122
15.3	Locking Dialog.....	123
15.4	Disconnection.....	125
Chapter 16 Upgrading to FDT3.07 .....		127

## i. Cautions

Renesas neither warrants nor grants licenses of any rights of Renesas' or any third party's patent, copyright, trademark, or other intellectual property rights for information contained in this document. Renesas bears no responsibility for problems that may arise with third party's rights, including intellectual property rights, in connection with use of the information contained in this document.

Products and product specifications may be subject to change without notice. Confirm that you have received the latest product standards or specifications before final design, purchase or use.

Renesas makes every attempt to ensure that its products are of high quality and reliability. However, contact Renesas' sales office before using the product in an application that demands especially high quality and reliability or where its failure or malfunction may directly threaten human life or cause risk of bodily injury, such as aerospace, aeronautics, nuclear power, combustion control, transportation, traffic, safety equipment or medical equipment for life support.

Design your application so that the product is used within the ranges guaranteed by Renesas particularly for maximum rating, operating supply voltage range, heat radiation characteristics, installation conditions and other characteristics. Renesas bears no responsibility for failure or damage when used beyond the guaranteed ranges. Even within the guaranteed ranges, consider normally foreseeable failure rates or failure modes in semiconductor devices and employ systemic measures such as fail-safes, so that the equipment incorporating Renesas product does not cause bodily injury, fire or other consequential damage due to operation of the Renesas product.

This product is not designed to be radiation resistant.

No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without written approval from Renesas.

Contact Renesas' sales office for any questions regarding this document or Renesas semiconductor products.

## ii. Preface

### **About this guide**

This guide explains the use of the Renesas Flash Development Toolkit(hereafter referred to as FDT).

Chapter 1 , Introduction, provides a brief explanation to the tool and lists its key features.

Chapter 2 , System Overview, describes how the different facilities make up the FDT Graphical User Interface.

Chapter 3 , Using FDT, describes how FDT is activated and the FLASH ROM is written.

Chapter 4 , Wait For Script

Chapter 5 , Access Rights describes configuration of User Names and Passwords

Chapter 6 , Configuring the User Interface, provides a way to configure the FDT Graphical User Interface.

Chapter 7 , Menus, and Chapter 8 , Windows, give reference information about the operations and facilities available through these respective areas.

Chapter 9 , Simple Interface Mode, describes use of FDT with a cut-down Graphical User Interface.

Chapter 10 , Basic Simple Interface Mode, describes use of FDT with a cut-down Graphical User Interface without the need to setup a workspace and project.

Chapter 11 , Field Programming, describes use the Field Programming feature for publishing projects for other FDT users, and use of published projects.

Chapter 12 , E8 Update, describes how FDT updates the E8 firmware.

Chapter 13 , E8Direct describes how an E8 can be used as an E8Direct

Chapter 14 , QzROM Programming describes the programming of the new style of Renesas Flash

Chapter 15, Locking describes FDT operation with those devices that support erase block locking.

Chapter 16 describes notes about upgrading to FDT3.07.

### **Assumptions**

It is assumed that the reader is experienced in using Microsoft® Windows® applications on PC-compatible computers.

### iii. Abbreviations

Target / Device	Refers to the programmable microcontroller or microcomputer that is connected to the PC ready for programming.
DLL	Dynamic Linked Library
FDT	Flash Development Toolkit
F-ZTAT <sup>®</sup>	Flexible Zero Turn-Around Time
QzROM	One time programmable memory
HEW	Renesas High-performance Embedded Workshop
PC	Personal Computer
USB	Universal Serial Bus 1.1 or above
FDM	Flash Development Module (USB interface board)
UPB	Universal Programming Board – Legacy HMSE interface board used with Protocol A (Legacy) kernels.
HMSE	Hitachi Micro Systems Europe Ltd – The former name of part of the Renesas Technology Europe Engineering Division.
FCF	Flash Configuration File – Used to store information about the device and flash characteristics
DDI	Device Data Image – a hybrid S-Record file format used for storing data for a device with multiple flash areas.
Kernel	Code which takes control of the device and allows reprogramming functions. This code is often downloaded to the device.
Micro Kernel	The part of the kernel which gets executed first and allows initial configuration and downloading of the main kernel.
Main Kernel	The part of the kernel which gets downloaded by the micro kernel and handles downloading of additional modules (if required) to perform the read / write / blank check / erase functions etc.
E8 / E8USB	This is a USB interface board. It supports programming over the single wire interface and clock synchronous programming using the four wire interface.
FoUSB	This is a USB interface board developed by RTA. It is used for Flash Programming and in-circuit debugging. This is currently not supported by FDT in the Asia region.
Field Programming - Publishing	The ability to package all kernel and data files into a single file for simplified sending to another FDT user.
E8Direct	A firmware update to the E8 (using the existing E8 hardware), designed to be as similar as possible to the FDM

## iv. Document Conventions

This manual uses the following typographic conventions:

CONVENTION	MEANING
<b>[Menu-&gt;Menu Option]</b>	Bold text with ‘->’ is used to indicate menu options (for example, [File->Save As...]).
‘dialog name’	The ‘ ’ is used to indicate the name of a dialog box or menu.
<b>‘filename.ext’</b>	Bold Courier Font is used to indicate filenames.
“ <u>enter this string</u> ”	Used to indicate text that must be entered (excluding the “ ” quotes).
Key+Key	Used to indicate required key presses. For example. Ctrl+N means press the Ctrl key and then, whilst holding the Ctrl key down, press the N key.
↪ (The “how to” symbol)	When this symbol is used, it is always located in the left-hand margin. It indicates that the text to its immediate right is describing “how to” do something.

Table 1: Typographic Conventions

Windows® is a registered trademark of Microsoft Corporation.

F-ZTAT is a trademark of Renesas, Ltd.



# Chapter 1 Introduction

The RenesasFlash Development Toolkit (FDT) is an on-board FLASH programming tool for Renesas F-ZTAT microcomputers that provides a high-performance and user-friendly Graphical User Interface (GUI).

Embedded software development projects created using the Renesas High-performance Embedded Workshop (HEW) may be programmed into Renesas F-ZTAT devices using FDT.

FDT may also be used as a general purpose S-Record or Hex editor.

## 1.1 Key Features

- Standard window operation based on the 32-bit Windows® GUI.
- Various online help functions.
- Selectable messaging levels.
- Simple programming environment; optionally controls an adapter board.
- Serial communication.
- USB communications supported via USB Interface Boards.
- USB communications directly to selected target devices.
- Wait for Script File facility
- Multiple flash areas in a single image (DDI)
- QzROM programming

## 1.2 New Features

FDT3.07 has the following new features:-

- Support for Lock Bit commands
- Support for R8C/2x Blank Check embedded function
- BPS (synchronous) selection support for QZ-ROM
- Display Clock Frequency / CKM / CKP in output Window
- SH2A Data Mat Support



## Chapter 2 System Overview

FDT provides a method of programming, erasing and reading from Renesas Flash devices. Workspaces and Projects can be used in order to save the settings between sessions, for easy switching between different settings, and to allow experienced users to configure the settings for less experienced operators. FDT is designed to provide a common look and feel, independent of the actual device to be programmed.

FDT employs a hierarchical structure so that work can be organised in a logical manner. The top level of the structure is the workspace.

To be useful, the workspace must contain at least one project. In order to create a project, a workspace must be created first.

Each project specifies its own target device configuration (specified when creating the project) and set of target files (S-Record / Hex / Binary) that can be used to program the device.

The project settings for the target device connection need only be set once, as they can be stored between sessions by saving the Project.

A single project within the workspace is active at any point in time. The active project is the context to which all 'Device' Menu, 'Project' Menu and 'Project' Toolbar commands will be directed.

When a project has been created, target files can be added to it. These files may:

- Be used to program the device.
- Be used to build a Device Image.
- Be opened in the hex editor.
- Be marked as being destined for the User Boot Area (for devices which support this).
- Have their checksum calculated
- Have their data block usage displayed
- Be compared against the data already programmed to the device

When using a project it is possible to take advantage of the following FDT features:

- Advanced messaging levels.
- Device Image builder.
- Uploading data from the target device.
- Viewing uploaded data in big or little endian, or floating point representation.
- Simple Interface Mode – for simplifying the user interface once the project is configured.
- Automatic masking of non-Flash areas in files upon opening project open
- Automatic masking of reserved sections within Flash areas

## 2.1 User Interface

The FDT Graphical User Interface is a Windows® application that presents a work environment, which allows the user to program FLASH memory.

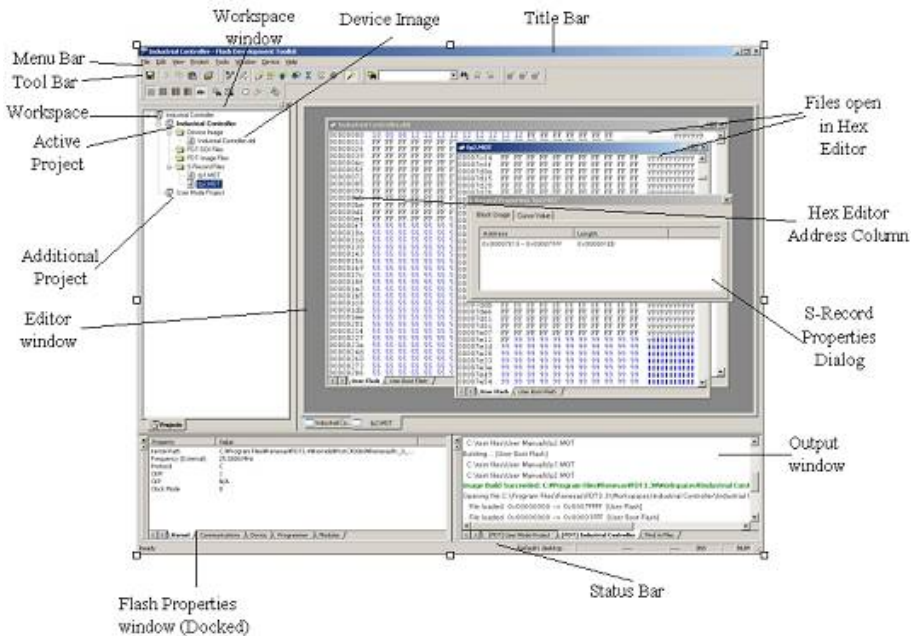
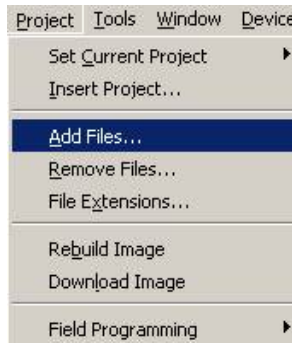


Figure 2-1 FDT Graphical User Interface

### 2.1.1 Menu bar

Commands are grouped into related areas on the Menu bar as indicated by the menu titles. Using the mouse the user can select a command operation, invoke a dialog box or a window to interact with the system. Clicking the left mouse button on a menu title will pull down that menu, from which a selection can be made.

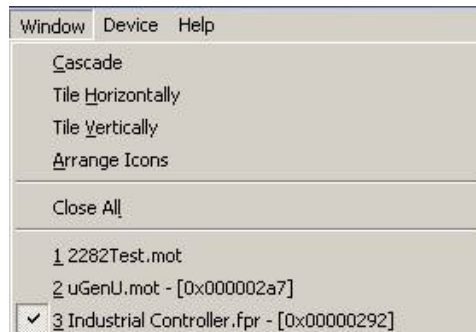
If a menu item does not perform an action directly, but instead displays a dialog box or window for further user interaction, then its name is appended with an ellipsis (three dots, ...).



**Figure 2-2 Menu Ellipsis**

If a menu item can also be invoked by pressing a hot key (a combination of keys), then the hot key is displayed to the right of the item.

If a menu item toggles a feature ON or OFF then a check mark (✓) will be displayed next to its text when it is ON:



**Figure 2-3 Checked Menu Items**

If a menu item has the symbol (▶) next to it then a cascading or hierarchical menu is available. Clicking on the menu item will reveal the cascading menu:



**Figure 2-4 Cascading Menus**

Menus can also be selected directly from the keyboard by pressing the ALT key followed by the corresponding key of the underlined letter or number for the menu item that the user wants to select, e.g. press ALT+F, S in sequence to Save the active file (**File->Save**).

## 2.1.2 Toolbars

FDT has several toolbars located below the Menu bar. This provides quick access to FDT features by clicking the respective button with the mouse.



**Figure 2-5 FDT Toolbars**

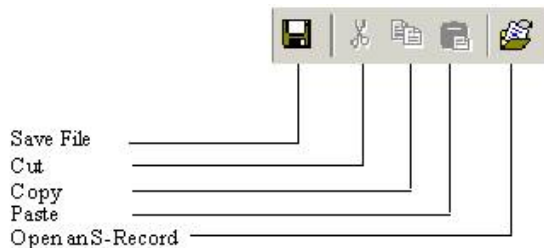
The buttons on each toolbar are arranged in associated groups.

To find out the function of the button, move the mouse over the button and a hint will be displayed next to the button and in the status bar.

The toolbar buttons can be customised to provide a button for the majority of the features available in FDT and can be arranged in an order that the user finds convenient.

For more details about changing the arrangement of the toolbar buttons and a detailed description of each button's function, see Chapter 6 .

### Edit toolbar



**Figure 2-6 Edit Toolbar**

*Save File* – saves the active file.

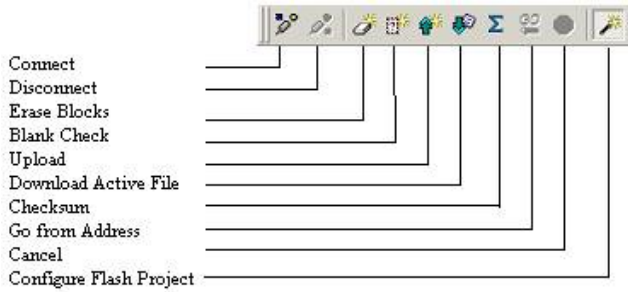
*Cut* – cuts data to the clipboard.

*Copy* – copies data to the clipboard.

*Paste* – pastes data from the clipboard.

*Open an S-Record* – Opens an S-Record or data file.

## FDT toolbar



**Figure 2-7 FDT Toolbar**

*Connect* - connects the device to the interface.

*Disconnect* - disconnects the device from the interface.

*Erase Blocks* - launches the 'Erase Block' dialog box to erase all or individual blocks of the device FLASH memory.

*Blank Check* - checks whether or not the FLASH section of the target device is blank.

*Upload* - launches the 'Upload Image' dialog box to allow data to be obtained from the target device.

*Download Active File* - downloads the file which is active in the hex editor.

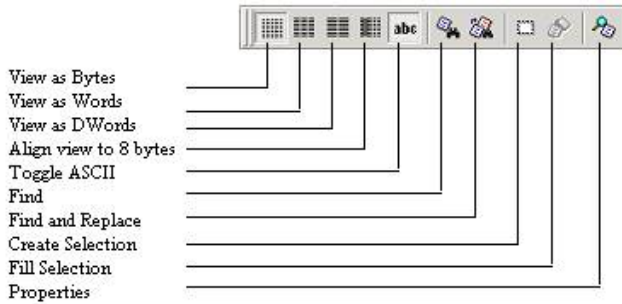
*Checksum* - returns a checksum of the data in the FLASH.

*Go from Address* - launches a dialog to select the address to execute from.

*Cancel* - Cancels the current FLASH operation.

*Configure Flash Project* - launches the 'Project Properties' dialog box (or the Project Wizard if the project is not yet configured).

## S Record toolbar



**Figure 2-8 S Record Toolbar**

*View as Bytes* - view the file data as 8 bit bytes.

*View as Words* - view the file data as 16 bit words.

*View as DWords* - view the file data as 32 bit double words.

*Align view to 8 bytes* - data is displayed on each line as 8 bytes. The number of bytes that can be accommodated on each line is dependent upon the size of the window.

*Toggle ASCII* - turns ASCII data ON or OFF

*Find* - launches the 'Find' dialog box.

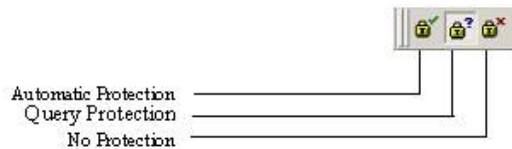
*Find and Replace* - launches the 'Replace' dialog box.

*Create Selection* - selects a block of the specified size in the active file.

*Fill selection* - launches the 'Fill' dialog box.

*Properties* - launches a dialog with information about the active file.

## QzROM toolbar



**Figure 2-9 QzROM Toolbar**



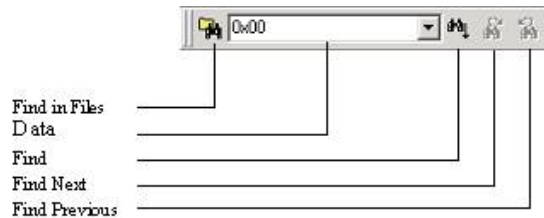
*Automatic Protection* – Set QzROM security protection automatically on disconnect (if more than one option is available a dialog will appear to select which type to automatically apply)

*Query Protection* – Ask the user whether to set QzROM security protection on disconnect.

*No Protection* – Never set QzROM security protection on disconnect.

**NB – setting the protection may stop any further connections to that particular target device**

## Search toolbar



**Figure 2-10 Search Toolbar**

*Find in Files* – finds selected data in selected files.

*Data* – is used for the searches.

*Find* – locates data in active file.

*Find Next* – locates next occurrence of search criteria.

*Find Previous* - locates previous occurrence of search criteria.

### 2.1.3 Status Bar

The Status Bar is located at the bottom of the FDT application window. It provides the user with information about what the application is doing.

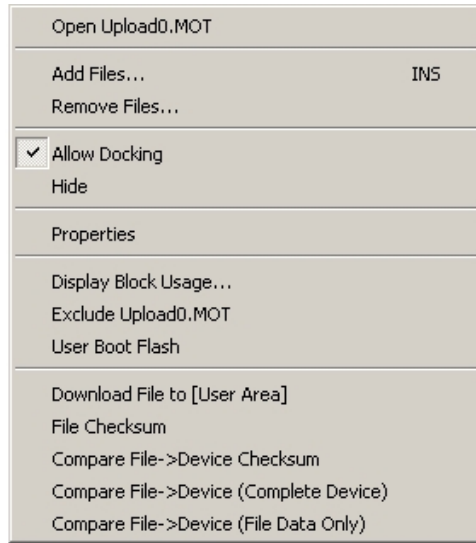
The left section of the Status Bar describes the action of the Toolbar items as the mouse arrow is positioned over a button, as well as displaying operation status information.

When an action is being performed, a progress bar is displayed to show the progress of the current operation.

The last section of the bar indicates the state of the toggling keyboard keys such as the Caps Lock, Insert and Number Lock keys.

### 2.1.4 Pop-up menus

Many windows have local pop-up menus in order to make commonly used features easier to access. These menus are invoked by clicking the right mouse button in the window (or pressing **SHIFT+F10**) and then selecting the required menu option. An example pop-up menu is:



**Figure 2-11 Pop-up Menu**

The contents and operation of specific pop-up menus are explained in detail in the description of each FDT window, see the Chapter 8 ,Windows.

## 2.2 Help

FDT has a standard Windows® help system. This provides on-line information for using the system.

Help can be invoked by pressing the **F1** key or via the Help menu.

## 2.3 Hot Keys

There are various hot keys for frequently used operations. These are as follows by category:-

<u>Description</u>	<u>Hot Key</u>
<i>Help</i>	
Help	F1
<i>File Commands</i>	
<u>C</u> lose File	Ctrl+F4
Open S- <u>R</u> ecord	Ctrl+R
Save <u>A</u> ll	Ctrl+Shift+H
<u>S</u> ave	Ctrl+S
<u>E</u> xit	Alt+F4
<i>Edit Commands</i>	
<u>U</u> ndo	Ctrl+Z
<u>R</u> edo	Ctrl+Y
<u>C</u> ut	Ctrl+X
<u>C</u> opy	Ctrl+C
<u>P</u> aste	Ctrl+V
Select <u>A</u> ll	Ctrl+A
<u>F</u> ind	Ctrl+F
Find <u>I</u> n Files	F4
<u>R</u> eplace	Ctrl+H
<i>Tools Commands</i>	
Login	Ctrl+Shift+U
<i>Simple Interface Commands</i>	
Login	Ctrl+Shift+U
<i>View</i>	
<u>W</u> orkspace	Alt+K
<u>O</u> utput	Alt+U

<u>Description</u>	<u>Hot Key</u>
<i>Window</i>	
View as <u>A</u> SCII	Alt+A
View as <u>B</u> ytes	Alt+1
View as <u>W</u> ord	Alt+2
View as <u>D</u> word	Alt+4
Align to <u>8</u> Bytes	Alt+8
 <i>Device</i>	
<u>C</u> onnect	Ctrl+Alt+C
<u>E</u> rase Flash Blocks	Ctrl+Alt+E
<u>B</u> lank Check	Ctrl+Alt+B
<u>U</u> pload Image	Ctrl+Alt+U
<u>D</u> ownload Active File	Ctrl+Alt+P
<u>F</u> lash Checksum	Ctrl+Alt+S
<u>G</u> o From Address	Ctrl+Alt+G
<u>B</u> lock Locking...	Ctrl+Alt+L
<u>C</u> ancel Operation	Ctrl+Break
<u>C</u> onfigure Flash <u>P</u> roject	Alt+Shift+R
 <i>Data File</i>	
Suppress request to save changes to a data file when closing it (do not save changes)	Hold Shift key when closing data file

# Chapter 3 Using FDT

## 3.1 Quickstart

The stages required in order to program a device are as follows:

- 1) Start FDT by selecting 'Start' -> 'Programs' -> 'Renesas' -> 'Flash Development Toolkit3.07', then the 'Flash Development Toolkit3.07' shortcut [see section 3.3]
- 2) Create a Workspace and Project [see section 3.4.1]
- 3) Configure the Project using the Flash Project Wizard (select device, kernel, port, mode etc) [see sections 3.4.2 to 3.4.8 or section 3.5 for Generic Boot]
- 4) Add the file or files that you want to download to the project [see sections 8.1.2.1/8.2.2/3.7]. You can now download individual files by right clicking on them in the Workspace Window, or merge multiple files into a device image (using the Project-> Rebuild Image menu item) and download that (using the Project-> Download Image menu item).

Alternatively, to start in FDT Basic Simple Interface Mode:

- 1) Select 'Start' -> 'Programs' -> 'Renesas' -> 'Flash Development Toolkit3.07', then the 'Flash Development Toolkit3.07 Basic' shortcut [see section 3.3]
- 2) Configure the settings using the Wizard (select device, kernel, port, mode etc) [see sections 3.4.2 to 3.4.8 or section 3.5 for Generic Boot]
- 3) Select the flash area that you want to program (User Area or User Boot Area).
- 4) Select the file to program for each enabled area
- 5) Select the Program Flash Button.
- 6) Disconnect when programming is completed

## 3.2 Background Information

### 3.2.1 What are the User Area, User Boot Area and Data Areas?

The normal area of (internal) flash for Renesas devices, for user programs is known as the **User Area**. This is where the code is stored which gets executed when the device is reset into User Mode. The **User area** can normally be programmed from a Boot Mode or User Mode connection.

On some devices there is an additional area of flash known as the **User Boot Area**. This is where the code is stored which gets executed when the device is reset into **User Boot Mode**. The User Boot Area tends to be much smaller than the User Area and is typically used for storing an alternative boot stub, for setting up communication via some other communications medium / protocol. Normally, the User Area and the User Boot Area both start at the same address – this is possible because only one is ever mapped into the memory map at one time. It is only possible to program the User Boot Area from Boot Mode - not from User Mode.

On some devices there is an additional area of flash known as the **Data Area**. This area normally resides at the other end of the memory map to the **User Area**, and unlike the case of the **User Boot Area**, the **Data Area** is not contiguous with the **User Area** and always exists in the memory map.

### 3.2.2 What is the difference between Boot Mode and User Mode?

The Renesas Flash devices have different modes of operation which are controlled by the state of the mode pins when the device is brought out of reset. Most devices have an asynchronous serial Boot Mode and a User/User Program Mode.

Generally, (asynchronous serial) **Boot Mode** connections, start with an autobaud – sending synchronisation data packets to setup initial communications at a given baud rate. Additionally, for most devices, the entire flash will be erased as part of the connection process, thus preventing any Intellectual Property held on the device from being read from Boot Mode.

For a **User Mode** connection, the device must already be programmed with some code which talks to FDT and allows programming. Most kernels are supplied with a User Mode micro kernel and test program (\*usually called **"uGenU.mot"** and **"xxxxtest.mot"**). These files provide a demonstration of User Mode programming and must be programmed into the device and be executing before FDT tries to connect in User Mode. User Mode does not normally force an erase of the flash during the connection process and does not autobaud to setup initial communications. The demonstration User Mode test programs have the required Bit Rate Register value compiled into them for setting up initial communications at 9600 baud; hence User Mode kernels need to be rebuilt for different clock frequencies.

- \* Note that where the files are to be used as part of an embedded monitor, the User Mode kernel files may be also output as binary files (.cde) without the reset vector populated. Some kernels use only a single file for the User Mode test program (e.g. Protocol C kernels).

### 3.2.3 What is the difference between User Mode and User Program Mode?

Some devices use an additional mode pin to protect the flash from being written or erased by mistake; this pin is normally known as the "Flash Write Enable" pin (FWE) if it has to be logic high to allow programming, or the "Flash Write Protect" pin (FWP) if it has to be logic low to allow programming. For these devices, this pin distinguishes "User Mode" from "User Program Mode". An FDT User Mode connection expects the device to be in User Program Mode at the point of programming and erasing, and if using the FDM, can reset the device into User Mode to execute the programmed code on disconnection.

### 3.2.4 What are "Protocol A", "Protocol B", "Protocol C" and "Protocol D" Kernels?

**"Protocol A"** - For some 0.8 $\mu$ m, 0.6 $\mu$ m and 0.35 $\mu$ m flash devices using FDT 1.5 style kernels.

These are legacy kernels that are not supported and not included in the FDT installer. Protocol A kernels are not available in the Japanese versions of FDT. Protocol A kernels support use of the UPB but not the FDM. The kernel needs to be rebuilt for different crystal frequencies in Boot and User Modes, and for changing the PMODE pin if using the UPB. Please refer to the documentation supplied with the kernels for additional information.

**"Protocol B"** – For 0.35 $\mu$ m devices. These support the FDM but not the UPB.

**Boot Mode** - The kernel (made up of the Boot Mode Micro Kernel, Boot Mode Main Kernel, and the Write and Erase modules) is held on the PC and downloaded as required. The Boot Mode kernel does not need to be rebuilt for different crystal frequencies; this information is stored in the FDT project and downloaded at runtime and initial communication is achieved by the autobaud (where FDT sends 0x00 data until the device synchronises with it).

User Mode – The User Mode Test Program and User Mode Micro-Kernel need to be already programmed and executing on the device. The initial Bit Rate Register value is held in the ‘gentest.h’ file and is compiled into the User Mode Test Program; this allows the kernel initialise the serial communications interface to 9600 baud. If the clock frequency changes from the default supplied value, the initial Bit Rate Register setting in ‘gentest.h’ [normally MA\_BRR\_SCI or MA\_BRR\_SCI1] will need to be modified and the kernel recompiled. The User Mode Main Kernel and the Write and Erase modules are held on the PC and downloaded as required.

“**Protocol C**” – For devices with embedded kernels (Currently 0.18µm). These support the FDM but not the UPB.

Boot Mode – A configuration file is stored on the PC which provides FDT with details about the device, but the kernel (including the Write and Erase modules) resides internally on the device. There is no need to rebuild anything for different clock frequencies in Boot Mode; this information is stored in the FDT project and downloaded at runtime and initial communication is achieved by the autobaud (where FDT sends 0x00 data until the device synchronises with it).

User Mode – Some boards are supplied with a User Mode kernel. Similarly to the Protocol B User Mode kernels, User Mode Kernel needs to be already programmed and executing on the device. The initial Bit Rate Register value is held in the ‘gentest.h’ file and is compiled into the User Mode Test Program; this allows the kernel initialise the serial communications interface to 9600 baud. If the clock frequency changes from the default supplied value, the initial Bit Rate Register setting in ‘gentest.h’ [normally MA\_BRR\_SCI or MA\_BRR\_SCI1] will need to be modified and the kernel recompiled. The main difference to the Protocol B User Mode Kernels is that the Write and Erase modules are held internally on the device.

“**Protocol D**” – For R8C and former Mitsubishi family devices (eg. M16C, M32 etc).

The E8 and FoUSB boards can be used with some Protocol D devices. The kernels are internal to the device, but FDT uses configuration files to store information about the device characteristics.

### **3.2.5 What are the fcf and fdm files, can I edit them?**

Stored alongside the other kernel files, the .fcf (Flash Configuration File) and .fdm (Flash Development Module) files hold details about the device and kernel. Each supported device in each kernel directory has its own .fcf and .fdm file. Information such as the RAM size, Flash size, Erase Block Register details, default baud rate and default crystal frequency is stored in the fcf file and details about the allowed pin settings for the FDM is stored in the FDM file. Each of these files is protected with a checksum, so editing will render them invalid and FDT will detect this. Upon detection of an invalid checksum an fcf or fdm file FDT will warn the user but will not prevent use of the modified files.

### **3.2.6 Can I use FDT if I do not have Local Machine Administrator rights for my PC?**

Yes, but only once it is installed and configured. To install FDT and for various configuration tasks, you must have local administrator rights; this is because some settings are stored in the HKEY\_LOCAL\_MACHINE registry area. Once FDT is installed and configured you will be able to use it with certain limitations. Configuration tasks such as adding new kernel entries to the registry so that they will be available in the Project Wizard in future requires Local Machine Administrator rights. This includes adding the Prior Generic Device entry (an administrator must first connect to the Generic Device to allow the Prior Generic Device entry to be created. Additionally the FDT Advanced Setup utility which is used

for changing settings such as fine tuning of timeouts, requires the Local Machine Administrator rights to save the settings. Password administration also requires Local PC Administrator rights.

### 3.2.7 How do I recompile a kernel?

It is recommended that the original kernel is copied to another directory and that this copy is edited, thus leaving the original kernel unmodified. When using the kernel for the first time, you will need to browse to the alternate location to find the fcf file.

If the kernel has been written using Hew, there will be a Hew workspace file in the project subdirectory of the kernel. This should be opened from Hew.

If the kernel has been written based upon compilation from the command line, you will need to run the buildall.bat file from a Dos shell session after setting up the environment variables for the compiler so that it can be run from the command line. Please refer to the compiler manual for full details of how to do this. An example environment batch file used to set the environment settings for the Renesas H8 compiler version 4.0a is shown below:

Setenv.bat:

```
SET CH38=C:\hew\tools\Renesas\h8\4_0_A\INCLUDE
PATH=%PATH%;C:\hew\tools\Renesas\h8\4_0_A\bin
set DOS16M= @1M-31M
SET CH38TMP=c:\TEMP
```

Notes:

From some operating systems (eg Windows 2000), cmd.exe should be used rather than command.com.

The above batch file should be modified to reflect the path of your compiler.

Ensure that the specified temporary directory exists and is writeable.

### 3.2.8 How do I use the supplied demonstration User Mode Kernels?

To use the demonstration User Mode kernels, you must first add the User Mode Micro Kernel and User Mode Test Program from the kernel directory ("**uGenU.mot**" and "**xxxxtest.mot**") to a Boot Mode Project and build the User Area Device Image. Next, connect in Boot Mode and program the image to the device, disconnect and reset the device into User Mode. Create another project in the workspace, but this time set it as a User Mode Connection Type, and connect in User Mode. FDT will start communications with the User Mode Test Program at 9600bps and then change to the baud rate set in the project. Note that the device does not automatically erase during connections in User Mode. It is not possible to reprogram the FLASH whilst the reprogramming code is executing from the FLASH so all of the kernel modules needed to keep the connection to FDT are copied into the RAM to leave the FLASH ROM available for reprogramming. Because of this, it is possible to erase the entire flash and reprogram it in user mode, but note that if the device is reset after erasure and before it is reprogrammed, it will not be possible to reconnect in User Mode until the device has had the User Mode Micro Kernel and User Mode Test Program reprogrammed into it Boot Mode.



### 3.2.9 Can I modify the User Mode kernels to work for a different clock frequency?

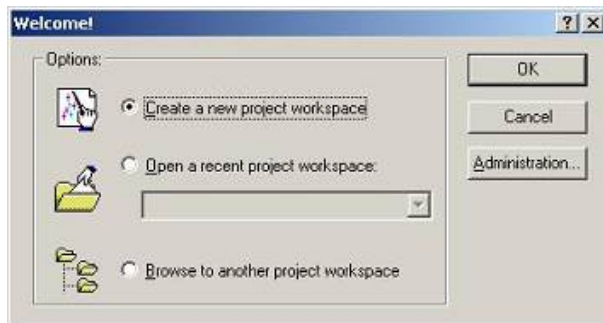
It is recommended that the original kernel is copied to another directory and that this copy is edited, thus leaving the original kernel unmodified. When using the kernel for the first time, you will need to browse to the alternate location to find the fcf file.

The User Mode kernels need to be rebuilt if a different clock frequency is going to be used. The initial Bit Rate Register value is held in the 'gentest.h' file and is compiled into the User Mode Test Program; this allows the kernel initialise the serial communications interface to 9600 baud. If the clock frequency changes from the default supplied value, the initial Bit Rate Register setting in 'gentest.h' [normally MA\_BRR\_SCI or MA\_BRR\_SCI1] will need to be modified and the kernel recompiled. Refer to the device hardware manual for details of how to calculate the Bit Rate Register value.

## 3.3 Starting FDT

### 3.3.1 Starting FDT without Security or Simple Interface Mode enabled

To start FDT, open the 'Start' menu of Windows® and select 'Programs' -> 'Renesas' -> 'Flash Development Toolkit3.07', then the 'Flash Development Toolkit3.07' shortcut. The 'Welcome' dialog box will open by default.



**Figure 3-1** Welcome Dialog

To create a new workspace, select 'Create a new Project Workspace' and click the 'OK' button (see section 3.4.1 for details). To open a recent workspace, select 'Open a recent project workspace' and the workspace to be opened from the list box. To open an existing workspace which is not in the list of recently used workspaces, select 'Browse to another project Workspace'. Finally, click the 'OK' button.

### 3.3.2 Starting FDT in Basic Simple Interface Mode

To start FDT in Basic Simple Interface Mode, open the 'Start' menu of Windows® and select 'Programs' -> 'Renesas' -> 'FLASH Development Toolkit 3.07', then the 'Flash Development Toolkit3.07Basic' shortcut. The previous FDT Basic settings will be restored, and new settings can be entered by selecting the **Options->New Settings...** menu item.

### 3.3.3 Starting FDT by double clicking on a workspace file

FDT can be started by double clicking on a workspace file from Windows® Explorer. This will cause FDT to load the selected workspace upon start up.

### 3.3.4 Starting FDT by double clicking on a data file

FDT can be started by double clicking on a registered data file from Windows® Explorer (if this default option was not unselected during installation). This will cause FDT to load the selected data file upon start up.

### 3.3.5 Starting FDT by double clicking on a w4f script configuration file

FDT can be started by double clicking on a .w4f script configuration file from Windows® Explorer. This will cause FDT to start in Wait For Script mode and load the selected configuration.

### 3.3.6 Starting FDT with Simple Interface Mode enabled

If FDT was exited from Simple Interface Mode (see section Chapter 9 ) FDT will return straight back to that mode.

### 3.3.7 Starting FDT with Access Rights password security enabled

If the Access Rights password security has been enabled, FDT will display the login dialog before allowing the workspace to be opened. See Chapter 5 for further details.

### 3.3.8 Starting FDT by double clicking on a .fpf packaged project file

FDT can be started by double clicking on a .fpf packaged project file from Windows® Explorer. This will cause FDT to unpackage the project and start in Simple Interface Mode.

### 3.3.9 Starting FDT from the command line

#### 3.3.9.1. Opening data files from the command line

```
"<FDT path>\FDT.exe" /DISCRETSTARTUP "osre \"< Filename >\""
```

*eg:*

```
"C:\Program Files\Renesas\FDT3.07\FDT.exe" /DISCRETSTARTUP "osre \"C:\Program Files\Renesas\files\myfile.mo\""
```

This is the method used when an FDT data file is double clicked in Windows® Explorer

#### 3.3.9.2. Opening workspace files from the command line

```
"<FDT path>\FDT.exe" /DISCRETSTARTUP "ow \"<Long Path Workspacename>\""
```

*eg*

```
"C:\Program Files\Renesas\FDT3.07\FDT.exe" /DISCRETSTARTUP "ow \"C:\Program Files\Renesas\FDT3.07\Workspaces\I234\I234.AWS\""
```

This is the method used when an FDT workspace is double clicked in Windows® Explorer

### 3.3.9.3. Opening data files from the command line with Access Rights password security enabled

```
"<FDT path>\FDT.exe" "arl\"<Username>\\"<Password>\\"" /DISCRETESTARTUP "osre \"<Filename >\\""
```

*eg*

```
"C:\Program Files\Renesas\FDT3.07\FDT.exe" "arl\Administrator\"\"password\"\" /DISCRETESTARTUP "osre \"C:\Program Files\Renesas\files\myfile.mo\""
```

### 3.3.9.4. Opening w4f Script Configuration files from the command line

```
"<FDT path>\FDT.exe" /DISCRETESTARTUP "w4fRun \"<filename>\\""
```

*eg*

```
"C:\Program Files\Renesas\FDT3.07\FDT.exe" /DISCRETESTARTUP "w4fRun \"C:\Program Files\Renesas\FDT3.07\myconfig.w4f\""
```

This is the method used when an FDT Script Configuration file is double clicked in Windows® Explorer

### 3.3.9.5. Opening w4f Script Configuration files from the command line with Access Rights password security enabled

```
"<FDT path>\FDT.exe" "arl\"<Username>\\"<Password>\\"" /DISCRETESTARTUP "w4fRun \"<filename>\\""
```

*eg*

```
"C:\Program Files\Renesas\FDT3.07\FDT.exe" "arl\Administrator\"\"password\"\" /DISCRETESTARTUP "w4fRun \"C:\Program Files\Renesas\FDT3.07\myconfig.w4f\""
```

### 3.3.9.6. Starting FDT Basic Simple Interface Mode from the command line

```
"<FDT path>\FDT.exe" /DISCRETESTARTUP "SimpleInterfaceMode /u \"<User Area Filename>\\" /b \"<User Boot Area or Data Area Filename>\\""
```

*eg*

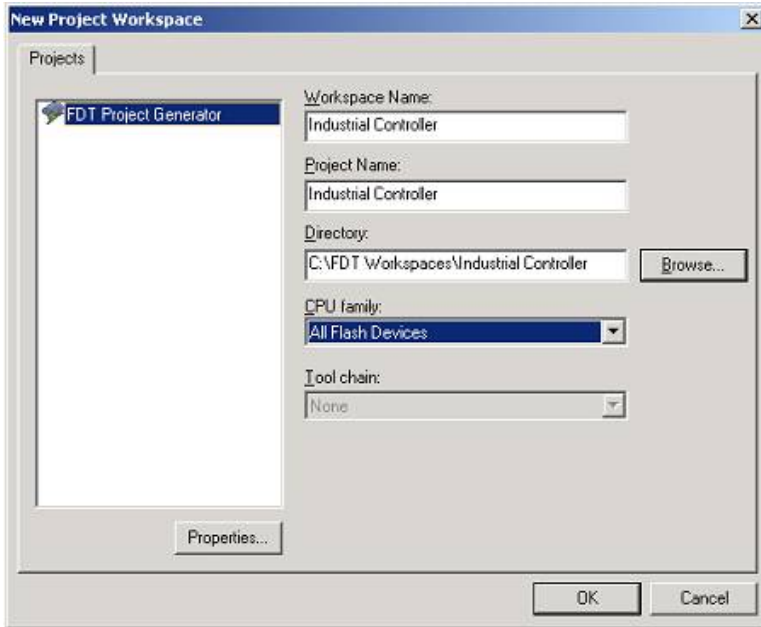
```
"C:\Program Files\Renesas\FDT3.07\FDT.exe" /DISCRETESTARTUP "SimpleInterfaceMode /u \"c:\upload1.mo\" /b \"c:\upload0.mo\""
```

Note that the /u <User Area Filename> and the /b <User Boot Area> are optional. Additionally, adding /r causes the last settings to be restored and the configuration Wizard will not be displayed.

## 3.4 Creating a New Workspace and Project

When FDT is started in full mode (not FDT Basic Simple Interface Mode), a workspace and project must be opened or created. See Chapter 10 Basic Simple Interface Mode for details on programming without setting up a workspace and project.

### 3.4.1 New Project Workspace



**Figure 3-2** New Workspace

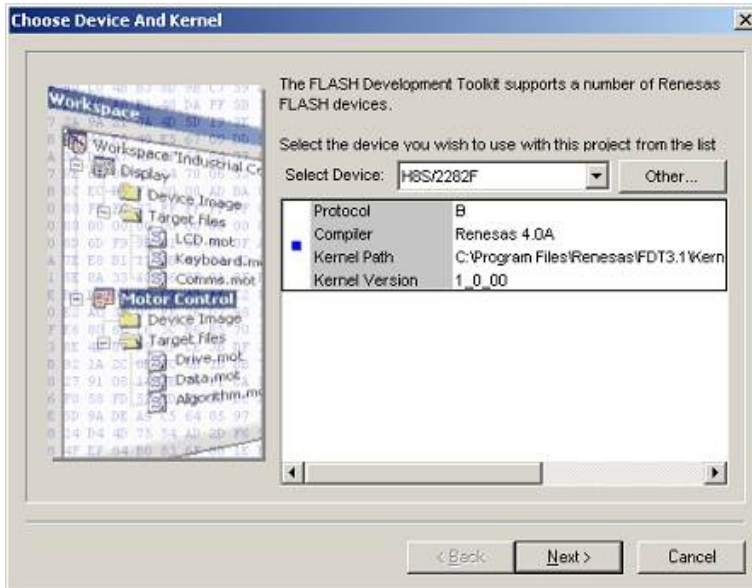
Enter the workspace name (the project name is the same by default), then click the 'OK' button. If you wish to change or create a directory, input a directory name or use 'Browse...' button.

### 3.4.2 Choose Device and Kernel (not Generic Boot\*)

Select the device you wish to use from the drop-down list. It is possible to have more than one kernel option displayed, and double clicking on a kernel will open an optional “readme.txt” file that has information about the kernel (such as the device, version number and the compilers it was created with).

Select the required kernel from the displayed list, or if you have an alternative kernel elsewhere, which is not in the list, select ‘Other...’ to specify the kernel Flash Configuration File (.fcf).

- \* For connection to a Generic Boot Device see the “Configuring a Project for a Generic Boot Device” section.



**Figure 3-3 Device and Kernel Selection**

### 3.4.3 Communications Port

Select the port (and interface if available) from the drop-down list, then click the 'Next' button. Note that in addition to the serial ports, there is also an option to use the FDM (USB interface board). For some kernels there are also other port options, such as USB Direct (eg. for the 2215UF), E8, E8Direct and FoUSB. For Protocol A (legacy) kernels the 'HMSE UPB' interface is available in addition to Direct Connection.

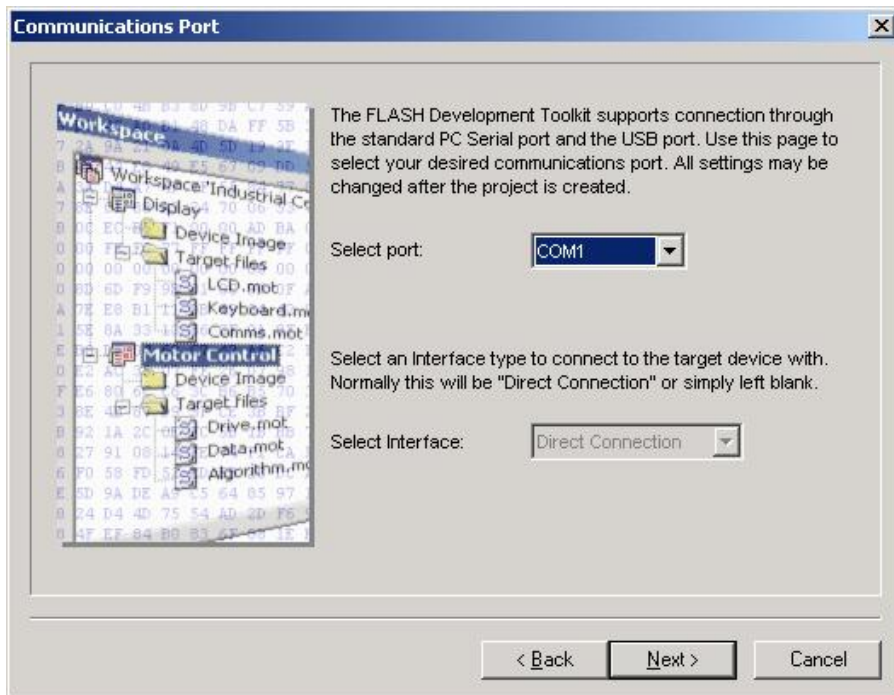


Figure 3-4 Communications Port

### 3.4.4 Device Settings

Enter values for the device settings; enter the input clock, then select the main and peripheral clock multipliers from the drop-down lists, and click the 'Next' button.

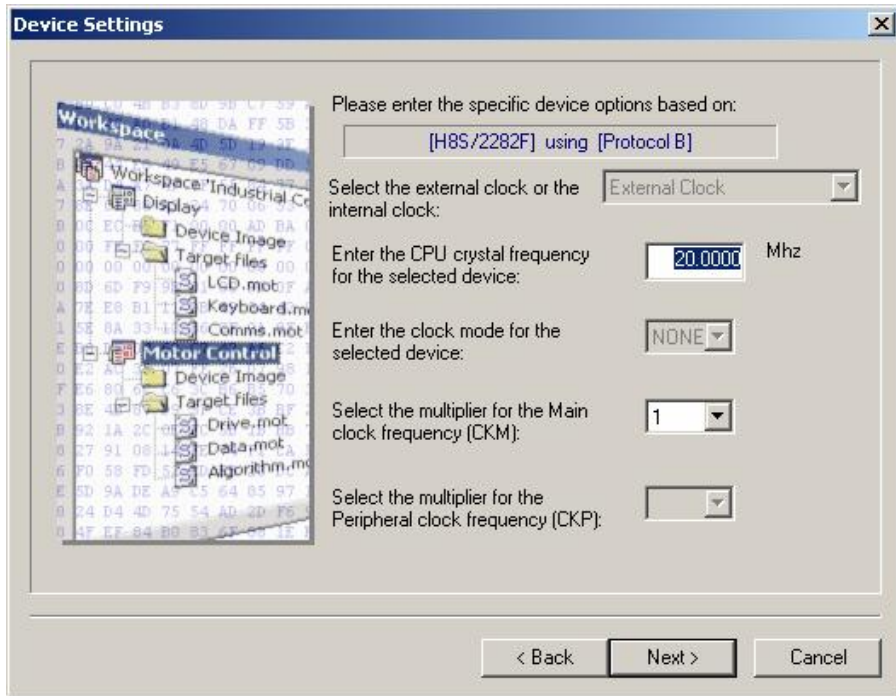


Figure 3-5 Device Settings

Supplementary:

#### 1. Clock Setting

Allow selection of the device external clock (default) or one or more of the device internal clock speeds.

#### 2. CPU Crystal Frequency

Enter the frequency of the CPU clock or the crystal generator. This can be specified up to 4 decimal places. The target device, the kernel and the Clock Mode determine the valid range.

#### 3. Clock Mode

If enabled, select the required value from the drop-down list.

#### 4. Multiplier for CKM

If enabled, select the input clock multiplier for the master (system) clock. The available options depend upon the target device, the kernel and the Clock Mode.

#### 5. Multiplier for CKP

If enabled, select the input clock multiplier for the peripheral clock. The available options depend upon the target device, the kernel and the Clock Mode.

**Note** To enter the clock mode, input clock, and multipliers, refer to the device hardware manual and/or the target board manual to confirm the values required.

### 3.4.5 Connection Type

Select whether the connection is for Boot Mode or User Mode in the 'Connection Type' dialog. Additionally, for a connection to a kernel that is still running on the device from a previous session, select the 'Kernel already running' check box. Next, select the communication speed (bps) if the default value is not correct or not present. Speeds where the target error rate is greater than 4%, or the FDM/E8Direct error rate is greater than 0% are automatically removed from the recommended list. When using the FDM/E8Direct the 'User Specified' connection speed option is enabled; when selected, this allows the user to specify a non-standard baud rate. The error rate will be displayed – this takes into account the target peripheral clock rate, the target Bit Rate Register calculation formula, and the FDM/E8Direct characteristics.

Click the 'Next' button to continue.

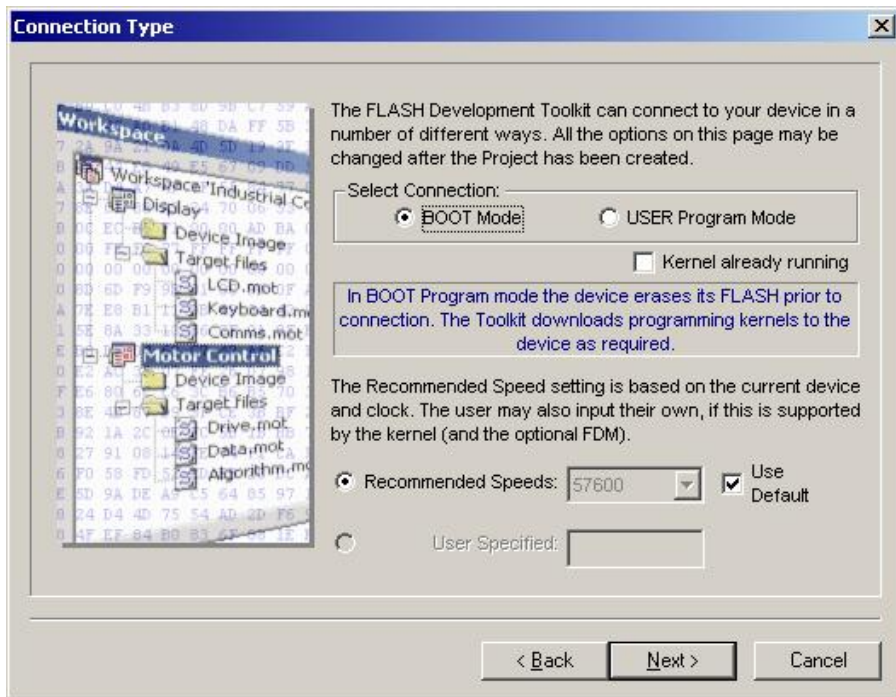


Figure 3-6 Connection Type

Note that 'Use Default' is designed for use with the Renesas Evaluation Board or Evaluation Development Kits. Uncheck this to allow manual selection of the bit rate.



### 3.4.6 Programming Options

Select the protection level for programming the FLASH ROM and the messaging level, then click the 'Finish' button, or if the FDM/E8Direct was selected, click 'Next' to continue.

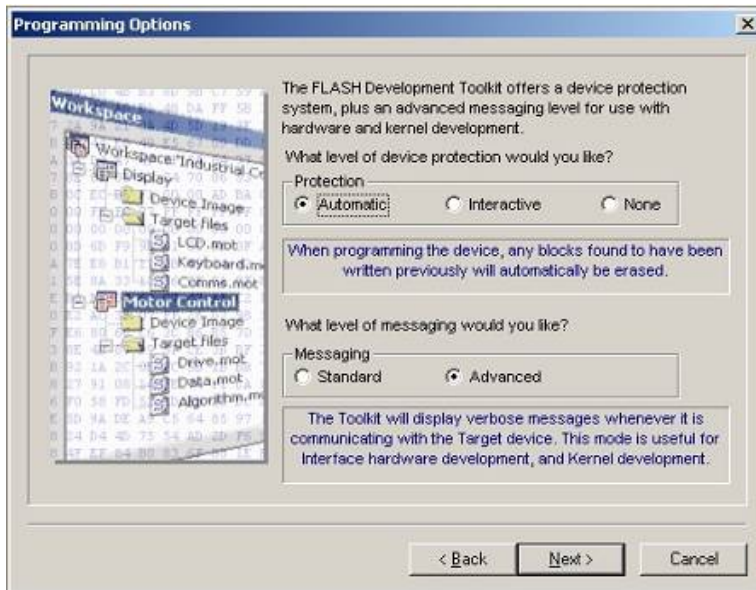


Figure 3-7 Programming Options

### 3.4.7 FDM/E8Direct Pin Settings

[Only if the FDM or E8Direct is selected] Select the 'Operating Mode' and 'Clock Mode' and if 'User Defined', set the pin settings. [For 'Boot Mode' projects set the 'Boot Mode Outputs' and the 'Boot Mode Setting' for the required mode, for 'User Mode' projects set the 'User Mode Outputs', 'User Mode Setting' and the 'User Program Mode Setting' for the required mode.] Click the 'Next' button to continue. “\*Mode Outputs” sets which pins are driven, “\*Mode Setting” sets whether output pins are driven High (checked) or Low (Unchecked).

**Notes:** Setting the mode pins using the FDM or E8Direct requires caution. Please confirm the pin settings with the hardware manual even if a default setting is available. Should you be using the E8Direct then a message-box will also appear requesting confirmation of this warning. Note that in some devices, changing the mode pin setting changes the maximum supply voltage.

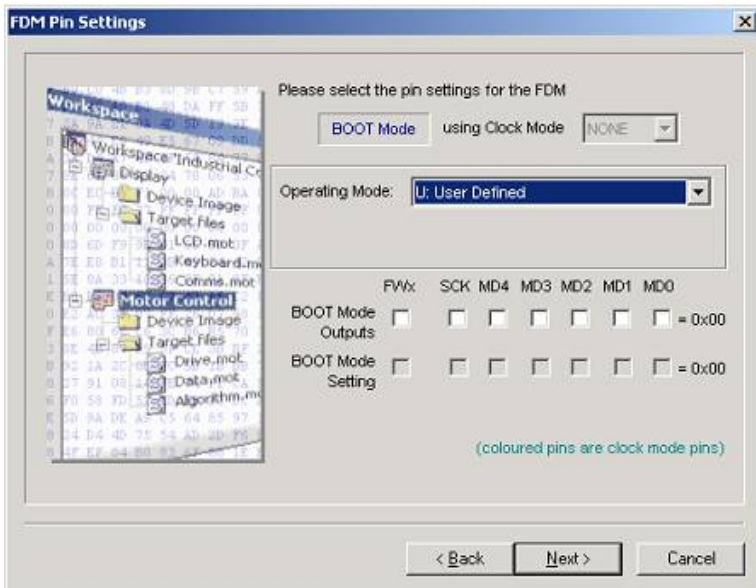


Figure 3-8 FDM Pin Settings

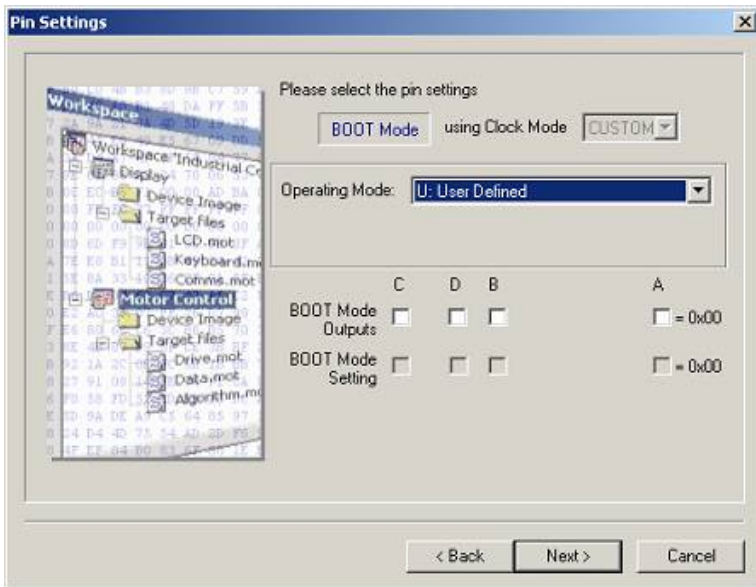


Figure 3-9 E8Direct Pin Settings

### 3.4.8 Reset Pin Settings

[Only if the FDM or E8Direct is selected] Select the 'Operating Mode' and 'Clock Mode' and if 'User Defined', set the Reset pin settings. Set the 'Reset Outputs' and the 'Reset Setting' for the mode required immediately after disconnection. Click the 'Finish' button. “\*Mode Outputs” sets which pins are driven by the FDM, “\*Mode Setting” sets whether output pins are driven High (checked) or Low (Unchecked).

**Notes:** Setting the mode pins using the FDM or E8Direct requires caution. Please confirm the pin settings with the hardware manual even if a default setting is available. Note that in some devices, changing the mode pin setting changes the maximum supply voltage.

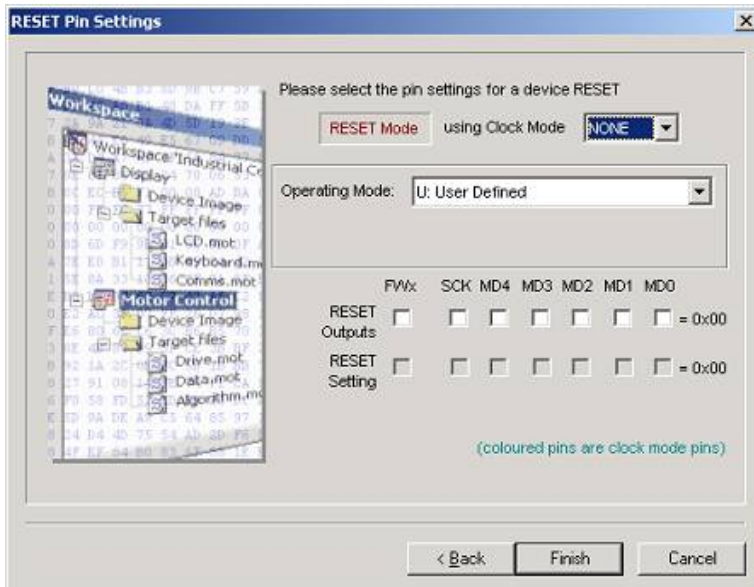


Figure 3-10 FDMRESET Pin Settings

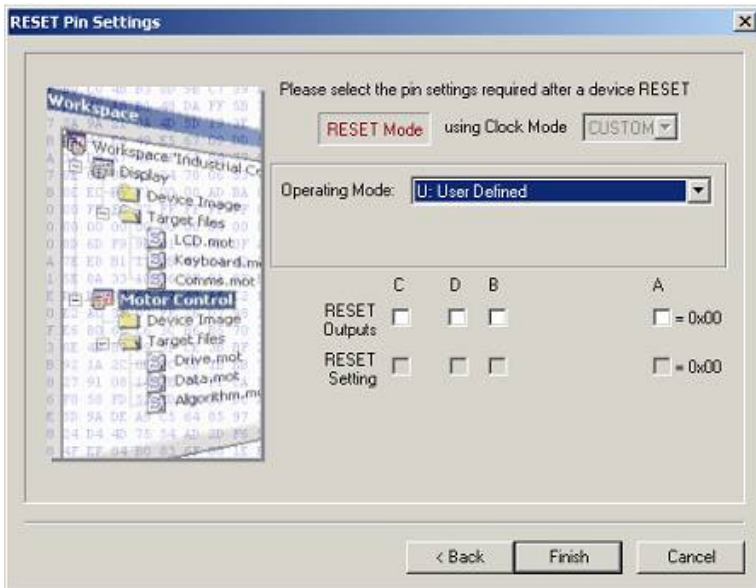


Figure 3-11 E8DirectRESET Pin Settings

## 3.5 Configuring a Project for a Generic Boot Device

It is also possible to connect to a Generic Boot Renesas Device - where you do not need a kernel or configuration file installed, instead FDT extracts the required information from the device itself. Because of this, the device must be connected and set into Boot Mode at the time you create the project. If a Generic Boot device is selected, then a series of screens appear displaying the results of the inquiry functions, you will be required to make selections about the device and clock mode (one boot program may be shared for multiple devices).

### 3.5.1 New Project Workspace

Enter the workspace name, then click the 'OK' button. (Specify whether or not a directory is to be created and a location.)

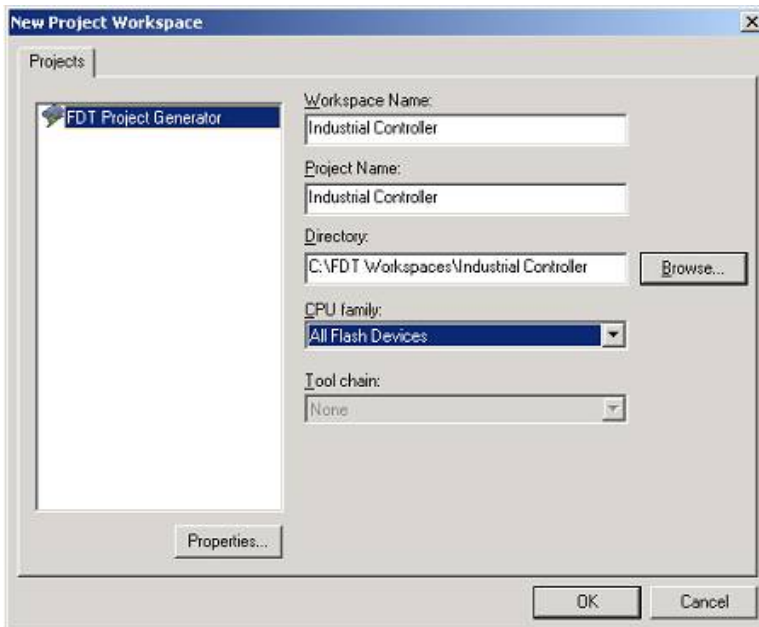
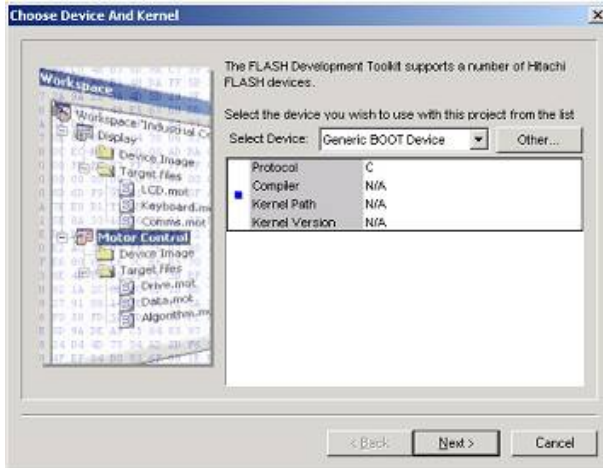


Figure 3-12 New Workspace

### 3.5.2 Generic Boot - Choose Device and Kernel

Select 'Generic Boot Device' (or a device supported by Generic Boot) and click 'Next'.



**Figure 3-13 Generic Boot Device Selection**

*Note* - Some Protocol C devices which appear in the device list are supported by using Generic Boot. This means that when one of these devices is selected in the Project Wizard, FDT will behave as if "Generic BOOT Device" was selected, and will connect to the device to obtain the device characteristics instead of using an existing fcf file for this information.



**Figure 3-14 Device Support via Generic Boot**

### 3.5.3 Generic Boot - Communications Port

Select the port (and interface if available) from the drop-down list, then click the 'Next' button.

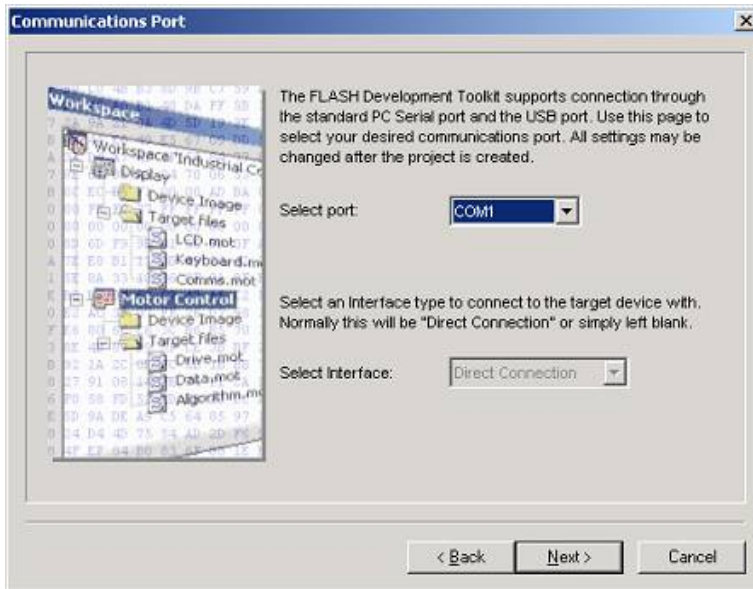


Figure 3-15 Communications Port

### 3.5.4 Generic Boot - FDM Pin Settings [FDM connection only]

If the FDM is selected the following screen will be shown. Set the mode pin settings required to place the device into Boot Mode. Click 'OK' to continue.



Figure 3-16 FDM Generic Device Pin Settings

**Note** Setting the mode pins using the FDM requires caution. Please confirm the pin settings with the hardware manual even if a default setting is available. Note that in some devices, changing the mode pin setting changes the maximum supply voltage.

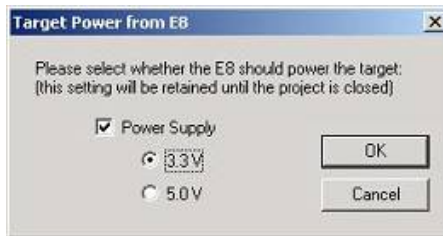
**When the ‘Select USB Device’ Dialog is displayed you must select which FDM unit is to be used to set up the connection.**



**Figure 3-17** Select USB Device

### 3.5.5 Generic Boot - E8Direct Pin Settings [E8Direct connection only]

If E8Direct is selected the following two screens will be shown. The first allows optional power to be supplied to the target board via the E8 (see also section 3.6.1)



**Figure 3-18** E8 Target Power Supply Selection

**WARNING: Damage to the target device may occur if the incorrect voltage setting is chosen!**

This will be followed by the mode pin settings dialog, required to place the device into Boot Mode. Click ‘OK’ to continue.





**Figure 3-19 E8DirectGeneric Device Pin Settings**

**Note** Setting the mode pins using the E8Direct requires caution. Please confirm the pin settings with the hardware manual even if a default setting is available. Note that in some devices, changing the mode pin setting changes the maximum supply voltage.

If the E8 has not yet been restarted as an E8Direct device then the following messages will appear in the FDT output window:

```
Preparing for re-enumeration of E8 to E8Direct...
Sending re-enumeration command...
Request sent successfully, closing E8 Comms...
```

At the same time Windows® will re-enumerate the device (see also Chapter 13 - E8Direct).

**When the ‘Select USB Device’ Dialog is displayed you must select which E8Directunit is to be used to set up the connection.**



**Figure 3-20 Select USB Device**

### 3.5.6 Generic Boot - Confirmation

FDT then waits for you to confirm that the device is connected and powered, and if not using the FDM or E8Direct, in Boot mode. FDT will try to connect to the device when you click 'OK'.



Figure 3-21 Generic Boot Confirmation

After this, FDT displays a dialog that displays the boot progress. At various points during the Generic Boot process, the user is required to select or confirm details of the device; see below for details.

### 3.5.7 Generic Boot – Select Device

Because some devices share a boot program, a list is provided of all those supported. The user should select the correct device from the list then click 'OK'.

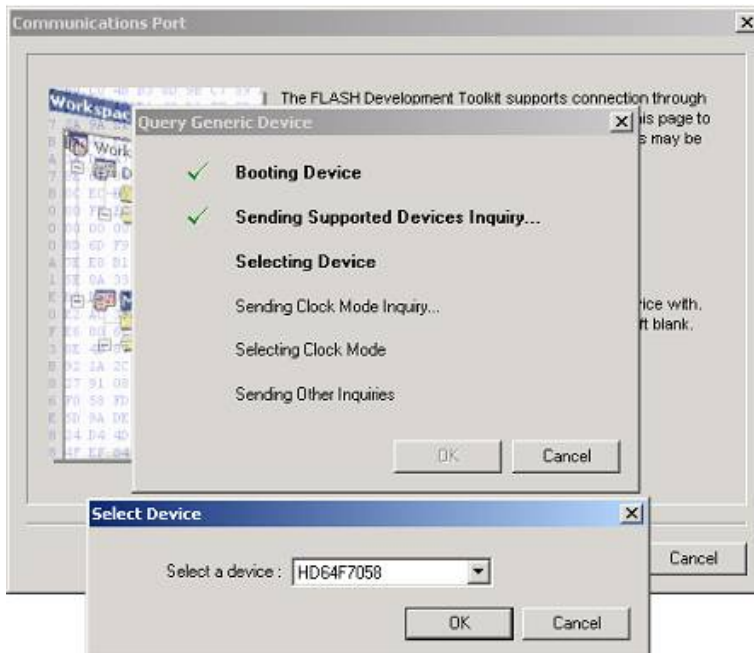


Figure 3-22 Generic Boot – Device

### 3.5.8 Generic Boot – Select Clock Mode

Because some devices have a number of Clock Modes, a list is provided of all possible Clock Modes. The user should select the correct Clock Mode from the list then click 'OK'. Please check with the hardware manual / target board manual to ensure that the correct mode is chosen.

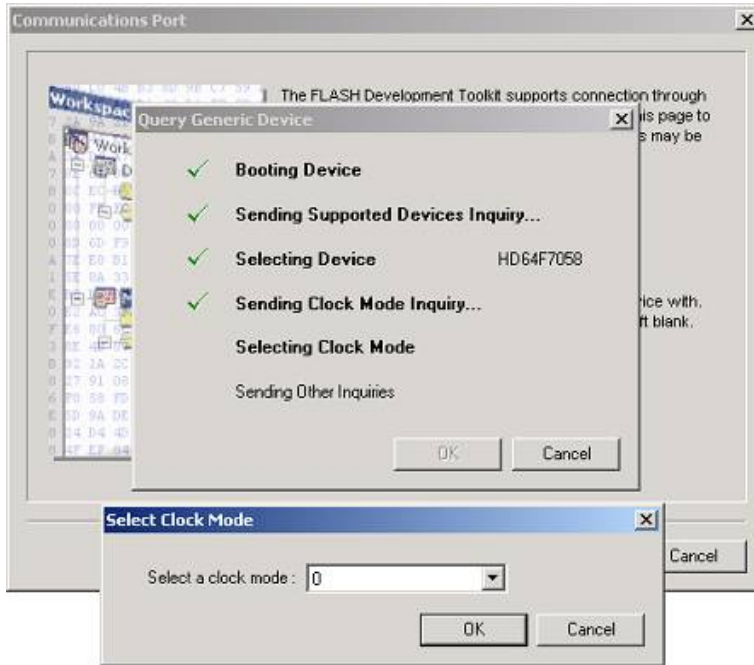


Figure 3-23 Clock Mode

### 3.5.9 Generic Boot – Setup Complete

Having selected the Clock Mode, the Generic wizard completes as below.



Figure 3-24 Generic Boot setup complete

The Project Wizard will then continue through the remaining pages

### 3.5.10 Generic Boot – Device Settings

Please Refer to Section 3.4.4

### 3.5.11 Generic Boot - Connection Type

Please Refer to Section 3.4.5

### 3.5.12 Generic Boot – Programming Options

Please Refer to Section 3.4.6

### 3.5.13 Generic Boot – Reset Pin Settings [FDM connection only]

Please Refer to Section 3.4.8

### 3.5.14 Generic Boot – Reinterrogate on connect option

The 'Programmer' tab of the 'Flash Properties' dialog allows the user to configure whether FDT should reinterrogate the Generic device on each connection. The default is set to 'No', so after a connection has been established once, FDT re-uses the data collected from the initial inquiry functions. If this option is changed to 'Yes', FDT will interrogate the device for the device details each time it connects in Boot Mode. This means that one Generic Boot project will work with any Generic Boot Device.

### 3.5.15 Prior Generic Device

Selection of the 'Prior Generic Device' device allows a connection to the last device configured using 'Generic Boot Device'. It uses the .fcf file which was generated when first connecting using 'Generic Boot Device'.

### 3.6 Connecting to the device

This is achieved by selecting the Device->Connect to Device menu item, using the Ctrl-Alt-C hotkey, or clicking the Connect toolbar button. If a flash operation (Upload, Download, Blank Check, Checksum etc) is selected before a connection is established, FDT will automatically perform a connect operation first.

#### 3.6.1 E8 USB Connection

When connecting for the first time since opening the workspace, and using the E8 USB interface board, a dialog will be displayed to allow selection of whether the E8 should supply power to the target, and at which voltage level. The selected power level will be supplied to the target device when connecting and switched off when disconnecting. The power level selection will be remembered until the workspace is closed.

**WARNING: Damage to the target device may occur if the incorrect voltage setting is chosen!**

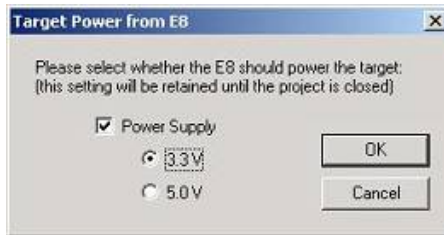


Figure 3-25 E8 Target Power Supply Selection

#### 3.6.2 ID Check Dialog

For devices which support the ID Check for unlocking the device, the ID check dialog will be displayed if the devices are not unlocked with automatic unlocking. The Browse button allows selection of a mot or an ID file to collect the ID codes from.

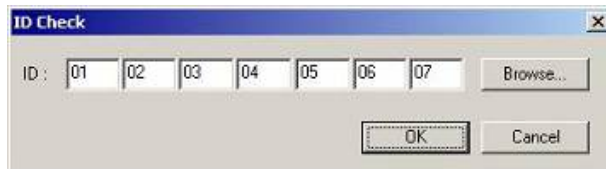


Figure 3-26 ID Check Dialog

### 3.6.3 Locking

For devices which support locking of erase blocks FDT may also check with the user whether to unlock the device. This is dependent on the current locking settings (see more on this in the Locking Connection section )



Figure 3-27 Locking Query Dialog

## 3.7 Programming the Data to the FLASH ROM

### 3.7.1 Programming an individual file to the User Area (normal flash area)

1. Select [**Project** ->**Add Files...**] or press the INS key to add the file to be downloaded to the project.
2. Click the right mouse button on the file displayed in the workspace window, and select 'Download File'. Programming will be completed when 'Image successfully written to device' is displayed in the output window.

### 3.7.2 Programming multiple files to the User Area (normal flash area)

1. Add all the required target files to the project by selecting [**Project** ->**Add Files...**] or pressing the INS key. Note that multiple files can be added at once from the 'Add File(s)' dialog.
2. Build the device image for the User Area (create a file which is the result of merging all of the included target files) by selecting the menu item:  
[**Project** ->**Rebuild Image**]
3. Click the right mouse button on the Device Image file (<ProjectName>.ddi) displayed in the workspace window, and select 'Download Image' (or select the menu item  
[**Project** ->**Download Image**])
4. Programming is complete when 'Image successfully written to device' is displayed in the output window.

### 3.7.3 Programming an individual file to the User Boot Area

1. Select [**Project** -> **Add Files...**] or press the INS key to add the file to be downloaded to the Project.
2. Click the right mouse button on the file displayed in the workspace window, and select '**User Boot Area File**' to mark this file as being for the User Boot Area.
3. Click the right mouse button on the file displayed in the workspace window, and select '**Download File to [User Boot Area]**'. Programming will be completed when 'Image successfully written to device' is displayed in the output window.

### 3.7.4 Programming multiple files to the User Boot Area

1. Add all the required target files to the project by selecting [**Project ->Add Files...**] or pressing the INS key. Note that multiple files can be added at once from the 'Add File(s)' dialog.
2. Click the right mouse button on each file displayed in the workspace window which is destined for the User Boot Area, and select '**User Boot Area File**' to mark it as being for the User Boot Area.
3. Build the device image for the User Boot Area (create a file which is the result of merging all of the included target files) by selecting the menu item: [**Project ->Rebuild Image**]
4. Click the right mouse button on the Device Image file (<ProjectName>.ddi) displayed in the workspace window, and select '**Download Image**' (or select the menu item [**Project ->Download Image**])
5. Programming is complete when '**Image successfully written to device**' is displayed in the output window.

### 3.7.5 Programming some files to the User Area and other files to the User Boot Area

1. Add all the required target files to the project by selecting [**Project ->Add Files...**] or pressing the INS key. Note that multiple files can be added at once from the 'Add File(s)' dialog.
2. Click the right mouse button on each file displayed in the workspace window which is destined for the User Boot Area, and select '**User Boot Area File**' to mark it as being for the User Boot Area.
3. Build the device image for the User Boot Area (create a file which is the result of merging all of the included target files) by selecting the menu item: [**Project ->Rebuild Image**]
4. Select the menu item [**Project ->Download Image**]
5. Programming is complete when '**Image successfully written to device**' is displayed in the output window for both the User Area and the User Boot Area

### 3.8 Erasing Data from the FLASH ROM

1. Select [Device -> Erase FLASH blocks...] to display the 'Erase Blocks' dialog box.
2. Select the block(s) to be erased. (When the checkbox is ticked, the block is selected.)\*
3. Click the 'Erase' button to start erasure.
4. When 'Erase complete' is displayed in the output window, erasure is completed.



Figure 3-28 Erase Blocks

**\*Note** To erase all blocks, clicking the 'Select All' button will tick every available check box. To erase all non-blank blocks, clicking the 'Select Written' button will tick every associated check box. Ticking the box of a parent item will toggle the state of all child items. A greyed check-box indicates that only some of the child items are ticked.



### 3.9 Uploading the FLASH ROM Data

1. Display the 'Upload Image' dialog box from [Device -> Upload Image...].either
  2. Select the Address tab
  3. Enter the start and end addresses, or the start address and length.
  4. Click the 'Upload' button to start reading.
- or**
5. Select the Tree tab
  6. Select the blocks to be uploaded (corresponding to the erase blocks and/or the device flash areas).
  7. Click the 'Upload' button to start reading.

When 'Upload operation complete' is displayed in the output window and the uploaded data is displayed in the hex editor, reading is complete.

If the user specifies an Absolute image, the uploaded data will be placed in an image the size of the FLASH memory at the same address it was uploaded from. Use an Absolute Image if the uploaded data needs to be added to the project. It is not possible to upload a region outside of the FLASH into an Absolute Image.

**Note** When FDT is started in boot mode, the flash memory data has normally already been erased.

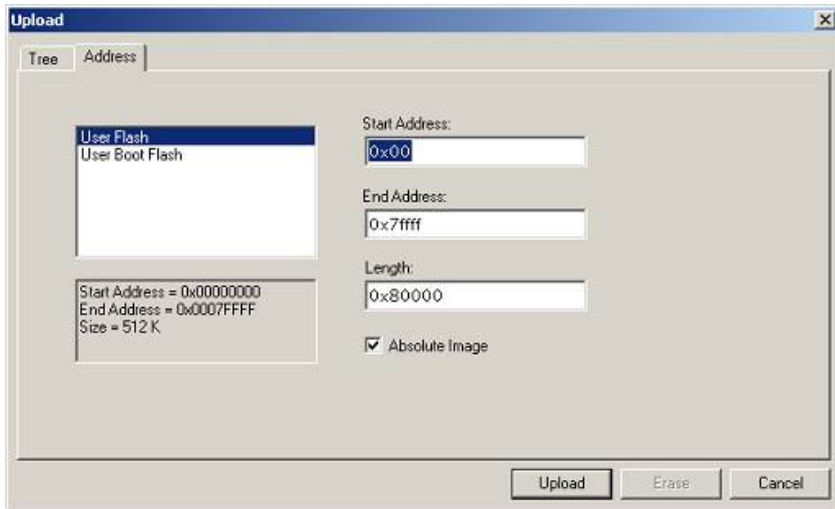
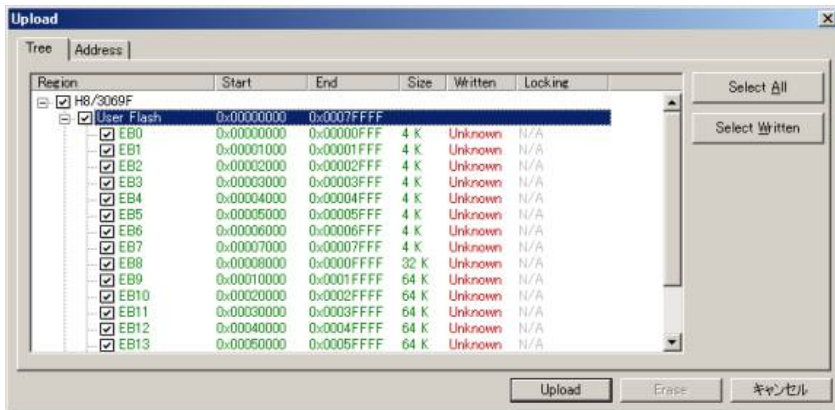


Figure 3-29 Upload Image Dialog - Address



**Figure 3-30 Upload Image Dialog – Tree**

The Upload via erase block has the same selection functionality as the Erase block dialog itself, i.e. ‘Select All’ will check every block (a complete device upload) and ‘Select Written’ will check the non-blank blocks (an upload of programmed areas).

### 3.10 Performing a Blank Check

For kernels which support it, a blank check can be performed by selecting the ‘Device->Blank Check’ menu item. This will perform a blank check on all available device areas (e.g. User and User Boot).

### 3.11 Disconnecting from the device

This is achieved by selecting the ‘Device-> Disconnect’ menu item, or clicking the Disconnect toolbar button.

### 3.12 Saving a Workspace

When the [File -> Save Workspace] menu option is selected, the FDT workspace can be saved.

### 3.13 Closing a Workspace

Select [File -> Close Workspace] to close the FDT workspace. If the workspace or its project has changed, a dialog box asks if the user wishes to save the project. Select “Yes” if the workspace is to be saved, “No” if the workspace is not to be saved, and “Cancel” to return to the workspace. If you are prompted to save the session, then select “Yes”. These prompts can be switched on and off by changing the setting in the Workspace tab of the [Tools -> Options] dialog.

### 3.14 Exiting FDT

To exit FDT, select [File -> Exit], press the Alt + F4 key, or select the ‘Close’ option from the system menu (the system menu can be opened by clicking the icon in the upper-left are of the title bar).

# Chapter 4 Wait For Script

## 4.1 Wait For Script Overview

This mode allows FDT to be automated and controlled by a text file containing a script of commands. Once in this mode FDT acts as a server which polls for the existence of a file in a predetermined location. Once this file is found, FDT executes each command in order and can then optionally delete or move the script file upon completion, so that FDT is ready to look for the next script file. This allows FDT to connect, program, erase, blank check etc, under the control of an external application, via the use of a simple text file.

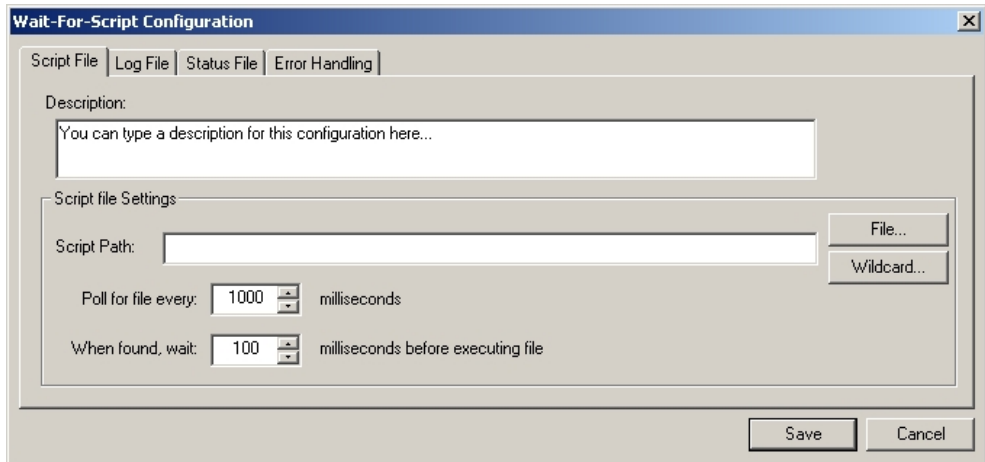
The messages in the output window can optionally be echoed to a log file for later analysis and traceability. Additionally, a status file can be created, which holds information about the state of the current and previous script processes. This status file can be used by the external application to determine whether the script has been processed successfully.

To set up FDT Wait For Script, a configuration file needs to be created to specify the search path for the script file and other parameters such as the settings for the log and status files, and how to handle errors.

## 4.2 Wait For Script – New Configuration

This is invoked from the Tools, Wait For Script, New Configuration... menu.

### 4.2.1 Script File Page



**Figure 4-1** Script File Page

This is the only essential page in the wait for script configuration.

#### 4.2.1.1. Description

This is an optional text description that may be used to describe, e.g. who created the file, why it was created, what it does, etc. It is not used in any way, other than for display.

#### 4.2.1.2. Script Path

This is the only mandatory field. This edit box allows the user to set up where to find the script file to run. This can either be a specific file (e.g. c:\Scripts\MyScript.fsf) or a directory and wildcard combination (e.g. c:\Scripts\\*.fsf).

The *File...* button to the right of the edit field invokes browse dialog to allow the user to navigate to the file they want to use. The *Wildcard...* button to the right of the Wildcard edit field invokes a standard Windows directory selection dialog. If a directory is chosen from this dialog then a trailing \\*.fsf (fsf is the default script extension) will also be appended to the path as a convenience to the user. This field can also be edited manually.

#### 4.2.1.3. Timings

This allows user control over how often FDT polls for the script file and how long FDT should wait before processing, after finding the script file and having access to it.

#### 4.2.1.4. Save

This saves the FDT Script Configuration file (with the default extension .w4f). FDT will prompt to ask if you want to execute the saved script immediately.

### 4.2.2 Log File Page

This is an optional page; leave the fields blank if no log file is required. It can either append to or overwrite existing files

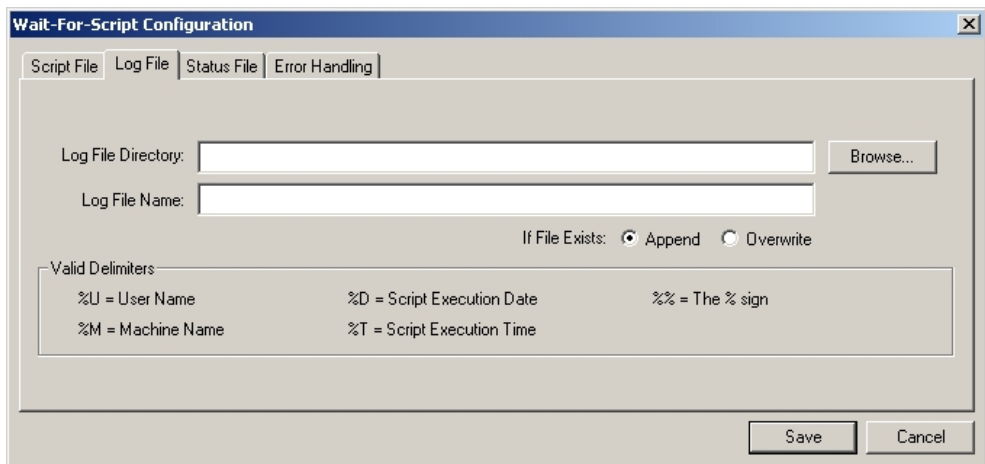


Figure 4-2 Log File Page

### 4.2.2.1. Log File Directory

Enter the directory that the optional log file should be created in. FDT can insert additional details into the directory path such as the user name, machine name, date or time by using the %U, %M, %D and %T delimiters.

### 4.2.2.2. Log File Name

Enter the name of the optional log file. FDT can insert additional details into the file name such as the user name, machine name, date or time by using the %U, %M, %D and %T delimiters.

### 4.2.2.3. Save

This saves the FDT Script Configuration file (with the default extension .w4f). FDT will prompt to ask if you want to execute the saved script immediately.

## 4.2.3 Status File Page

This is an optional page; leave the Enable status output to file checkbox unchecked if no status file is required. See section 4.7 for details of the contents of the status file.

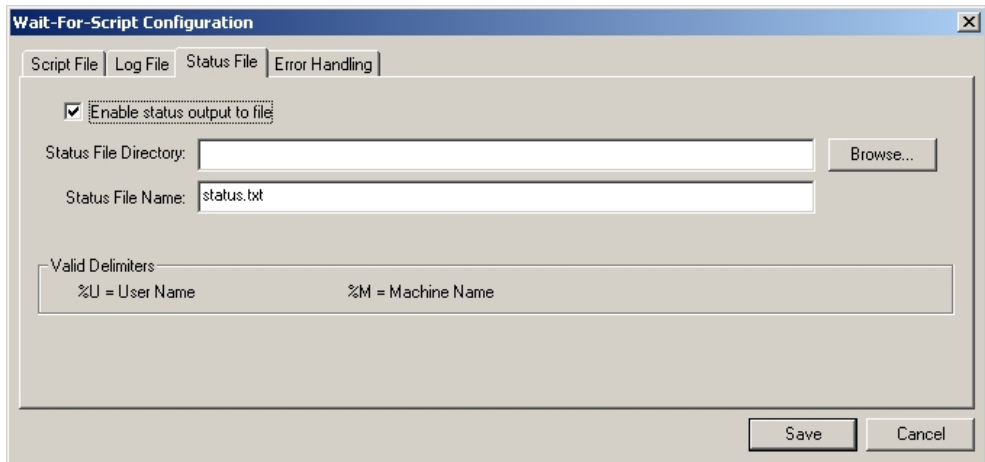


Figure 4-3 Status File Page

### 4.2.3.1. Status File Directory

Enter the directory that the optional status file should be created in. FDT can insert additional details into the directory path such as the user name and machine name by using the %U and %M.

### 4.2.3.2. Status File Name

Enter the name of the optional status file. FDT can insert additional details into the file name such as the user name and machine name date or time by using the %U and %M delimiters.

### 4.2.3.3. Save

This saves the FDT Script Configuration file (with the default extension .w4f). FDT will prompt to ask if you want to execute the saved script immediately.

### 4.2.4 Error Handling Page

This is an optional page.

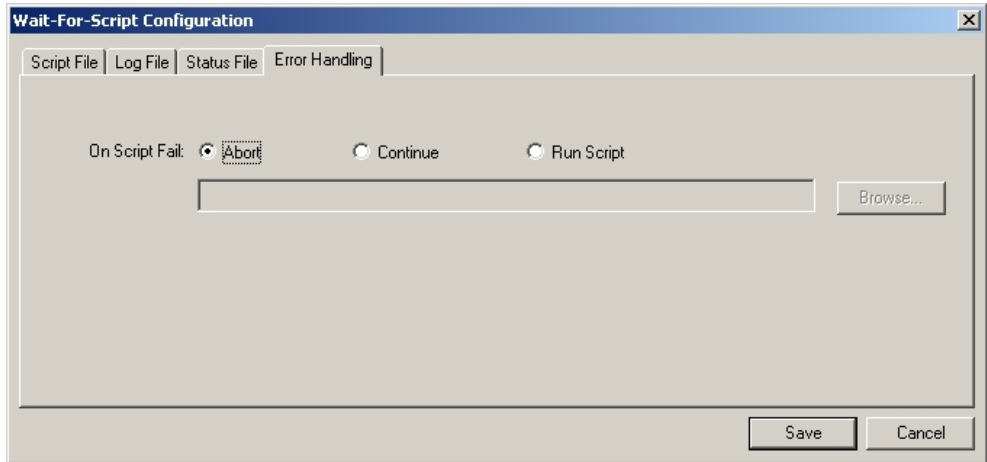


Figure 4-4 Error Handling Page

#### 4.2.4.1. On Script Fail

Abort - leave the On Script Fail radio button set to the default Abort to cause FDT to abort processing of further scripts upon encountering an error.

Continue - allows FDT to ignore the fact that the error occurred and continue processing further scripts.

Run Script - forces FDT to process an error handling script upon encountering an error. The script file to execute on script fail can be entered in the input box manually or by using the browse button. Typically a fail script may contain the disconnect command, an alert command and the abort command.

```
# Script processing failure handling script example 1

# Disconnect to allow reconnection
Disconnect

# Display a message to the user
Alert
output|*****
Alert output|***** THE CURRENT SCRIPT PROCESSING HAS FAILED
*****
Alert
output|*****
Alert output|** The script which was being processed will now be deleted. **
```

```

Alert output|** Please reset the board and regenerate the script to retry **
Alert
output|*****

# Delete the script file which failed
# so that FDT will wait for it to be regenerated before retrying
delete $script

# Script processing failure handling script example 2

# Disconnect to allow reconnection
Disconnect

# Display a message to the user
Alert
output|*****
Alert output|***** THE CURRENT SCRIPT PROCESSING HAS FAILED
*****
Alert
output|*****
Alert output|**          Wait For Script Processing will now be
aborted          **
Alert output|**          To retry, use Tools, Wait for Script,          **
Alert output|**          Run configuration... and select the configuration          **
Alert
output|*****

#Stop script processing
Abort

```

#### 4.2.4.2. Save

This saves the FDT Script Configuration file (with the default extension .w4f). FDT will prompt to ask if you want to execute the saved script immediately.

### 4.3 Wait For Script – Edit Configuration...

This is invoked from the ‘Tools’, ‘Wait For Script’, ‘Edit Configuration...’ menu and allows you to edit the configuration settings for a given configuration file.

### 4.4 Wait For Script Run Configuration...

This is invoked from the Tools, Wait For Script, Run Configuration... menu and allows the user to browse to select the desired Wait For Script Configuration file (.w4f). FDT then enters Wait For Script mode:

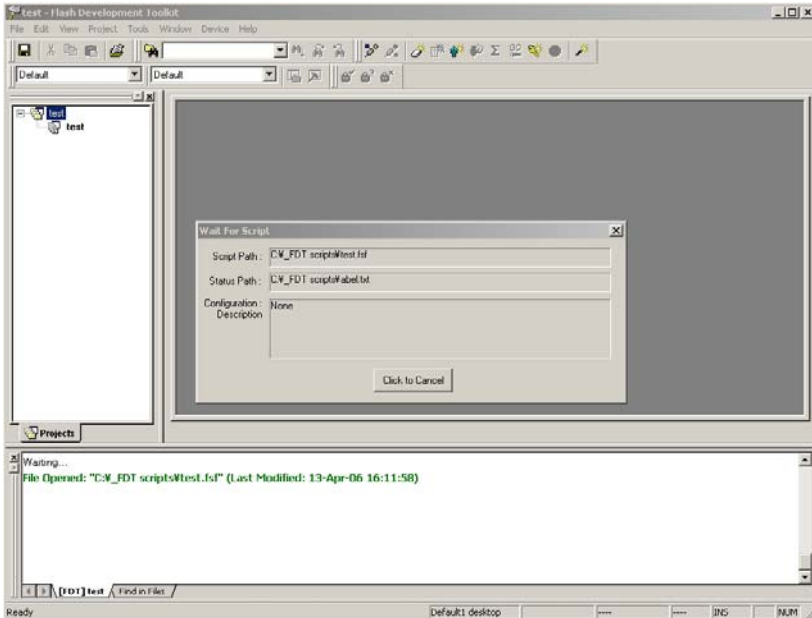


Figure 4-5 Wait For Script execution

### 4.5 Wait For Script – Script Commands

A script file is nothing more than a text file containing a series of commands that FDT will recognise, one per line. Each command may also have a series of arguments associated with it, on the same line. By default these script files will have a “.fst” extension.

The syntax for a script line is the command, followed by a space, followed by required arguments (separated by the ‘|’ operator). Optional extra arguments follow these, and can be placed in any order.

Normally a script file should end with the ‘delete \$script’ command. This causes FDT to delete the script once processing is finished, thus preventing the same script file being processed more than once. FDT will then start looking for the next script file. Alternatively the move command could be used to move the script file to an alternative directory once completed

Each script command should be written on a separate line of the text file; FDT will cope with DOS and UNIX style line breaks (CR or CR LF). File paths within the script file can include drive letters or network paths (including IP addresses).



### 4.5.1 Workspace

Description = loads a specified workspace into FDT (Note: a valid workspace and configured project must be opened to use any of the device-related commands)

Arguments:

<Workspace Path>	Required	The path to the workspace
------------------	----------	---------------------------

Example:

Workspace C:\Workspaces\3069serial\3069serial.AWS

NB: FDT will use the active project from this workspace – there is currently no facility for changing which is the active project from the script

### 4.5.2 Connect

Description = connect to a device

Arguments:

<PORT>		Optional	If the port specified does not match the project port settings the command will fail. For USB Direct, FDM, E8 and E8Direct connections this must be set to “USB”
When <PORT> is “USB”	“USB” <USB PID> <USB BID>	Optional (One of)	When using USB Direct it is possible to pre-empt the Select Device connection dialog normally seen with these settings.
	USB” “FDM” “SN: ”<SERIAL NUM>		If using an FDM the PID should be set to “FDM” and the BID should be set to “SN: “ followed by the FDM’s serial number
	“USB” “E8” <USB BID> <CONNECT VOLTAGE> <DISCONNECT VOLTAGE>		If using E8 or E8Direct the BID of the device, the initial connection voltage and the voltage to be applied at disconnect time can be specified. (NB – currently supported voltage values are “5.0”, “3.3” or “0”)
	“USB” “E8Direct” <USB BID> <CONNECT VOLTAGE> <DISCONNECT VOLTAGE>		

Note that the connection will only be attempted with the project port setting and not to change the port used to connect. Specification of the port in this command is only to force a failure if the project does not match this setting, and to suppress the Select Device connection dialog for USB devices by allowing the PID/BID or FDM and Serial number to be preselected.

Examples:

Connect

Connect COM1

Connect USB|0x000D|0

Connect USB|FDM|SN: 93506

Connect USB|E8|0001|5.0|0.0

### 4.5.3 Download

Description = download the selected project image to the device

Arguments:

"image"	Required	Download the device image
<filename>	(One of)	Download the named file
"user"	Required (One of)	Download the user image\Download to the user area
"userboot"		Download the user boot image\Download to the user boot area. This can also be used to download to the User Area.
"both"		Download the user and user boot image (only valid when specifying "image")
"norebuild"	Optional	Don't rebuild the image(s) before downloading, even if they are out of date (only valid when specifying "image")
"verify"	Optional	Perform a readback verify when the download completes

Example:

Download image|user|norebuild|verify

Download C:\files\MySrecord.mot|userboot

### 4.5.4 Erase

Description = erase specified flash blocks on the device

Arguments:

<block num(s)>	Required (One of)	A list of block numbers to erase
"written"		Erase all blocks currently marked as written
"device"		Erase the entire device

Example:

Erase 0|1|4|7

Erase device

#### 4.5.5 BlankCheck

Description = check if the device is blank

Arguments:

"userboot"	Optional	Tells FDT to check the user boot area of flash (if supported)
"stoponwritten"	Optional	Stop processing the script if the device is not blank
"stoponblank"	Optional	Stop processing the script if the device is blank

Example:

```
BlankCheck
```

```
BlankCheck userboot|stoponwritten
```

#### 4.5.6 Upload

Description = upload data from the device

Arguments:

<filename>	Required	File to write the upload out to – will be created or overwritten if necessary. If not needed, specify "none"
<start address>	Required	Start address in Hex to start the upload from
<length>	Required	Number of bytes (in Hex) to upload
"noabsolute"	Optional	Don't upload an absolute image
"userboot"	Optional	Upload from the user boot area or the data area (if supported)

Example:

```
upload none|0|0x6000
```

```
upload c:\upload\uploadtest.mot|0x100|0x200|userboot|noabsolute
```

#### 4.5.7 Checksum

Description = perform a checksum on a device

Arguments:

"device"	Required	Checksum the device
----------	----------	---------------------

Example:

```
Checksum device
```

#### 4.5.8 Disconnect

Description = disconnect from a device. Note that issuing of a disconnect command even if FDT is not connected is allowed. This can be useful for ensuring that FDT is disconnected before a connect operation.

Arguments:

"reset"	Optional	Tells FDT to perform a reset as part of the disconnect
---------	----------	--

Example:

Disconnect

Disconnect reset

#### 4.5.9 RebuildImage

Description = rebuild the specified image from the project target files

Arguments:

"user"	One of	Rebuild the user image
"userboot"		Rebuild the user boot image
"both"		Rebuild the user and user boot image

Example:

RebuildImage user

RebuildImage both

#### 4.5.10 # Comment

Description = placed at the start of a line, this symbol comments out the rest of the line

Arguments:

<i>NONE</i>	-	-
-------------	---	---

Example:

# this is my commented line

#### 4.5.11 Alert

Description = indicate to the user that something has happened

Arguments:

"beep"	Required	FDT will output a warning beep
"output"	(One of)	FDT will output a message on the active project tab

Example:

Alert output|Please make sure the board is connected

#### 4.5.12 Pause

Description = this command will pause script file execution for the specified number of milliseconds

Arguments:

<time>	Required	Number of milliseconds to pause for
--------	----------	-------------------------------------

Example:

Pause 2000

#### 4.5.13 Delete

Description = deletes the script file. NB: FDT will process the script no further than this line before deleting it

Arguments:

"\$script"	Required	Delete the script file
------------	----------	------------------------

Example:

Delete \$script

#### 4.5.14 Move \ Copy

Description = move or copy the script file. NB: FDT will process the script no further than this line

Arguments:

"\$script"	Required	Copy\move\rename this script file
<filename> <directory>	Required	The file\directory the source file is to be copied\moved\renamed to

Example:

```
Move $script \\server\networkfiles\done.txt
```

```
Copy $script[C:\backups\
```

#### 4.5.15 Abort

Description = stop processing the script and return failure. NB: This command is primarily designed to be used in an error processing script

Arguments:

<i>NONE</i>	-	-
-------------	---	---

Example:

```
Abort
```

## 4.6 Wait For Script – Example Script File

What follows is an example of how a script file for FDT may look. This highlights some of the available commands and gives an idea of how they might be used:

```
# EXAMPLE SCRIPT FILE
# open this workspace and associated project
# (for which the flash project wizard must have completed successfully)
workspace C:\Workspaces\2215usbdirect\2215UsbDirect.AWS

# Ensure that comms is closed (in case previous attempt failed and didn't
disconnect)
disconnect

# connect over the USB to the device with this Product ID + Bus ID
connect USB|0x000D|0

# do an initial blank check
blankcheck

# download the file "C:\2215uAuto.mot" and verify that programming was OK
# (script processing will stop if this verify fails)
download C:\2215uAuto.mot|user|verify

# upload data from address 0 to address 0x1000 and save to file "upload.mot"
upload c:\UPLOADS\upload.mot|0x0|0x1000

# erase all blocks
erase device

# do a blank check (and stop if the user area is dirty)
blankcheck stoponwritten

# rebuild the user area image
RebuildImage user

# download the user image
download image|user

# get the checksum for the user area of the device
checksum device

# pause for 100 milliseconds
pause 100

# disconnect from the device
disconnect
```

```

# Display message in output window
alert output|This text will be displayed in the output window and the log file

# delete this script
# (will stop FDT from running through the file again)
delete $script

```

## 4.7 Wait For Script – Status File Format

While FDT is in Wait For Script mode, a remote user may wish to find out what state the application is currently in. This is achieved by having a status file that FDT writes to as it carries out actions and which an external user can read. This has the following options:

By default, status file writing will be disabled. Ticking the Enable checkbox will open the Directory and Name fields for editing; allowing the user to specify where the status file will be created.

```

[FDTScript Status v1.0] ← The configuration file version

CurrentScript=      ← The location of the current script being executed, or the search path being
used whilst waiting

UserID=            ← The user name of the person running the copy of FDT writing to this file

Machine=          ← The machine name the copy of FDT is executing on

LastUpdate=       ← The time and date this file was last changed

CurrentStatus=    ← The current state of FDT. This will be either 'waiting' (if FDT is looking for
a script), 'processing' (if FDT has found a script and is using it), 'error processing' (if something has gone
wrong with the script) or 'waiting for user input' (if a dialog is waiting for user intervention). If FDT exits
wait for file mode correctly this entry will be blanked

PreviousScript=   ← The location of the previously run script

PreviousState=    ← The outcome of the previously run script. This will either be 'COMPLETED'
or 'FAILED'

JobNo=            ← An incrementing number of successfully completed runs. This number will
wrap back to 0 when it reaches 32,768.

```

When FDT begins a Wait For File session, with status writing enabled, it will first check for the existence of the specified file. If it is not found then it will be created and FDT will continue. If it is found the file will be opened and checked. If CurrentStatus is blank (implying that a previous session exited correctly and that no-one else is using the file) FDT will continue. If an entry is found in CurrentStatus, e.g. 'Processing' then FDT will alert the user, via a message box, that a previous session may have crashed or that another copy is running and using this file also. The option will then be given to continue and overwrite the values or abort.



# Chapter 5 Access Rights

## 5.1 Access Rights Overview

To prevent accidental changing of settings when using FDT to program devices (especially in a multi-user environment), the Access Rights feature allows creation of usernames and passwords to give different levels of access to different users. Typically, there would be three levels of access rights:

**Administrator** level which allows the user to create and modify user accounts

**Supervisor** level which allows project creation and editing

**SIM User** level which allows a Simple Interface Mode user the right only to program the device image from Simple Interface Mode. A supervisor would need to configure the project before a SIM User could use FDT.

## 5.2 Enabling Access Rights

FDT is supplied by default with the user accounts specified above but without passwords set. You will not need to log in to FDT until the Access Rights feature is enabled by setting the password for the built in **Administrator** account. This is achieved by using the 'Tools', 'Access Rights', 'Administration...' menu item. The first time this is run, you will be prompted to enter a password for the Administrator account.

## 5.3 Access Rights Administration

This is reached via the 'Tools', 'Access Rights', 'Administration...' menu item.



Figure 5-1 User Admin

### 5.3.1 Disable All

This button allows disables all user accounts and switches off Access Rights password security

### 5.3.2 Modify User

This button allows modification of the settings for the selected user by invoking the Modify User dialog.

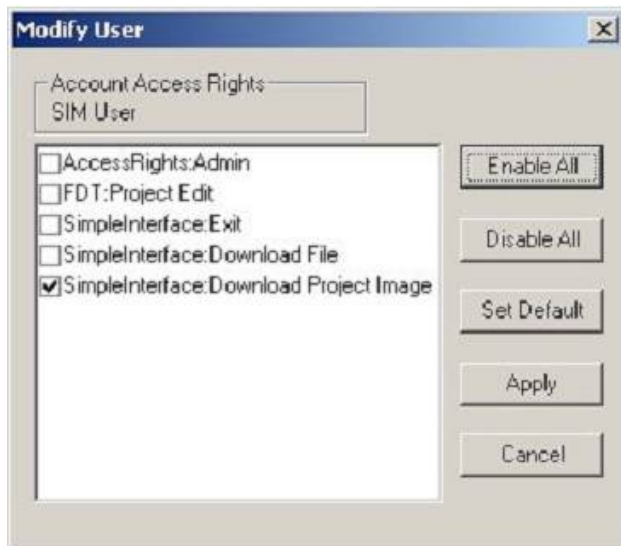


Figure 5-2 User Admin

The available access options are:

**AccessRights:Admin** – Allows administration of user accounts and passwords

**FDT:Project Edit** – Allows editing of FDT Projects; enables the ‘Back to Project’ button in Simple Interface Mode

**SimpleInterface:Exit** – Allows exiting of FDT; enables the ‘Exit’ button in Simple Interface Mode.

**SimpleInterface:Download File** – Allows user selection of the file to download; enables the ‘Download File’ section of the Simple Interface Mode.

**SimpleInterface:Download Project Image** – Allows the user to download the project image(s); enables the ‘Download Project Image’ section of the Simple Interface Mode.

### 5.3.3 Add User

This button allows creation of new user accounts. FDT will prompt for a user name and password then invokes the Modify User dialog.

#### **5.3.4 Delete User**

This button allows deletion of user accounts.

#### **5.3.5 Reset Pwd**

This button allows setting of the password for the selected user.

#### **5.3.6 Close**

This button closes the User Admin Dialog.



# Chapter 6 Configuring the User Interface

When the user interface for FDT was designed an attempt was made to make frequently used operations quickly accessible and have related operations grouped in a logical order. However, when the user is in the middle of a long session he may find it more useful to have a different arrangement of the user interface items. FDT facilitates this user customisation. This chapter describes how the user interface can be arranged and how various aspects of the display can be customised.

## 6.1 Arranging Windows

### 6.1.1 Minimising windows

When an operation is finished on an open Editor window but it may be necessary to look at it in its current state later, it can be reduced to an icon, this is called *minimizing* the window. To minimize a window click on the 'minimize' button of the window.

The window is minimised to an icon at the bottom of the Editor window.

**Note** The icon may not be visible if there is another window open over the bottom of the screen.

To restore the icon back to a window either double click on the icon, or click once to pop up the icon menu and select [**R**estore], or select required window from the Menu bar [**W**indow] drop-down menu.

### 6.1.2 Tiling windows

After some time there may be many windows open on the screen. All the windows can be arranged in a tile format with none of them overlapping each other using the "Tile Windows" option. To invoke this select the [**W**indow->**T**ile Horizontally] or [**W**indow->**T**ile Vertically] menu option.

All currently open windows are arranged in a tile format. Windows that are minimised to icons are not affected.

### 6.1.3 Cascading windows

The open windows can be arranged in a cascaded format by using the "Tile Windows" option. To invoke this select the [**W**indow->**C**ascade] menu option.

## 6.2 Locating Currently Open Windows

When many windows are open in the FDT application window it is quite easy to lose one of them behind the others.

### 6.2.1 Locating a specific window

There are two methods to find the lost window.

1. To select a specific window, invoke the [**W**indow] menu. Click on the required window from the list of open windows at the bottom of the menu. The currently selected window will have a check mark next to it in the window list.

The window selected will be brought to the front of the display. If it is minimised the icon is restored to a window.

2. A specific window can also be selected by clicking on the tab containing the file name at the bottom of the Editor window. If the window is not minimised it will be brought to the front of the display. If it is minimised, the minimised icon will be brought to the front of the display.

### **6.3 Enabling/disabling the Toolbar**

The user has the option to enable or disable the Toolbar. By default, the Toolbar is displayed at the top of the FDT application window. To disable the display of the Toolbar, select each of the displayed toolbars, and disable their view.

To display or disable toolbars, use the [**Tools->Customise**] menu option or Right Click on the Toolbar area..

### **6.4 Enabling/disabling the Workspace Window**

The user has the option to enable or disable the Workspace. By default, the Workspace is displayed. To disable display of the Workspace, select the [**View->Workspace**] menu option.

If the disable option is selected, the Workspace will be disabled and removed from the FDT application window display. To re-enable the Workspace display, select the [**View->Workspace**] menu option again. The Workspace will be enabled and added to the FDT application window display and the menu option checked.

### **6.5 Enabling/disabling the Output Window**

The user has the option to enable or disable the Output Window. By default, the Output Window is displayed. To disable display of the Output Window, select the [**View->Output**] menu option.

If the disable option is selected, the Output Window will be disabled and removed from the FDT application window display. To re-enable the Output Window display, select the [**View->Output**] menu option again. The Output Window will be enabled and added to the FDT application window display and the menu option checked.

## 6.6 Customising the Toolbar

The selection and arrangement of buttons displayed on the Toolbar can be customised to suit a user's requirements. To change the display, invoke the [**T**ools->**C**ustomise...] menu option. The defaults are to display the Editor, Search, FDT, QzROM and S-Record toolbars. Note that the S-Record Toolbar option is only available when there is a data file open.

The 'Customise' dialog box will be displayed:

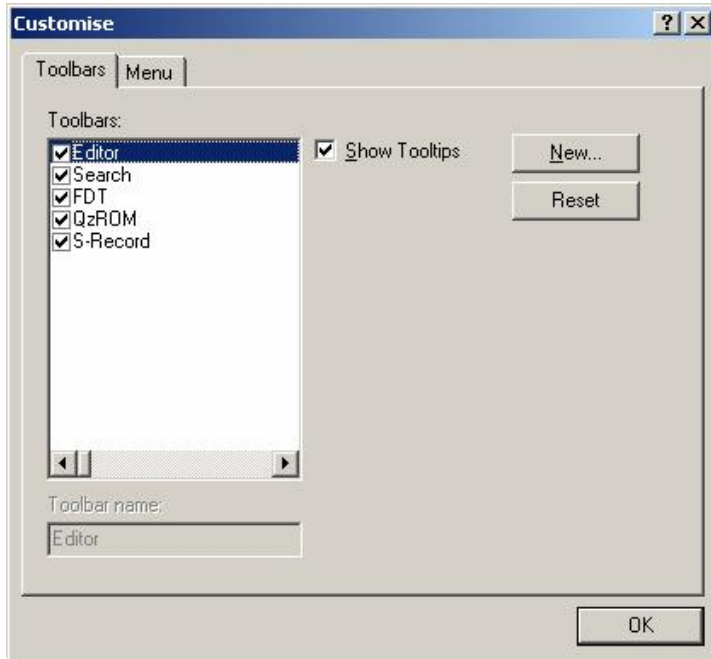


Figure 6-1 Customise FDT

The 'Customise' dialog box has a number of tabs, which are further described in Chapter 8 Windows.

### 6.6.1 Toolbars Tab

The Toolbars tab allows the user to select which toolbars are displayed. The default is to display only the Editor, Search, and FDT toolbars.

The **New...** button launches the 'Toolbar Name' dialog box which allows the user to name and generate a new toolbar. After entering the new name and clicking on **OK**, the new name will be added to the Toolbars list and an empty toolbar is displayed on the interface.

Selecting the Commands tab allows the user to select and add buttons to the new toolbar. The user can place the new toolbar anywhere on the GUI by dragging it with the mouse.

When a user created toolbar is selected in the Toolbars list, the **Reset** button is renamed to **Delete**. Clicking on this button will remove the new toolbar from the list and from the GUI.

The **Reset** button resets the selected toolbar to its default settings.





# Chapter 7 Menus

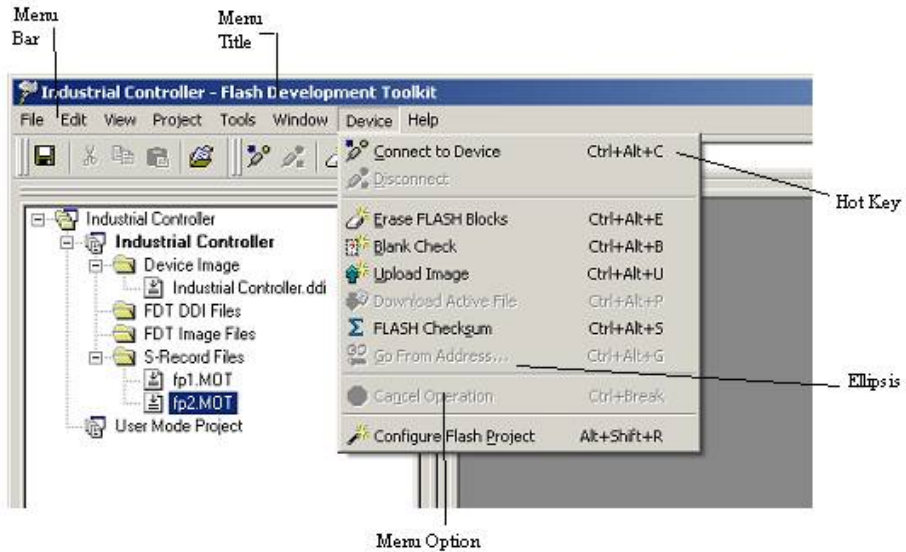


Figure 7-1 FDT Menu

## 7.1 File Menu

The 'File' menu is used for aspects of the program that access data files.

### 7.1.1 Close

This will close the current active file in the editor window.

### 7.1.2 New Workspace...

Launches the 'New Workspace' dialog box allowing the user to specify the name and location of a new workspace and creates a new workspace directory.

### 7.1.3 Open Workspace...

Launches the 'Open' dialog box allowing the user to open an existing FDT Workspace file (.aws).

### 7.1.4 Save Workspace

Saves the details of the currently active Workspace.

### 7.1.5 Close Workspace

Closes the currently active Workspace.

### **7.1.6 Open an S-Record...**

Launches the 'Open an S-Record' dialog box allowing the user to open an existing file. The file may be an S-Record file (.rec, .mot, .a20, .a37, .s2), a Device Image file (.fpr, .ddi), or a binary file (.bin, .cde).

### **7.1.7 Save**

Saves the current active file.

### **7.1.8 Save As...**

Launches the 'Save As' dialog box allowing the user to save and name the currently open file..

### **7.1.9 Exit**

Closes and exits the FDT application.

## **7.2 Edit Menu**

The 'Edit' menu is used for aspects of the program that access or alter data in the Editor window.

### **7.2.1 Undo**

Allows the user to reverse the previous editing operation.

### **7.2.2 Redo**

Allows the user to reverse the previous Undo operation.

### **7.2.3 Cut**

This will remove the contents of the highlighted block from the window and place it on the clipboard in the standard Windows manner. This option is only available if a block is highlighted.

### **7.2.4 Copy**

This will copy the contents of the highlighted block to the clipboard in the standard Windows manner. This option is only available if a block is highlighted.

### **7.2.5 Paste**

This will copy the contents of the Windows clipboard into the child window at the current cursor position.

### **7.2.6 Select All**

Selects all the information in the active file in the editor window.

### **7.2.7 Find...**

This will launch the 'Find' dialog box allowing the user to enter either hexadecimal or ASCII data. The scope of the search is bounded by the area selected, for the active file in the Editor window. If a match is found, the Editor window is updated to display the found data.

### **7.2.8 Find In Files...**

This launches a dialog to allow data to be located in files external to the active file.

### **7.2.9 Replace...**

This will launch the 'Replace' dialog box allowing the user to enter either the hexadecimal or ASCII data to be found and the replacement data, in the same format. The scope of the search is bounded by the area selected, for the active file in the Editor window. If a match is found, the Editor window is updated to display the found data, click on the now enabled Replace button to replace the data.

## **7.3 View Menu**

The 'View' menu is used to display or remove FDT GUI features, such as the Workspace and Output Windows.

### **7.3.1 Workspace**

Displays the Workspace window.

### **7.3.2 Output**

Displays the Output window.

## **7.4 Project Menu**

The 'Project' menu provides high level control and facilities for projects within the workspace.

### **7.4.1 Set Current Project**

This option allows the user to select the active project where there are multiple projects in the workspace.

### **7.4.2 Insert Project...**

This option allows the user to create a new project, and add it to the workspace.

### **7.4.3 Add Files...**

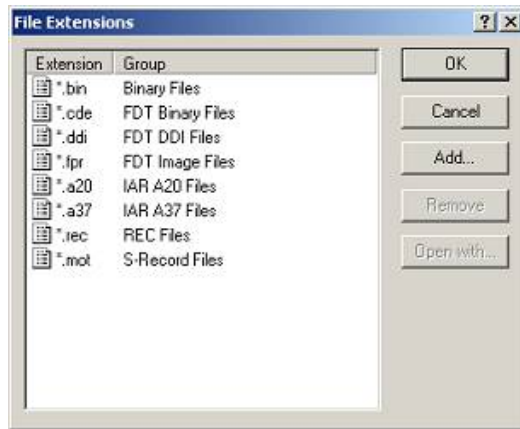
Launches the 'Open' dialog box allowing the user to add data files to the project.

### **7.4.4 Remove Files...**

Removes the selected file from the project.

### **7.4.5 File Extensions...**

This option allows the user to select which file extensions are recognized by FDT and configure how they are handled.



**Figure 7-2 File Extensions Dialog**

#### 7.4.5.1. File Extensions Add...

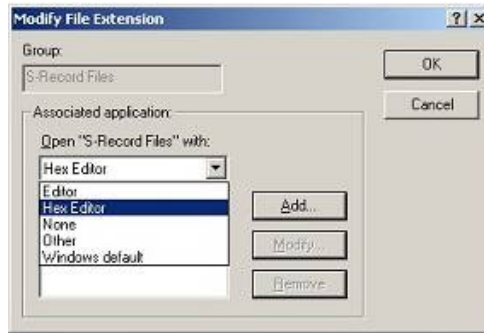
Selecting 'Add...' allows FDT to handle files of new extension types



**Figure 7-3 Add File Extensions Dialog**

### 7.4.5.2. File Extensions Modify...

Selecting 'Modify...' allows selection of how FDT handles files of this extension type.



**Figure 7-4** Modify File Extensions Dialog

### 7.4.6 Rebuild Image

Rebuilding the image merges all files in the project that are not excluded. The default ROM area is the User Area, some devices also have a User Boot Area or a Data Area. FDT will warn if the component files overlap. Building an image can be advantageous when the data you wish to program is made up of separate component files – especially if data from different files will reside in the same erase block area because downloading each component file separately could cause the device to erase the block before the second download if automatic protection is used.

### 7.4.7 Download Image

Downloads the image built by a Rebuild Image. If the image is out of date (i.e. source files have changed) then FDT will offer to rebuild before the download occurs.

### 7.4.8 Field Programming

Publish... Create a single .fpf file which contains all kernel and data files to send to another FDT user.

Run Published File... Unpackage a .fpf file and use it in Simple Interface Mode.

See Chapter 11 Field Programming for more details

## **7.5 Tools Menu**

The ‘Tools’ menu is used to launch additional FDT features, which are not related to either projects or target devices.

### **7.5.1 Administration...**

This displays a dialog with the tools available to the user under different categories.

### **7.5.2 Customise...**

This allows the user to customise FDT, and is covered in more detail in section 8.9Customise - Toolbars.

### **7.5.3 Options...**

Allows access to additional features, some of which are reserved for future enhancements. The workspace tab is the most useful tab as it allows the user to specify options such as whether to open the last workspace at start-up, whether to prompt before saving workspaces and sessions, and the default directory for new workspaces.

### **7.5.4 Format Views...**

This allows the user to change the appearance of different windows.

### **7.5.5 Simple Interface...**

Simplifies the appearance of FDT. For more details see section Chapter 9

### **7.5.6 Wait For Script**

This allows creation, editing or running of a Wait For Script configuration. See Chapter 4 for further details.

### **7.5.7 Access Rights**

#### **7.5.7.1. Access Rights – Login...**

This allows changing of which user is logged in to FDT. The hotkey Ctrl-Shift-U also invokes the Login dialog. See Chapter 5 for further details

#### **7.5.7.2. Access Rights – Change Password...**

This allows the currently logged in user to change their password. See Chapter 5 for further details

#### **7.5.7.3. Access Rights – Administration...**

This allows configuration of user names and passwords for restricting access to FDT. See Chapter 5 for further details

## 7.6 Window Menu

The 'Window' menu is used to alter the display of currently open windows within the FDT GUI. Files displayed in the Editor window are appended to the following list, these files are identified by their filename and the currently active file is denoted by check mark.

### 7.6.1 Cascade

This option allows multiple windows to be staggered so that each file is visible.

### 7.6.2 Tile Horizontally

This option allows the windows to be displayed with maximum possible horizontal width.

### 7.6.3 Tile Vertically

This option allows the windows to be displayed with maximum possible vertical height.

### 7.6.4 Arrange Icons

This option arranges the file icons in rows at the bottom of the screen.

### 7.6.5 Close All

This option closes any files open in the editor window.

### 7.6.6 'file name'

This option becomes active when a file name is selected and checked.

## 7.7 Device Menu

The 'Device' menu provides control facilities for interacting with the target device.

### 7.7.1 Connect to Device

This connects the GUI to the device if it is not connected. Progress messages will appear in the Output window.

### 7.7.2 Disconnect

This disconnects the GUI from the device if it is connected. Progress messages will appear in the output window.

### 7.7.3 Erase FLASH blocks

This launches the 'Erase Blocks' dialog box.

The 'Erase Blocks' dialog box allows the user to specify which blocks are to be erased. The Written column indicates whether data is present in the block.

To start the operation, select the required block names in the list and then click the **Erase** button.

#### **7.7.4 Blank check**

This launches a blank check on the device FLASH and reports back to the Output window.

#### **7.7.5 Upload Image**

This launches the 'Upload Image' dialog box, which enables the uploading of a range of data from the target device. This is intended for access with FLASH memory. The data is placed in the Editor window.

#### **7.7.6 Download Active File**

This downloads the current active file into the target FLASH memory.

#### **7.7.7 FLASH Checksum**

This launches a checksum calculation on the device FLASH and reports back to the Output window. This is an EPROM Style checksum which is a 32 bit addition of the byte values within the range of the flash, padded with 0xFF's to the next EPROM size (eg 64Kb, 128Kb etc).

#### **7.7.8 Go From Address...**

This launches a dialog that allows the user to select an address to execute code from. There is an option to select an indirect address mode, so that the address referenced is itself an address of where the code needs to run from.

#### **7.7.9 Block Locking...**

This launches a dialog that allows the user to setup the locking settings for the project - should the configured device support this feature.

#### **7.7.10 Cancel Operation**

Cancels the current FLASH operation.

#### **7.7.11 Configure Flash Project**

Displays the Flash Properties dialog or launches the Project Wizard if the device is not yet configured.

### **7.8 Help Menu**

The 'Help' menu is used to access additional information on how to use the functionality provided by FDT.

#### **7.8.1 Help Topics**

Launches the Help system for FDT. The 'Help Topics' dialog box for FDT is displayed, enabling help to be accessed on required FDT subjects through several methods.

#### **7.8.2 Technical Support**

This allows the user to report a problem with FDT, or to check the website for the latest updates.



### **7.8.3 About...**

Launches the 'About Flash Development Toolkit' dialog box, through which additional information regarding FDT can be accessed:

- FDT version.
- Copyright information.



# Chapter 8 Windows

This chapter describes each child window type, the features each supports and the options available through their associated pop-up menus.

There are three main windows - Workspace window, Editor window and Output window.

Most windows have local pop-up menus in order to make commonly used features easier to access. These menus are invoked by clicking the right mouse button within the window (or pressing **SHIFT+F10**) and then selecting the required menu option.

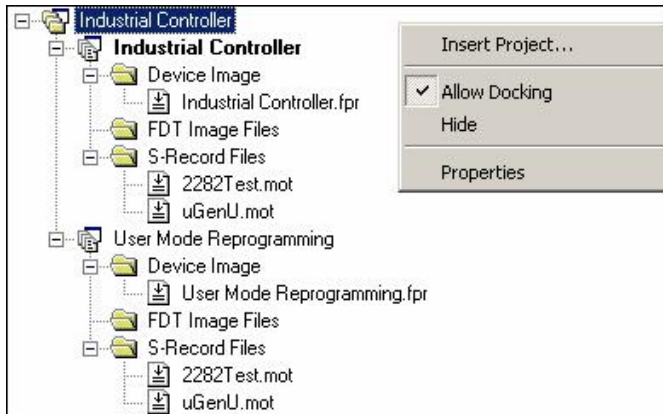
Windows may also be launched from a number of the main menu items.

## 8.1 Workspace window

The Workspace window contains details of the items in the workspace including the workspace name, projects contained in the workspace and for each project their Device Image files and Target files.

### 8.1.1 Workspace

The Workspace is the first item in the window, in the following example this is “Workspace ‘Industrial Controller’”:



**Figure 8-1 Workspace Window**

Invoking certain menu items from a pop-up menu in the Workspace window will launch additional windows.

The Workspace window pop-up menu has the following items:

#### 8.1.1.1. Insert Project...

Invoking this option allows the user to add a project to the Workspace.

### 8.1.1.2. Allow Docking

Invoking this option allows the user to select if the window is docked, or if it can “float” around the main window.

### 8.1.1.3. Hide

Invoking **Hide** conceals the Workspace window. It is restored by [**View->Workspace**].

### 8.1.1.4. Properties

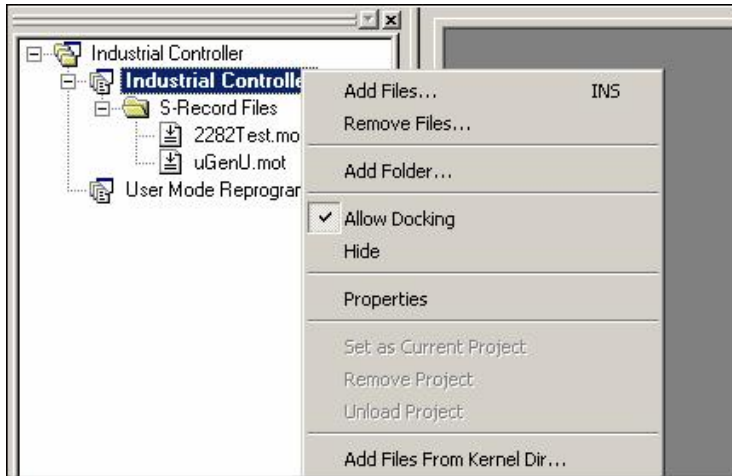
This option displays information about the Workspace including the file path.



**Figure 8-2**      **Workspace Properties**

## 8.1.2 Project

The Project is the next item in the hierarchy below the Workspace. In the following example, the Workspace is called “Industrial Controller” and there are two projects: the first is also called “Industrial Controller” and the second is called “User Mode Reprogramming”.



**Figure 8-3 Workspace Project**

In addition to the Allow Docking and Hidemenu items inherited from the workspace, the pop-up menu for Projects has the following items:

### 8.1.2.1. Add Files...

Launches the ‘Open’ dialog box allowing the user to add S-Record files to the project.

### 8.1.2.2. Remove Files...

Launches a dialog to select the files to remove.

### 8.1.2.3. Add Folder...

Adds a folder to the Project

### 8.1.2.4. Allow Docking

See Section 8.1.1.2

### 8.1.2.5. Hide

See Section 8.1.1.3

### 8.1.2.6. Properties

Displays the project properties dialog. Note that this is not the same as the Flash Properties dialog which can be reached by selecting **Configure Flash Project** from the FDT toolbar or device menu.

### 8.1.2.7. Set as current Project

Sets the highlighted project as the current active project. This option is not available if the highlighted project is already the active project.

### 8.1.2.8. Remove Project

Remove the highlighted project from the Workspace. This option is not available if the highlighted project is the active project.

### 8.1.2.9. Unload Project

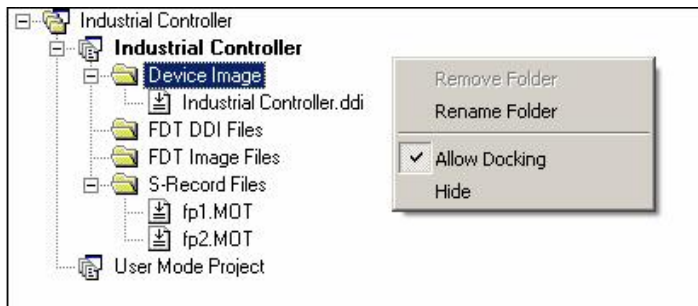
Unloads the highlighted project from the Workspace. This option is not available if the highlighted project is the active project. Note that any changes to the project made since the last save, will be lost when the project is unloaded (FDT will not prompt to save changes before unloading).

### 8.1.2.10. Add Files From Kernel Dir...

This allows the user to select files from the current kernel directory and add those files to the current project.

## 8.1.3 Device Image - Folder

This contains the Device Image file(s) for the project. The device image will be called the same as the project but with a .ddi extension.



**Figure 8-4 Device Image Folder**

In addition to the Allow Docking and Hide menu items inherited from the workspace, Device Image - Sub Folder pop-up menu has the following items:

#### 8.1.3.1. Remove Folder

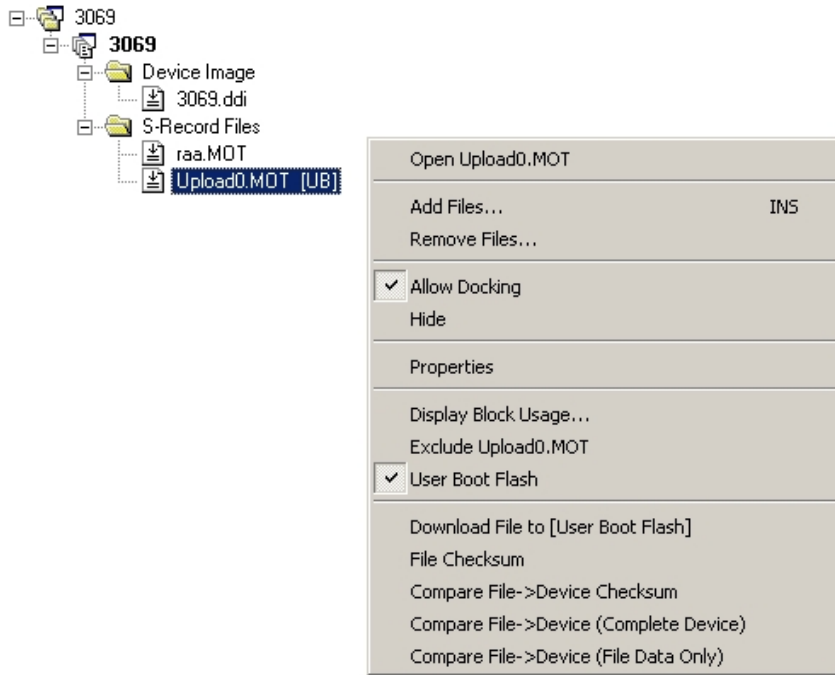
Allows the user to delete folders.

#### 8.1.3.2. Rename Folder

Allows the folder to be renamed.

## 8.2 Device File

The Device File is the item containing the data to Flash into the device.



**Figure 8-5 Device File**

The pop-up menu for Device File has the following items:

### 8.2.1 Open 'filename'

This feature opens the selected file.

### 8.2.2 Add Files...

Allows additional files to be added to the project.

### 8.2.3 Remove Files...

Removes files from the project.

### 8.2.4 Allow Docking

Allows the Workspace window to dock or be floating.

### 8.2.5 Hide

Hides the Workspace window.

## 8.2.6 Properties...

Launches the device file properties dialog box.

## 8.2.7 Display Block usage...

Launches the 'S-Record Properties' dialog box and displays the Block usage tab.

## 8.2.8 Exclude 'filename'

If a file is excluded, it will not be used to form the built image if Rebuild Image is selected. Once selected as excluded, using the right mouse click, it is possible to "Include 'filename'".

## 8.2.9 User Boot Area

User Boot Area : When this option is selected (ticked), the selected file is tagged as being destined for the User Boot Area. If not ticked, it will default to the User Area. This option is only available if the target device supports a User Boot Area

## 8.2.10 Download File to [User Area] /[User Boot Area]

Downloads the selected file to the device FLASH memory. Note that the Flash Area used is determined by whether the file is tagged as a User Boot Area file.

## 8.2.11 File Checksum

The file checksum returns the checksum for the file using the algorithm in the kernel (EPROM style) and also a raw checksum. The raw checksum is a simple 32 bit addition of the byte values within the range of the flash. The EPROM style checksum pads the data with 0xFF's to the next EPROM size (eg 64Kb, 128Kb etc) before calculating the checksum. The result is shown in the output window. An example is shown below:-

```
File Checksum: 0x07F5A651 (EPROM style),  
Raw Checksum: 0x07F5A651 (over address range 0x00000000 -  
0x0007FFFF)
```

## 8.2.12 Compare File->Device Checksum

This command allows the user to compare the selected file checksum to the device checksum. The output window will show the sum check for both the file and the device. The user may then confirm that they are the same.

An example is shown below:-

```
Calculating device checksum  
Flash Checksum: 0x07F5A651 (User Area)  
File Checksum: 0x07F5A651 (EPROM style),  
Raw Checksum: 0x07F5A651 (over address range 0x00000000 - 0x0007FFFF)
```



### 8.2.13 Compare File->Device (Complete Device)

This mode would compare the data in the device flash (starting at the device start address) to the corresponding data in the file. The process would only stop once the entire device had been checked, or differences were found. Areas that are not covered by the file would be assumed to be 0xFF and would therefore require a corresponding blank area on the device to be verified correctly.

### 8.2.14 Compare File->Device (File Data Only)

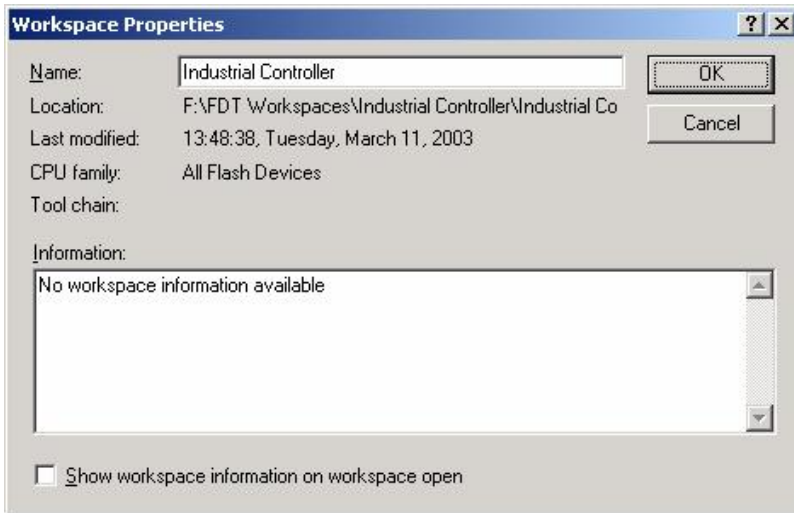
This mode would compare the data in the file to the corresponding data on the device. This is identical in behaviour to the after-write readback verify process. Areas not covered by the file would not be checked. Typically this will be a faster operation, as the entire flash does not have to be uploaded, however, it is therefore a less thorough process.

## 8.3 Workspace Properties

Invoked from the Workspace pop-up menu item **Properties...** See also section 8.1.1.4.

The 'Workspace Properties' dialog box shows details of:

- The Workspace Name
- The filename and path
- The time and date when the workspace was last saved



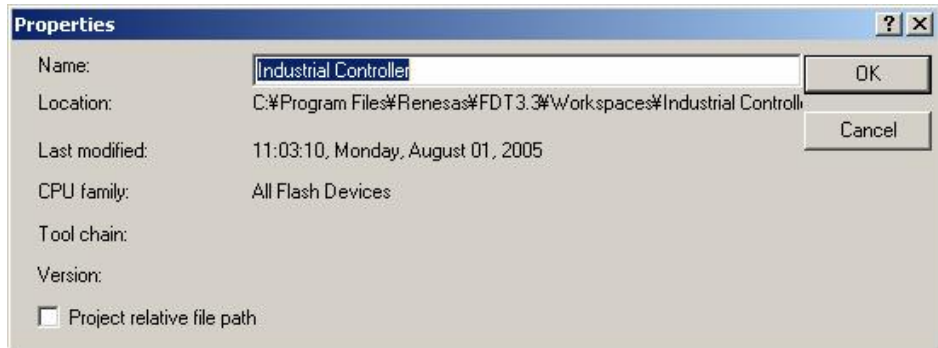
**Figure 8-6** Workspace Properties

## 8.4 Project Properties

Invoked from the Project pop-up menu item **Properties...**

The 'Properties' dialog box shows details of:

- The name of the project
- The location and the filename of the project file
- The last modification date and time



**Figure 8-7 Project Properties**

## 8.5 Configure Flash Project (Flash Properties)

The 'Configure Flash Project' dialog box can be invoked from the 'Configure Flash Project' toolbar button or Device Menu item. It defaults to being docked at the bottom of the FDT window but can be dragged to dock in other locations or to float. To prevent unintentional docking, hold the Ctrl key whilst dragging.

### 8.5.1 KernelTab

This shows the kernel path, frequency, protocol, clock multipliers, and clock mode. Double clicking on these entries when disconnected will invoke the Project Wizard at the appropriate screen.

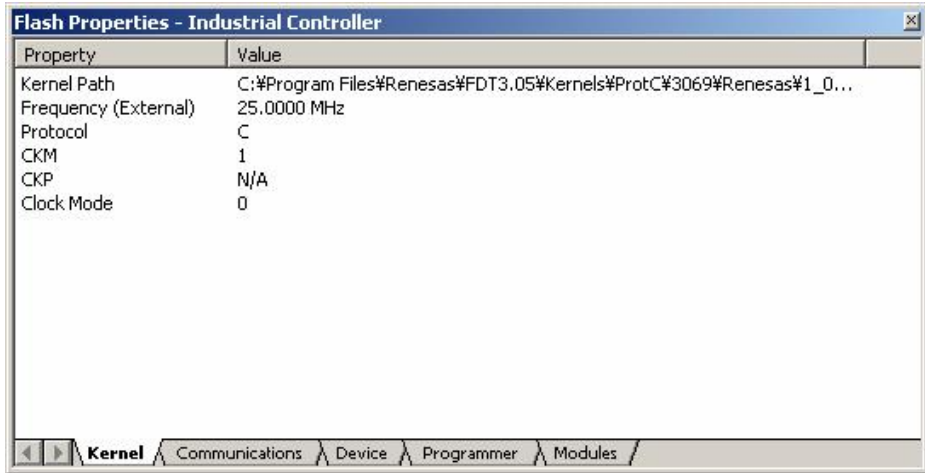


Figure 8-8 Kernel Properties

#### Kernel Path

This shows the kernel path.

#### Frequency

This shows the input clock and whether that clock is external or internal to the target

#### Protocol

This shows the type of communication protocols. See also section 3.2.4

Protocol B: 0.35- $\mu$ m devices

Protocol C: 0.18- $\mu$ m devices

Protocol D: R8C, M16C, M16C/Tiny, M32C or 740 devices

#### CKM

This shows the multiplier of the system clock (master clock) for the input clock.

## CKP

This shows the multiplier of the peripheral clock for the input clock.

## Clock Mode

This shows the clock mode.

### 8.5.2 Communications Tab

This shows the default and current baud rates and the current port. It also allows the user to change the Port and Baud Rate when disconnected.

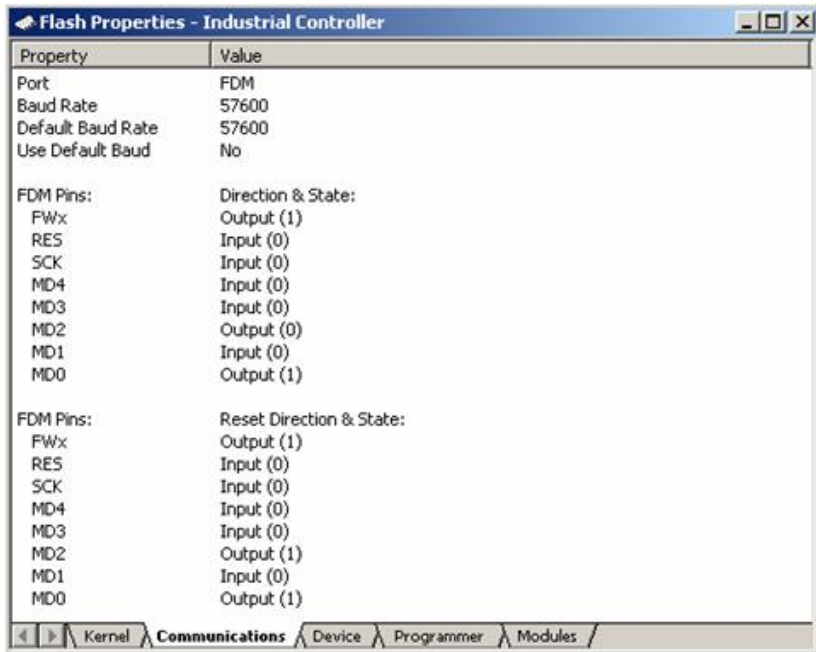


Figure 8-9 Communications Properties

## Port

This drop-down list provides the user with a list of supported COM ports available for connection to the device. This list includes the USB port options.

## Baud Rate

Display the currently selected Baud Rate. Double clicking invokes the Project Wizard to allow editing of this item.

## Default Baud Rate

Display the Default Baud Rate.

## Use default baud

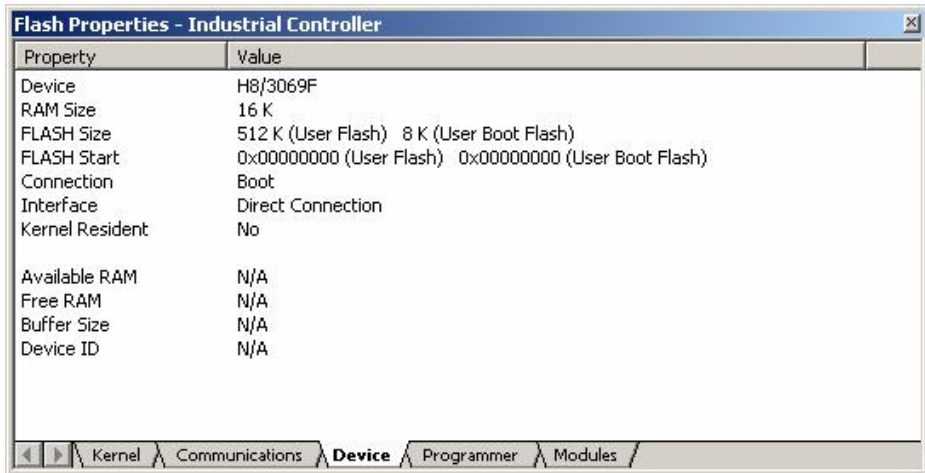
If this is checked, the default Baud Rate will be used. (This option should be set only when the RenesasEvaluation Board or Evaluation Development Kit is used.)

## FDME8Direct Pins

This section displays the direction and state of the controlled mode pins for the Connect and Disconnect with Reset operations.

### 8.5.3 Device Tab

This shows the device information. It also allows the user to select **BOOT Mode** or **USER Program Mode** and the 'Direct Connection' interface. In addition, it can specify whether or not the kernel is in the target device.



Property	Value
Device	H8/3069F
RAM Size	16 K
FLASH Size	512 K (User Flash) 8 K (User Boot Flash)
FLASH Start	0x00000000 (User Flash) 0x00000000 (User Boot Flash)
Connection	Boot
Interface	Direct Connection
Kernel Resident	No
Available RAM	N/A
Free RAM	N/A
Buffer Size	N/A
Device ID	N/A

Figure 8-10 Device Properties

#### Device

Displays the name of the active device. Double clicking invokes the Project Wizard to allow editing of this item.

#### RAM Size, FLASH Size, Flash Start

These are for display only and are not editable.

#### Connection

**Boot Mode:** This connection type specifies that the on-board programming BOOT mode sequence is to be initiated. This will cause the entire FLASH memory to be erased and a kernel loaded.

**USER (Program) Mode:** This connection type specifies that the on-board programming USER Program mode sequence is to be initiated by a previously loaded user program that is used to reprogram the FLASH memory.

## Interface

If the target is connected directly to the host computer, the connection interface should be selected to 'Direct Connection'. Alternative options include FDM and for Protocol A (legacy) kernels, the UPB.

Double clicking invokes the Project Wizard to allow editing of this item.

## Kernel Resident

This specifies that a main kernel is already resident and running on the target device before connection by FDT. Double clicking invokes the Project Wizard to allow editing of this item.

## Available RAM and Free RAM

These are for display only and are not editable.

## Buffer Size

This shows a size of programming the flash memory.

## Device ID

This is not used in communication protocol B, C or D.

### 8.5.4 Programmer Tab

This shows the state of the Function Map and allows the user to select the Device Protection Option of Automatic, Interactive, or None and the message level option of Advanced or Standard.

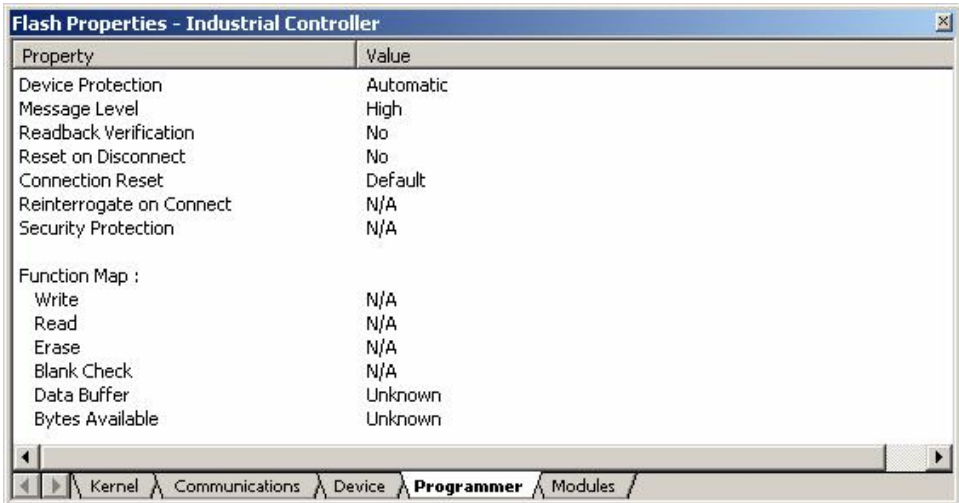


Figure 8-11 Programmer Properties

## **Device Protection**

This allows the user to select either automatic protection (Automatic) or interactive protection (Interactive) to protect the FLASH device from accidental over-erasure and over-programming. Automatic protection will erase blocks from the device prior to programming, as necessary. Interactive protection will ask the user before an erase occurs. In addition, None can be selected to disable device protection. Double click to edit the setting.

Whilst connected to a device FDT keeps track of the state of the FLASH blocks and is able to determine when a programming operation will over-program a region of FLASH memory.

**Note** If device protection is disabled the user is responsible for erasing the device prior to programming.

## **Message level**

The user can select either Standard or Advanced message levels. A Standard message level generates general FDT/target device status messages regarding high-level communications details. An Advanced message level generates more detailed information regarding lower level communications details. Double click to edit the setting.

## **Readback Verification**

After programming, FDT can verify the data was programmed successfully by performing a read-back verification. Use this option to specify whether you would like to perform read-back verification, always, after confirmation, or never. Double click to edit the setting.

## **Reset on Disconnect**

When disconnecting from the device when used in combination with a UPB or FDM, FDT can be used to reset the target hardware. Use this option to specify whether you would like to reset the device, always, after confirmation, or never. Double click to edit the setting.

## **Connection Reset**

This option controls whether FDT should set the mode pins and apply a reset when connecting in User Mode with the FDM. The options are:

Default – Uses the default settings (Set Reset & Mode Pins). This is held in a registry setting.

Reset & Mode – Apply the Reset and set the Mode Pins

Mode Only – Suppress applying the Reset but set the Mode pins. Note that care must be taken to ensure that this will not damage the device or cause unpredictable behaviour. This option is mainly for making the transition between User Mode and User Program Mode

None – Suppresses applying the Reset and setting the Mode pins.

## **Reinterrogate on Disconnect**

This option forces FDT to reinterrogate Generic Boot devices on connection. With this option set to "No" (default), the device settings are saved in an automatically generated fcf file and re-used. With this option set to "Yes", FDT will always reinterrogate the device for its details upon connection; this allows one

project to work with many Generic Boot devices. If this option is set to "Query", FDT will ask you to select whether to reinterrogate the device. Double click to edit the setting.

### Security Protection

This option controls whether FDT should security protect QzROM devices as it disconnects. It echoes the setting on the QzROM toolbar (see 2.1.2). Select "Automatic" to always protect the device at disconnection. Select "Prompt" to query the user what to do and select "None" to never protect the device at disconnect. *NB – It might not be possible to reconnect to a protected device.*

### Function Map

This section is for Protocol A (Legacy) kernels only.

#### 8.5.5 Modules Tab

This shows each Kernel file name. Some entries may be blank because most kernels do not use all of the modules.

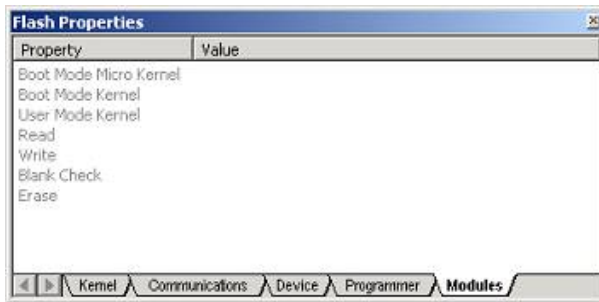


Figure 8-12 Module Properties



## 8.6 S-Record Properties Window

This dialog is invoked by clicking the Properties toolbar button or from the Right-Click, Properties Menu when the file is open and selected in the Hex Editor window.

### 8.6.1 Block Usage Tab

This tab shows the starting, finishing address and sizes of the data blocks contained in the active file tab. If the file is open in the editor window, double clicking on a range highlights the selected data.

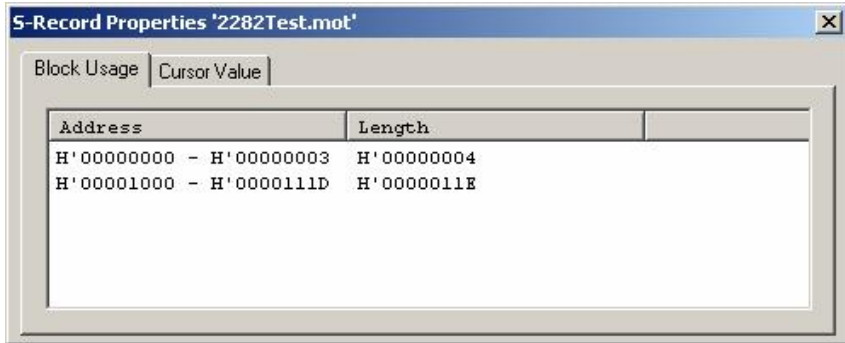


Figure 8-13 S Record Properties – Block Usage

### 8.6.2 Cursor Value

This tab shows various representations of the data at the cursor position. The display depends upon whether the display unit is set to Byte, Word or DWord. Note that it assumes data alignment of the display unit size so for example, the same value is displayed wherever the cursor is positioned within a DWord.

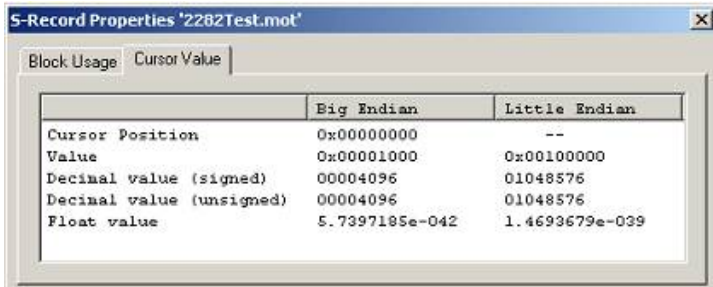


Figure 8-14 S Record Properties – Cursor Value

### 8.6.3 Selection Value

This tab shows various representations of the data that is currently selected. The display depends upon whether the selection is a Byte, Word or DWord.

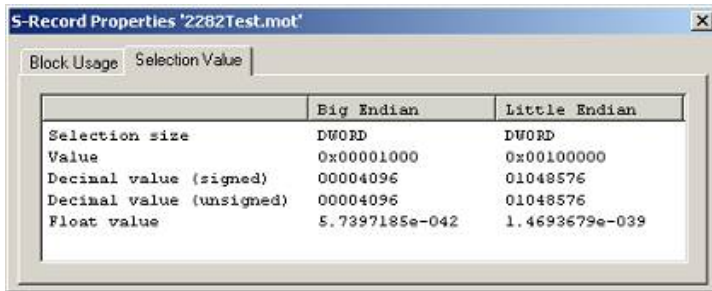


Figure 8-15 S Record Properties – Selection Value

## 8.7 Output Window

The Output window is one of the main windows contained in the FDT GUI.



Figure 8-16 Output Window

The window has available pop-up menu that contains the following items:

### 8.7.1 Copy

Copies any selected text to the clipboard

### 8.7.2 Clear Window

This will clear the Output window.

### 8.7.3 Save

Allows the contents of the window to be saved to a text file. A Save Dialog will appear to specify the intended file location.

## 8.7.4 Allow Docking

With this option checked the Output window is capable of being docked within the FDT application window. With the option unchecked the Output window is a floating window.

## 8.7.5 Hide

Invoking **Hide** conceals the output window. It is restored by [**View->Output**].

*Note – the output window will not log messages indefinitely. The current default is 500 lines of text, after which older messages will be purged.*

## 8.8 Hex Editor Window

The Hex Editor window is one of the main windows contained in the FDT GUI. It is used to display file contents or target uploads within FDT. There are several ways of opening a file within the editor:

- double-click on a project file from the Workspace view
- double-click on an FDT file type from Windows® Explorer
- drag and drop a file from Windows® Explorer onto the editor space
- choose 'Open an S-record...' from the File menu or toolbar icon
- upload data from a target device

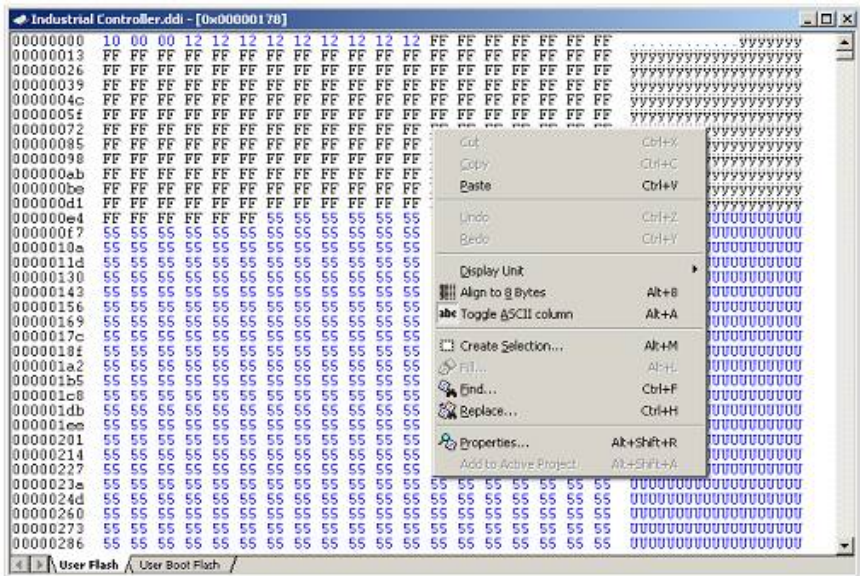


Figure 8-17 Hex Editor Window

The window will have one or more tabs along the bottom of the view, each of which will correspond to a flash area on the target device for the project. Each tab will be set to the address range of that flash area and will show the data that corresponds to that area. DDI files and uploads via blocks are capable of adding into

multiple tabs simultaneously. Normal S-record and binary files (and uploads via address) will only have data for one flash area at a time, which will leave the other tabs empty (although they can still be edited manually).

To close an open file, click the cross in the top right corner of that file. If the file is unsaved, FDT will prompt to ask if you want to save the open file unless the shift key is held down whilst the cross is clicked. After closing and saving, FDT will ask if you wish to add the file to the current project if it is not included already.

When saving a .ddi file as a different file type, e.g. S-Record, then FDT is not able to output the data from each of the DDI tabs into a single file. In this case it will issue a warning that only the data from the active tab will be saved into the file. The active tab is the currently viewable tab. NOTE - This may result in data loss when compared to the original DDI file.

*Also note that FDT will default to saving text based file formats (e.g. S-Record or DDI) as uppercase, even if the original file was lowercase. This is to improve compatibility with other Renesas tools. This behaviour is new from 3.06 and can be modified by adding an entry into the FDT.ini file (found in the root install directory):*

```
[ECXSRecordView]
LowerCaseSRecordSave=1
```

This will make FDT case aware and ensure it saves lowercase files in their original format.

The above window has a right-click pop-up menu that contains the following items:

### **8.8.1 Cut**

This will remove the contents of the highlighted block from the window and place it on the clipboard in the standard Windows<sup>®</sup> manner. This is only available if a block is highlighted.

### **8.8.2 Copy**

This will copy the contents of the highlighted block to the clipboard in the standard Windows<sup>®</sup> manner. This is only available if a block is highlighted.

### **8.8.3 Paste**

This will copy the contents of the Windows<sup>®</sup> clipboard into the child window at the current cursor position.

### **8.8.4 Undo**

Reverses the last editing operation on the selected data.

### **8.8.5 Redo**

Reverses the last undo operation.

### **8.8.6 Display Unit**

Invokes the cascaded menu as follows:

**Byte** - view the data as 8-bit bytes.

**Word** - view the data as 16-bit words.

**DWord** - view the data as 32-bit double words.

### **8.8.7 Align to 8 Bytes**

Data is displayed on each line as 8 bytes. The number of bytes that can be accommodated on each line is dependent upon the size of the window.

### **8.8.8 Toggle ASCII Column**

Allows the ASCII column to be removed or displayed.

### **8.8.9 Create Selection...**

Launches the 'Create Selection' dialog box allowing the user to create a selection area by entering its start address, end address and length. The selected area can be used in conjunction with the Clipboard, Fill or Search and Replace.

### **8.8.10 Fill...**

Launches the 'Fill' dialog box for the current active file to write the specified data to the selected area. When the 'ASCII Fill' check box is selected, the data to be written can be specified with ASCII characters.

### **8.8.11 Find...**

This will launch the 'Find' dialog box allowing the user to enter either hexadecimal or ASCII data. The scope of the search is limited to the data within the active file in the Editor window. If a match is found, the Editor window is updated to display the found data.

### **8.8.12 Replace...**

This will launch the 'Replace' dialog box allowing the user to enter either the hexadecimal or ASCII data to be found and the replacement data, in the same format. The scope of the search is limited to the data within the active file in the Editor window. If a match is found, the Editor window is updated to display the found data, click on the now enabled **Replace** button to replace the data.

### **8.8.13 Properties...**

Launches the 'S-Record Properties' dialog box.

### **8.8.14 Add to Active Project**

Adds the file to the active project. This menu item will be disabled if the file is already added to the project.

### 8.8.15 Data EntryNote

When entering data in the 'Find', 'Replace' or 'Fill' dialog boxes only hexadecimal or ASCII values can be used, based on the ASCII Search check box. When entering hexadecimal values a preceding '0x' is fixed in the entry text box, so hexadecimal values are appended.

### 8.8.16 Goto Address

Double clicking on the address column displays the Goto Address dialog. This allows entry of a hex address to set the hex editor cursor position to. Note that this is entirely different to the similarly named kernel command "Go From Address".

## 8.9 Customise - Toolbars

Invoked by [Tools->Customise...] in the 'Customise' dialog box.

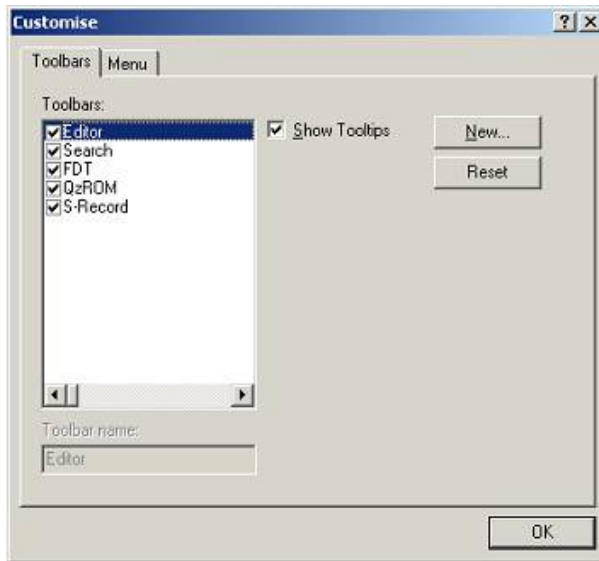


Figure 8-18 Customise Toolbars Dialog

### 8.9.1 Show Tooltips

A check mark in the **Show Tooltips** box indicates that the action of the button will be displayed when the mouse arrow is pointing to the button. If the box is unchecked, no action is displayed.

### 8.9.2 New...

The **New...** button launches the 'Toolbar Name' dialog box which allows the user to name and generate a new toolbar. After entering the new name and clicking on **OK**, the new name will be added to the Toolbars list and an empty toolbar is displayed on the interface.

Selecting the Commands tab allows the user to select and add buttons to the new toolbar. The user can place the new toolbar anywhere on the GUI by dragging it with the mouse.

### **8.9.3 Reset**

The **R**eset button resets the toolbar to the default.

### **8.9.4 Toolbar name**

Reflects the toolbar selected in the list.

### **8.9.5 OK**

Clicking on **O**K, in the 'Toolbar Name' dialog box adds the new name to the Toolbars list on the Toolbars window and invokes a blank toolbar on the GUI.

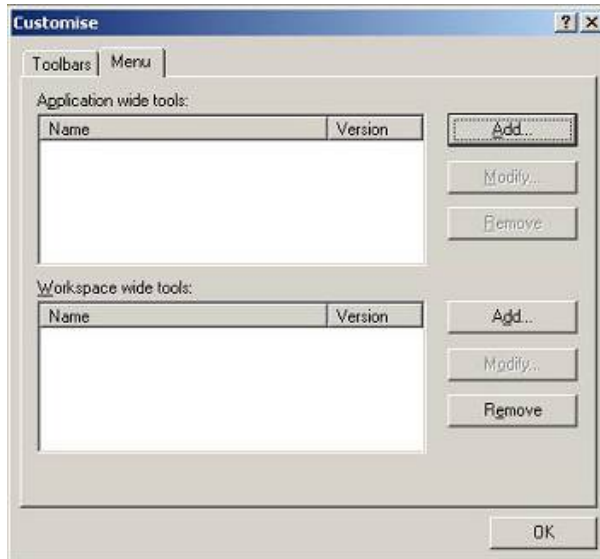
Clicking on **O**K in the 'Customise' dialog box saves any changes made and closes the dialog box.

### **8.9.6 Delete**

If a new toolbar has been added to the Toolbars list and it is selected, the **R**eset button is renamed to **D**elete. Clicking on this button will remove the new toolbar from the list and from the GUI.

## 8.10 Customise – Menu

Invoked by [**T**ools->**C**ustomise...] and then select the Menu tab in the ‘Customise’ dialog box.

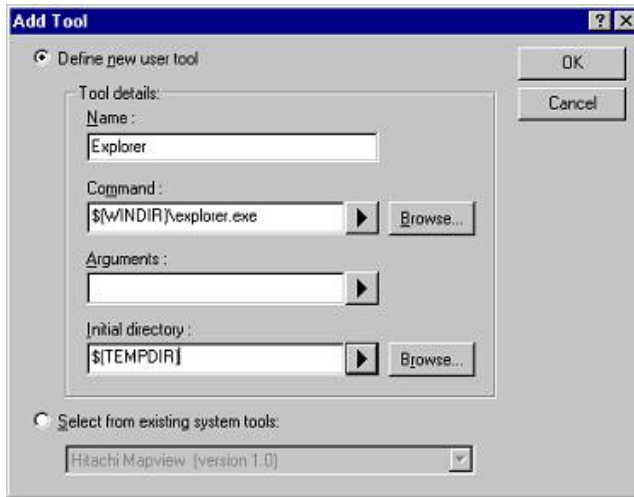


**Figure 8-19** Customise Menu Dialog

➡ To add a new menu option:

1. Select [**T**ools->**C**ustomise...]. The dialog shown above will be displayed. Select the “Menu” tab. The first thing for you to decide is whether you are adding a global application wide tool (“Application wide tools:”), which will be available to all of your workspaces. Or whether you wish to add a workspace wide tool (“Workspace wide tools:”), which is only valid for the current workspace. Once you have made the choice choose the relevant section of the dialog.
2. Click the “Add...” button. If you would like to add an existing system tool to the menu then select the “Select from existing system tools” radio button, choose the tool from the drop-down list and then click “OK”. Alternatively, if you would like to add a tool of your own then follow the remaining steps.
3. Enter the name of the tool into the “Name” field.
4. Enter the command, excluding arguments, into the “Command” field.
5. Enter any arguments that you would like to pass to the command into the “Arguments” field.
6. Enter an initial directory in which you would like the tool to run, into the “Initial directory” field.
7. Click “OK” to add the menu option to the “Tools” menu.



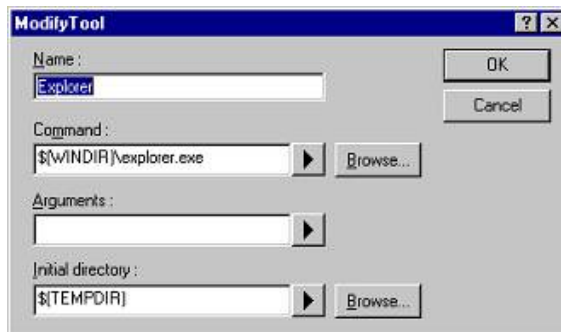


**Figure 8-20 Add Tool Dialog**

New menu options are added to the bottom of the list (i.e. bottom of the tools menu) by default. The order of menu options in the **[Tools]** menu can also be modified.

☞ To modify a menu option:

1. Select **[Tools->Customise...]**. The dialog shown below will be displayed. Select the “Menu” tab.
2. Select the menu option that you would like to modify and then click the “Modify...” button.
3. Make the desired changes on the “Modify Tool” dialog and then click “OK”.



**Figure 8-21 Modify Tool Dialog**

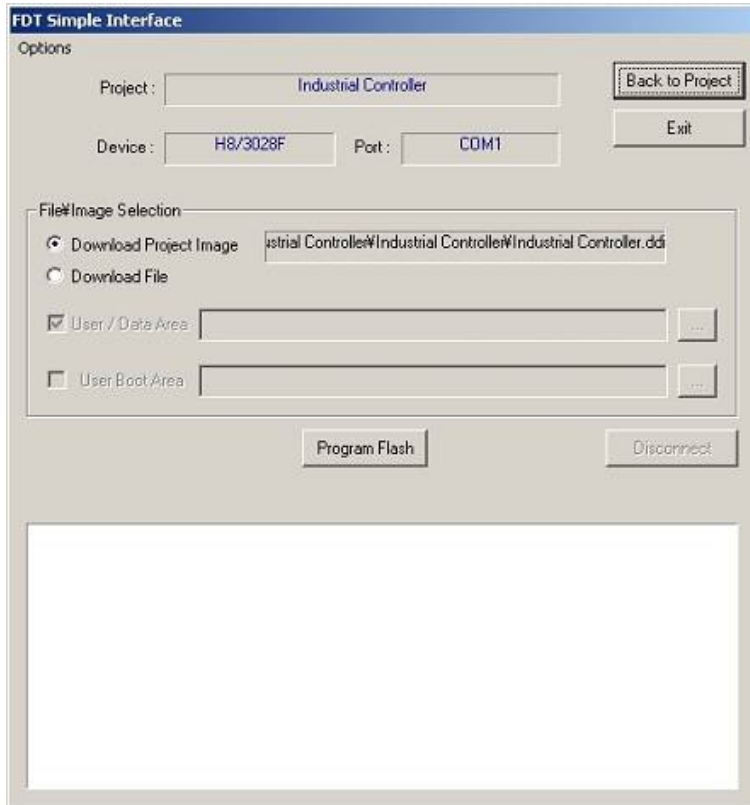
☞ To remove a menu option:

1. Select **[Tools->Customise...]**. The dialog shown in Figure 6-19 will be displayed. Select the “Menu” tab.
2. Select the menu option that you would like to remove and then click the “Remove” button.



## Chapter 9 Simple Interface Mode

Invoked by [**T**ools->**S**imple Interface...]. This is used to simplify the look and feel of FDT once a project is configured. See Chapter 5 Access Rights for details about using this in conjunction with user names and passwords. This chapter describes the full Simple Interface Mode; see Chapter 10 for details of Basic Simple Interface Mode.



**Figure 9-1 FDT Simple Interface**

## **9.1 Options Menu**

### **9.1.1 Login...**

This allows changing of which user is logged in to FDT. The hotkey Ctrl+Shift+U also invokes the Login dialog.

### **9.1.2 Batch Erase Mode...**

This will change the dialog to be erase based, rather than programming based. The 'Program Flash' button will change to 'Erase', the file/image selection buttons will be disabled and the programming options will also be disabled. Clicking 'Erase' will erase the whole device (it is not possible to specify the area to erase).

### **9.1.3 AutoDisconnect**

This forces FDT to automatically disconnect after programming (or erase if in Batch Erase Mode), so that the user does not need to press the Disconnect button.

### **9.1.4 Readback Verify**

This option controls whether FDT automatically performs a read and compare operation after programming. The default setting is taken from the project setting ("Readback Verification" on the Programmer tab of the Project Properties window), but the user can override the default by selecting this menu item.

### **9.1.5 Request Checksum**

This option controls whether FDT automatically performs a checksum request operation after programming. The setting is remembered between sessions.

### **9.1.6 Erase Device Before Program**

This option controls whether FDT automatically performs an erase operation before programming the device. This will erase all blocks in all memory areas prior to programming. The setting is remembered between sessions.

### **9.1.7 Security Protection**

This option controls whether FDT should security protect QzROM devices as it disconnects. Select "Automatic" to always protect the device at disconnection. Select "Prompt" to query the user what to do and select "None" to never protect the device at disconnect. *NB – a protected device cannot be reconnected to.*

### **9.1.8 Field Programming->Run Published File...**

Unpackage an .fpf file and use it in Simple Interface Mode.

### **9.1.9 About...**

Shows the About box.

## 9.2 Dialog Controls

### 9.2.1 Back to Project

This returns to the full FDT view. This will be disabled if the user currently logged in to FDT does not have the 'FDT:Project Edit' access. [see Chapter 5 for details]

### 9.2.2 Exit

Clicking this will exit FDT and FDT will start up in the Simple Interface Mode next time it is loaded. This will be disabled if the user currently logged in to FDT does not have the 'SimpleInterface:Exit' access. [see Chapter 5 for details]

### 9.2.3 Download Project image / Download File radio buttons

These are mutually exclusive and allow the user to specify whether to download the Project Image or another file. These will be disabled if the user currently logged in to FDT does not have the 'SimpleInterface:Download File' or 'SimpleInterface:Download Project Image' access. [see Chapter 5 for details]

### 9.2.4 Project Image / Download File Area

If the Download Project Image radio button is selected, then FDT will display the image filename that will be downloaded.

If the Download File radio button is selected, for devices with a User Boot Area, the user can choose whether to download the file to the **User Area** or the **User Boot Area**.

### 9.2.5 Program Flash \ Erase

Clicking on **Program Flash** allows FDT to download the specified file or image to the FLASH memory. Additionally, if Readback Verify is switched off, the Checksum will also be calculated and output to the screen. If in Batch Erase mode this button will switch to 'Erase' and will perform an entire device erase.

### 9.2.6 Disconnect

Clicking on **Disconnect** forces FDT to disconnect.

### 9.2.7 File Selection

The filename can be specified in the File Selection section (by typing or browsing), for each area enabled by the Download File Area checkboxes



## Chapter 10 Basic Simple Interface Mode

Invoked from the start menu “Flash Development Toolkit3.07 Basic”. This is used to simplify the look and feel of FDT and does not need a Workspace and Project. See Chapter 5 Access Rights for details about using this in conjunction with user names and passwords. This chapter describes the Basic Simple Interface Mode; see Chapter 9 for details of the full Simple Interface Mode.

Basic Simple Interface Mode will restore any previous settings on start up. To change the device, kernel or port settings, the **Options**->**New Settings...** menu will invoke the Wizard to collect the new details.

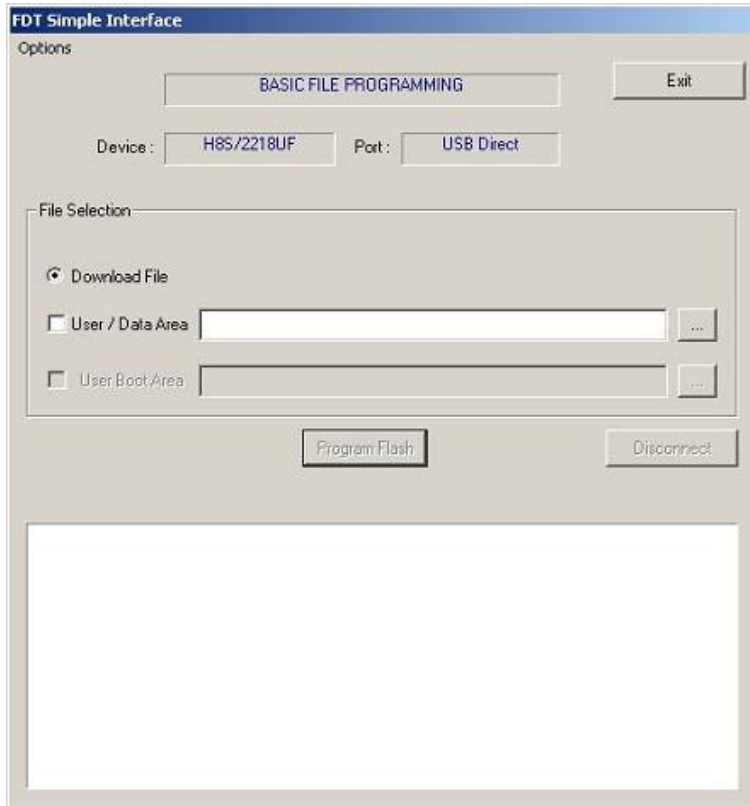


Figure 10-1 FDT Basic File Programming

## **10.1 Options Menu**

### **10.1.1 Login...**

This allows changing of which user is logged in to FDT. The hotkey Ctrl+Shift+U also invokes the Login dialog.

### **10.1.2 Batch Erase Mode...**

This will change the dialog to be erase based, rather than programming based. The 'Program Flash' button will change to 'Erase', the file/image selection buttons will be disabled and the programming options will also be disabled. Clicking 'Erase' will erase the whole device (it is not possible to specify the area to erase).

### **10.1.3 AutoDisconnect**

This forces FDT to automatically disconnect after programming (or erase if in Batch Erase Mode), so that the user does not need to press the Disconnect button.

### **10.1.4 Readback Verify**

This option controls whether FDT automatically performs a read and compare operation after programming. The setting is remembered between sessions.

### **10.1.5 Request Checksum**

This option controls whether FDT automatically performs a checksum request operation after programming. The setting is remembered between sessions.

### **10.1.6 Erase Device Before Program**

This option controls whether FDT automatically performs an erase operation before programming the device. This will erase all blocks in all memory areas prior to programming. The setting is remembered between sessions.

### **10.1.7 Security Protection**

This option controls whether FDT should security protect QzROM devices as it disconnects. Select "Automatic" to always protect the device at disconnection. Select "Prompt" to query the user what to do and select "None" to never protect the device at disconnect. *NB – a protected device cannot be reconnected to.*

### **10.1.8 New Settings...**

Run the Wizard to allow entry of the new settings.

### **10.1.9 About...**

Shows the About box.



## 10.2 Dialog Controls

### 10.2.1 Exit

Clicking this will exit FDT and FDT will start up in the Simple Interface Mode next time it is loaded. This will be disabled if the user currently logged in to FDT does not have the 'SimpleInterface:Exit' access. [see Chapter 5 for details]

### 10.2.2 Download File radio button

This will be disabled if the user currently logged in to FDT does not have the 'SimpleInterface:Download File' access. [see Chapter 5 for details]

### 10.2.3 Download File Area

For devices with a User Boot Area, the user can choose whether to download the file to the User Area or the User Boot Area.

For devices with a Data Area, the user can choose whether to download the file to the User Area or the Data Area.

### 10.2.4 Program Flash \ Erase

Clicking on **Program Flash** allows FDT to download the specified file or image to the FLASH memory. Additionally, if Readback Verify is switched off, the Checksum will also be calculated and output to the screen. If in Batch Erase mode this button will switch to 'Erase' and will perform an entire device erase.

### 10.2.5 Disconnect

Clicking on **Disconnect** forces FDT to disconnect.

### 10.2.6 File Selection

The filename can be specified in the File Selection section (by typing or browsing), for each area enabled by the Download File Area checkboxes



# Chapter 11 Field Programming

This is a feature designed to assist FDT users to send their FDT projects to other FDT users. Once a project is configured correctly and the project images are built, FDT can package all required files (Kernel and data files) into a single .fpf file. This can be sent to another FDT user (using the same version of FDT), who can simply double click on the .fpf file to start FDT and unpack the project files. FDT will unpack the files and update the references to the file paths for the new location, and allow programming of the Project Images using FDT Simple Interface Mode (subject to the user having the correct access rights).

To publish (create) a .fpf packages file, Select the **Project->Field Programming...->Publish...** menu item when in the full FDT Workspace view. Because a project image is a prerequisite for publishing a project, if the project does not have a Project Image built already, FDT will ask you if you want to build one now. FDT will then prompt you to select an output directory for the published .fpf file.

To run a published file, select the '**Project->Field Programming...->Run Published File...** menu item from the full FDT Workspace view, or the **Options-> Field Programming->Run Published File...** menu item from the Simple Interface Mode screen. Also, double clicking on the .fpf file from Windows® will start FDT in Simple Interface Mode using the selected .fpf file.

FDT will unpack the .fpf files into the **Published Projects** subdirectory of FDT.

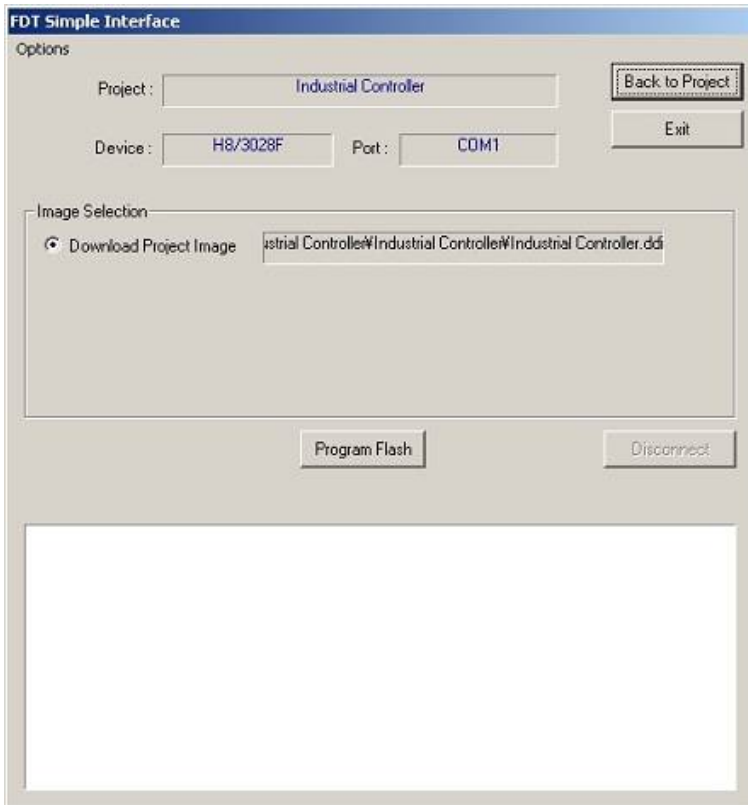


Figure 11-1 FDT Field Programming Interface

## **11.1 Options Menu**

### **11.1.1 Login...**

This allows changing of which user is logged in to FDT. The hotkey Ctrl+Shift+U also invokes the Login dialog.

### **11.1.2 Batch Erase Mode...**

This will change the dialog to be erase based, rather than programming based. The 'Program Flash' button will change to 'Erase', the file/image selection buttons will be disabled and the programming options will also be disabled. Clicking 'Erase' will erase the whole device (it is not possible to specify the area to erase).

### **11.1.3 AutoDisconnect**

This forces FDT to automatically disconnect after programming (or erase if in Batch Erase Mode), so that the user does not need to press the Disconnect button.

### **11.1.4 Readback Verify**

This option controls whether FDT automatically performs a read and compare operation after programming. The setting is remembered between sessions.

### **11.1.5 Request Checksum**

This option controls whether FDT automatically performs a checksum request operation after programming. The setting is remembered between sessions.

### **11.1.6 Erase Device Before Program**

This option controls whether FDT automatically performs an erase operation before programming the device. This will erase all blocks in all memory areas prior to programming. The setting is remembered between sessions.

### **11.1.7 Security Protection**

This option controls whether FDT should security protect QzROM devices as it disconnects. Select "Automatic" to always protect the device at disconnection. Select "Prompt" to query the user what to do and select "None" to never protect the device at disconnect. *NB – a protected device cannot be reconnected to.*

### **11.1.8 Field Programming->Run Published File...**

Unpackage another .fpf file and use it in Simple Interface Mode.

### **11.1.9 About...**

Shows the About box.

## **11.2 Dialog Controls**

### **11.2.1 Back to Project**

This returns to the full FDT Workspace view. This will be disabled if the user currently logged in to FDT does not have the 'FDT:Project Edit' access. [see Chapter 5 for details]

### **11.2.2 Exit**

Clicking this will exit FDT and FDT will start up in the Simple Interface Mode next time it is loaded. This will be disabled if the user currently logged in to FDT does not have the 'SimpleInterface:Exit' access. [see Chapter 5 for details]

### **11.2.3 Download Project image radio button**

This will be disabled if the user currently logged in to FDT does not have the SimpleInterface:Download Project Image' access. [see Chapter 5 for details]

### **11.2.4 Program Flash \ Erase**

Clicking on **Program Flash** allows FDT to download the specified file or image to the FLASH memory. Additionally, if Readback Verify is switched off, the Checksum will also be calculated and output to the screen. If in Batch Erase mode this button will switch to 'Erase' and will perform an entire device erase.

### **11.2.5 Disconnect**

Clicking on **Disconnect** forces FDT to disconnect.



## Chapter 12 E8 Update

When connecting to an E8 interface adaptor board, FDT will check that the E8 firmware (adaptor software) is the correct version and correct type for supporting the connection to the selected device. If there is a mismatch then FDT will offer to update the E8 firmware in order that the connection can be continued. Also if the E8 Adaptor checksum is incorrect, an update will be required. Note that in some cases, an update to an earlier version number may be required if the current version does not support the selected device. It is very important that the E8 is not unplugged during the update process because this could result in the E8 being left in an erased state.

NOTE - The E8 has two firmware areas, a 'Level 0' and a 'Level 1' (the Bootstub and Adaptor Software respectively). Typically FDT will only update the Level 1 area but in some cases a Level 0 update is also required. This is usually seen when connecting with E8Direct (see Chapter 13 - E8Direct) for the first time and will result in a slightly different update process, shown below for comparison:

### Level 0 and Level 1 update:

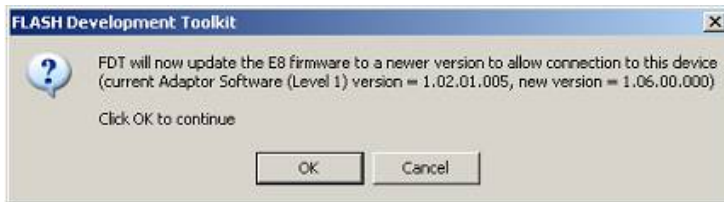
```
E8 target power has been set to ON at 5.0V
Connecting to device 'R5F21174' on 'E8'
Configuration:
'BOOT Mode' connection - using emulated interface
Opening port 'E8' ...
Checking for E8 update...
Adaptor checksum OK
Processing Data file : 'c:\FDT\ADPFiles\E8-Rel-Img.mot'
Mismatched Bootstub (Level 0) version number found (current E8 version =
1.00.01.000, FDT holds version 2.00.00.000, FDT requires version 2.00.XX.XXX)
```



```
Update selected
Adaptor update started (do not unplug)...
Processing Data file : 'c:\FDT\ADPFiles\MonpMain.mot'
  Adaptor update program downloaded...
Processing Data file : 'c:\FDT\ADPFiles\AdpErase0.mot'
  Adaptor erased...
Processing Data file : 'c:\FDT\ADPFiles\AdpWrite0.mot'
Updating...
Adaptor updated successfully
<Connection continues...>
```

## Level 1 only update:

```
E8 target power has been set to ON at 5.0V
Connecting to device 'R5F21174' on 'E8'
Configuration:
'BOOT Mode' connection - using emulated interface
Opening port 'E8' ...
Checking for E8 update...
Adaptor checksum OK
Processing Data file : 'c:\FDT\ADPPFiles\E8-Rel-Img.mot'
Adaptor version number OK
  (current E8 Bootstub (Level 0) version = 2.00.00.000, FDT holds version
  2.00.00.000, FDT requires version 2.00.XX.XXX)
Adaptor version type OK
  (current type = E8EMULATOR)
Processing Data file : 'c:\FDT\ADPPFiles\E8-Rel-Img.mot'
Mismatched Adaptor Software (Level 1) version number found (current E8 version
= 1.02.01.005, FDT holds version 1.06.00.000, FDT requires version
1.06.XX.XXX)
```



```
Update selected
Adaptor update started (do not unplug)...
Processing Data file : 'c:\FDT\ADPPFiles\MonpMain.mot'
  Adaptor update program downloaded...
Processing Data file : 'c:\FDT\ADPPFiles\AdpErase.mot'
  Adaptor erased...
Processing Data file : 'c:\FDT\ADPPFiles\AdpWrite.mot'
  Updating...
Adaptor updated successfully
<Connection continues...>
```



# Chapter 13 E8Direct

The E8 is an existing USB interface board, which has been used to implement the communications protocol for programming R8C, and M16C devices. Originally this did not support a low level interface allowing its use without the E8 having knowledge of the device protocol. From FDT 3.05 onwards a new E8 firmware is supported that offers “E8Direct” - a low level interface using the existing E8 hardware, designed to be as similar as possible to the FDM. Because of some hardware differences, the command protocol is not identical, but it has been designed to be backwardly compatible.

**WARNING: E8Direct is intended for use with Renesas Starter Kits. Use of E8Direct with other boards is not supported and is untested - damage may occur to the E8 and/or the target board if incorrect settings are used. It is important to always ensure that the target board power is switched off when plugging the E8 connector into the target board**

## 13.1 Firmware and Device ID

When plugged in initially the E8 will report itself as a Renesas E-Series Device.

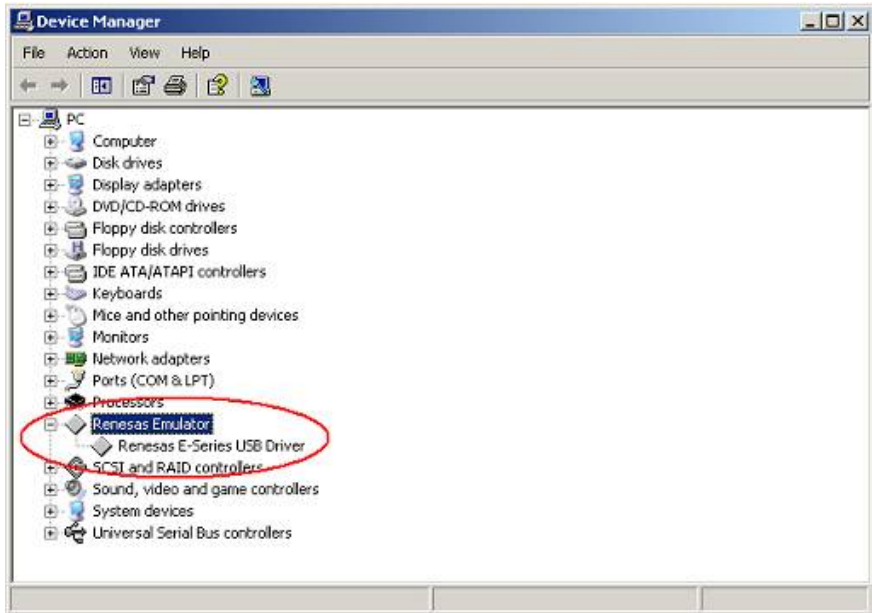


Figure 13-1 Initial E8 enumeration

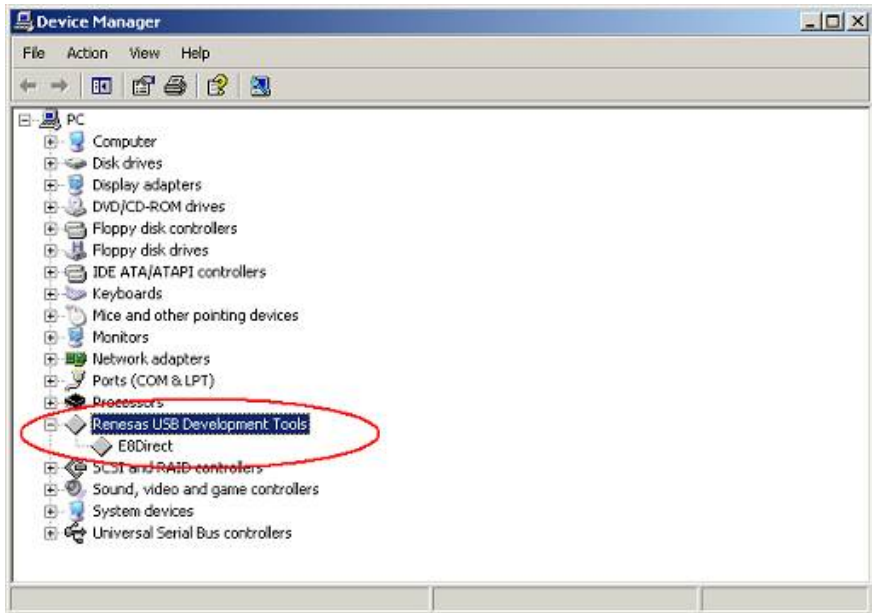
To use the E8Direct functionality it must be restarted, a task done automatically by FDT during a connection operation (which may involve a firmware upgrade, see Chapter 12 - E8 Update). The Output window will contain the following messages

Preparing for re-enumeration of E8 to E8Direct...

Sending re-enumeration command...

**Request sent successfully, closing E8 Comms...**

At this point the E8 will re-identify itself to Windows as an E8Direct device. NOTE – it will remain as an E8Direct device until next unplugged or the machine is powered down.



**Figure 13-2 E8Direct enumeration**

## 13.2 Hardware Differences

The FDM had no capability of supplying the target power whereas the E8Direct can supply 3.3V or 5V and the FDM-E8Direct pin usage is different. This means that it is not possible to simply swap an FDM for an E8. The general purpose I/O pins on the FDM were labelled as MD0 to MD4 and FWE, and they had a direct mapping with the mode pins with matching names on the target devices. Because the number of available general purpose I/O pins is reduced on the E8, and the locations on the 14-pin header are different, we can no longer have a direct mapping between device mode pin names and E8 I/O pins, so we have chosen to label them as 'A', 'B', 'C' and 'D'. The pin settings required for E8Direct will have to be board specific rather than device specific.

Pin No.	FDM Pin Name		E8 Direct Pin Name
	14-pin connector	20-pin connector (Japan only)	
1	SCK	/RES	Output D
2	GND	GND	GND
3	FWx	FWx	Output C
4	MD0	GND	Output A
5	Txd (Input from Txd of Target device to Rxd of 2148AF on FDM)	MD0	Target Txd (Input from Txd of Target device to Rxd0 of 2215UF on E8)
6	MD1	GND	GND (Connected to GND on E8)
7	MD4	MD1	Output B
8	UVCC Input	GND	UVCC (can supply 3.3 or 5V or detect if target is powered)
9	PVCC Input	MD2	BUSY (Not used for E8Direct)
10	MD2	GND	GND (Connected to GND on E8)
11	Rxd (Output to Rxd of Target device from Txd0 of 2148AF on FDM)	MD3	Target Rxd (Output to Rxd of Target device from Txd0 of 2215UF on E8)
12	MD3	GND	GND (Connected to GND on E8)
13	RESET	MD4	RESET
14	UCONNECT	GND	UCONNECT (Connect to GND on Target board)
15	-	RXD (TXD for the user system)	-
16	-	GND	-
17	-	TXD (RXD for the user system)	-
18	-	VIN (Vcc or PVcc)	-
19	-	NC	-
20	-	VIN (PVcc)	-

The FDM used a H8S/2148AF with an 18.432MHz crystal, but the E8 uses an H8S/2215UF with a 16MHz crystal.

The numbering system above applies to the 14-pin FDM and E8. This numbering system does not apply to the 20-pin FDM connector.



# Chapter 14 QzROM Programming

QzROM is a new type of programmable flash memory, found in some 4 and 8 bit MCUs. It is based most closely around the Protocol D style device but has several important differences:

- One time programming - each byte of QzROM can only be written once.
- QzROM cannot be erased - there is no erase command.
- Security Protection - QzROM has security protection. After protection, attempts to start the target MCU for connection can result in an error status, i.e. further reads from (or writes to) the device are not possible.

Blank check - QzROM has a Blank check command whereas other Protocol D devices don't. If Device Protection is set to "Automatic" or "Interactive" FDT will use this to check the device write state before programming.

FDT is aware of QzROM based on the device that is configured for the project.

## 14.1 Reserved Areas

QzROM introduces the concept of 'reserved areas' into FDT, for example the M37545G4 has an 8-byte reserved area within the ROM. A reserved area is a defined area of Flash that FDT will not read, write or blank check. Within the Hex Editor these reserved areas are shown as a series of "\*" characters.

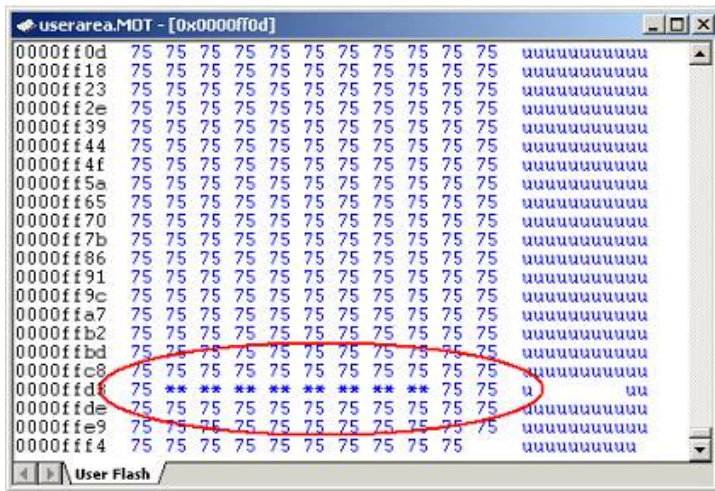


Figure 14-1 Hex Editor view of Reserved Area

Any attempt to type or copy data, etc. into these areas will be ignored, as will attempts to copy data out. Files saved from this view will have no data for these areas (internally FDT treats this data as 0xFF despite what may be read in).

## 14.2 Security Protection

QzROM also introduces the concept of Security Protection into FDT, which is optionally set as FDT disconnects (Note – once set, this cannot be undone).

The majority of devices have two levels of protection, Full (which prevents any later reconnection to the device) or None (which allows reconnection normally). Newer devices have more levels - allowing a partial protection of the device and yet still allowing a later connection (but with the protected data region masked).

There are three options for the setting of security protection:

1. **Automatic** – FDT *will* protect the device as it disconnects. The level of protection will be determined when Automatic is initially selected, typically 'Full'. If the device offers multiple levels of protection then a dialog will appear to allow the user to select the level that should automatically be applied:



Figure 14-2 Automatic Protect choice

2. **Prompt** – FDT will *ask* what to do during the disconnect, i.e. whether to protect or not. The choice is between full protection (or other levels if supported by the device) or no protection at all. The default is 'No'.



Figure 14-3 Disconnect Protect choice

3. **None** – FDT *will not* protect the device on disconnect.

As the disconnect actually happens the result of the operation will be echoed to the Output window. There are several ways of changing this setting depending on the current mode of operation:

### 14.2.1 Toolbar

In full FDT mode and with a QzROM based project the QzROM toolbar will be enabled. This shows the currently chosen option, which can be changed by simply clicking on an adjoining icon.



Figure 14-4 Automatic



Figure 14-5 Prompt



Figure 14-6 None

Note – clicking on the icons will not immediately protect the device. This is still merely the current disconnect setting.

### 14.2.2 Project

It is also possible to change this option from the project properties dialog, as well as viewing the current setting. This is done from the Programmer tab:

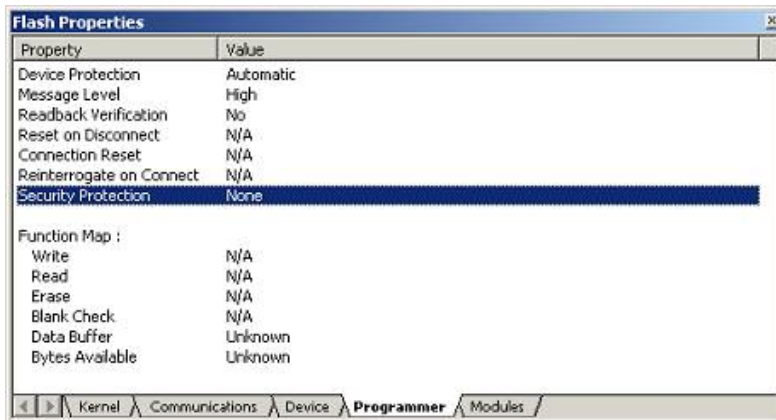
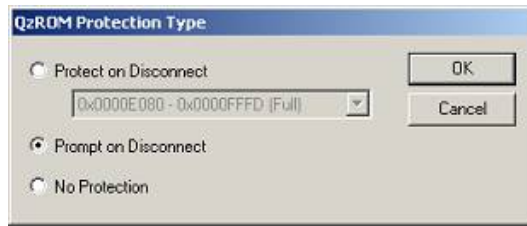


Figure 14-7 Security Protect property

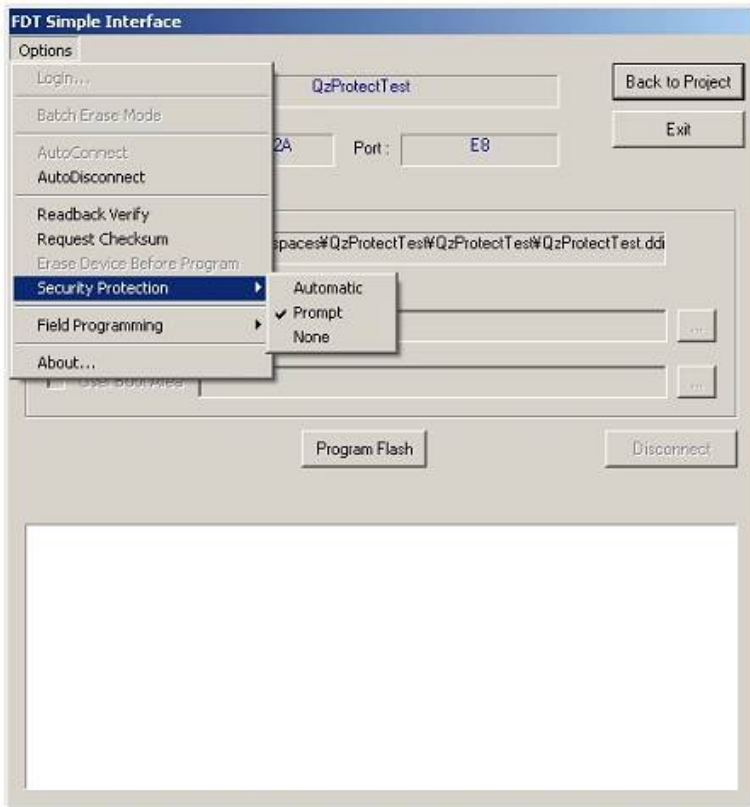
Double clicking on the “Security Protection” field will bring up a new dialog allowing the entry to be changed.



**Figure 14-8 Security Protect dialog change**

### 14.2.3 Simple Interface \ Basic Mode

Should the user be in a mode where only the Simple Interface dialog is accessible, this setting can instead be changed via a menu option.



**Figure 14-9 Security Protect Simple Interface**

The check mark shows the currently selected option. Note – these menu options will be disabled if the project is not QzROM based.



# Chapter 15 Locking

## 15.1 Locking Overview

Locking of erase blocks against writing or erasing is a feature supported by some of the M16C family devices.

FDT offers control over these locking settings at connection, disconnection and during normal operations. The options are available from a toolbar button and a menu entry:



Figure 15-1 ToolbarButton

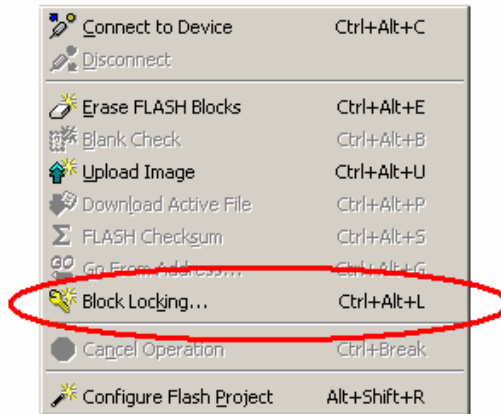


Figure 15-2 Menu Entry

Each erase block that supports the feature can be locked and unlocked individually (*Note: an unlock can only be performed by erasing the block*). However, the setting will have no effect until disconnecting to the device. The lock status of each erase block can be set automatically at disconnect by FDT.

## 15.2 Connection

If FDT is connecting to a device that supports the locking feature it will check the current locked state of each erase block that has a corresponding lock bit (as defined in the project configuration file). Then, depending on the current connection setting (selectable from the Locking Dialog - see section 15.3), FDT will do one of three things:

### 1. Automatically Disable Locks

FDT will automatically disable the locking of device on connection. This will allow all blocks on the device to be written or erased - whether they are currently locked or not.

### 2. Query User

FDT will ask the user what to do about the locking of device on connection:



**Figure 15-3 Query on Connection Message**

If "Yes" is clicked, then the master lock bit will be disabled, as in (1). If "No" is selected then the locking settings will be left alone, as below.

### 3. Do Nothing

FDT will not change any of the current lock settings on the device. Subsequently, FDT will abort the operation if an attempt is made to write or erase to a locked block.

## 15.3 Locking Dialog

The Locking Dialog, accessible from either the toolbar or the 'Device->Block Locking...' menu, controls how FDT behaves with those devices that support protected erase blocks. The dialog, with example device is shown below:

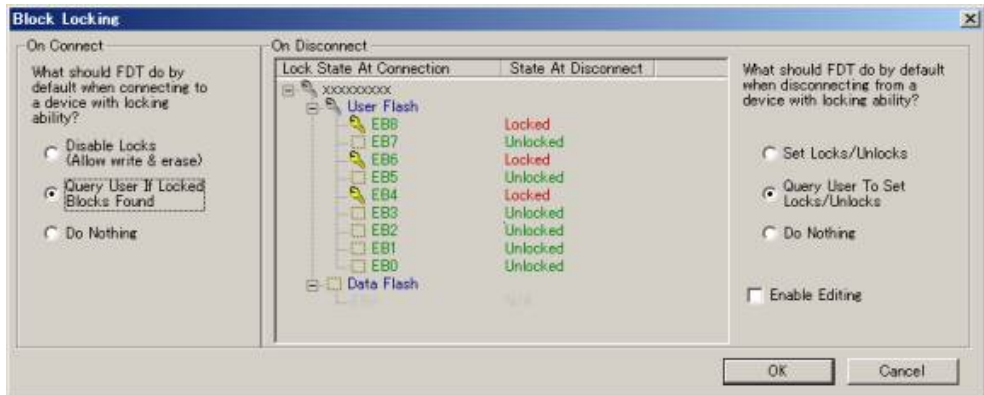


Figure 15-4 Block Lock Configuration Dialog

### 1. On Connect

The left hand side of the dialog controls how FDT behaves as it connects to a device with locking ability. See more on this in the Connection section.

### 2. On Disconnect

The right hand side of the dialog controls how FDT behaves as it disconnects from a device with locking ability. See more on this in the Disconnection section.

### 3. Lock State Tree View

The central part of the dialog shows the lock status of the device as a tree view, similar to the Erase Blocks Dialog. The view highlights several important pieces of information:

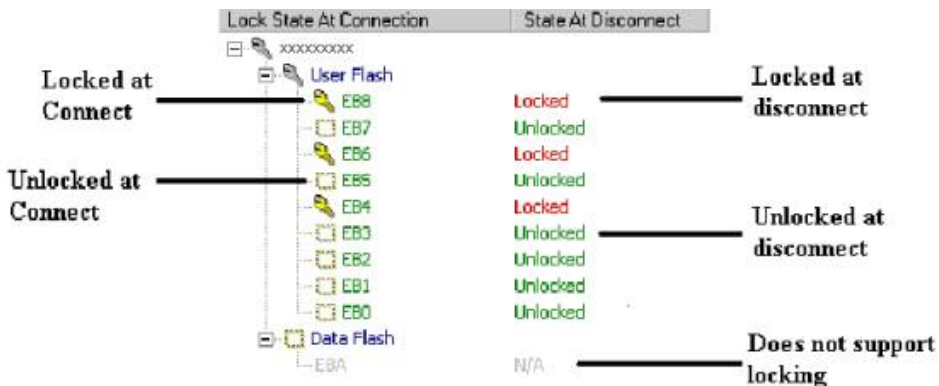


Figure 15-5 Lock State Tree View Settings

## Lock State at Connection

The 'Lock State at Connection' column shows the lock state of each erase block as FDT made the initial connection. There are two symbols to show this state:



- the erase block was found to be locked at the initial connection



- the erase block was found to be unlocked at the initial connection

The erase block may have no symbol at all and be greyed. This is a sign that the block does not have locking support. This is automatically detected by FDT from the project configuration file.

## State at Disconnect

The 'State at Disconnect' column shows what the lock state of each erase block will be set to when FDT disconnects from the device (assuming that the user does not select the 'Do Nothing' radio option). There are two possible values:

**Locked** - the erase block will be protected at disconnect.

**Unlocked** - the erase block will be unprotected at disconnect. NB - if the block was locked at connection FDT will have to erase the block to unlock it. This action will be confirmed as necessary (see the Disconnection section).

If the erase block does not support locking this entry will read "N/A". By default the disconnect state will mirror the lock state found at connection. To change this default, check the 'Enable Editing' box next to the tree view. This will enable the view and allow the disconnect settings to be changed. Double-clicking the setting will toggle it's current state.

*Note - if FDT is not yet connected to the device then this tree will be blank. In addition the 'Enable Editing' checkbox will be unavailable.*

## 15.4 Disconnection

Depending on the current disconnection setting (selectable from the Locking Dialog - see section 15.3), FDT will do one of three things when a user chooses 'Disconnect' from the toolbar or menu:

### 1. Automatically Set

As part of the disconnection process FDT will automatically lock and unlock those blocks specified in the locking dialog. *NB - should locked blocks need unlocking FDT will prompt to confirm the action as it would require those blocks to be erased.*



Figure 15-6 Erase Confirmation Message

If the user selects "Yes" FDT will lock and unlock as appropriate. If the user selects "No" then no erase will occur, no locks will be set - *however, the disconnection operation will continue.*

### 2. Query User

If the Query user option is selected then FDT will display a limited view of the Locking Dialog. This shows the lock state of the device at connection, and the lock state of the device as it will be set at disconnection. This state can still be edited at this point by checking the 'Enable Editing' checkbox. Two other options are given, set the locks or do nothing (the default):

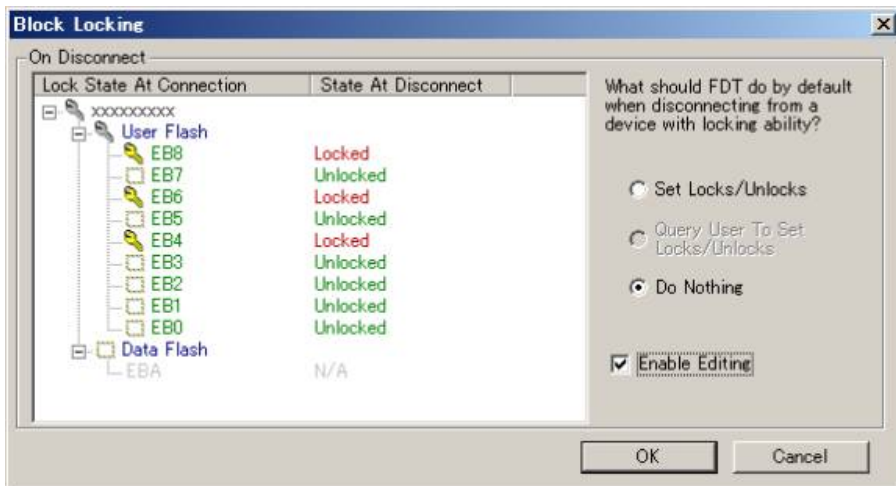


Figure 15-7 Query on Disconnect Dialog

Selecting to set the locks will cause FDT to lock and unlock the blocks as appropriate and also set the master lock enable - as in (1) above. Selecting to do nothing will cause FDT to ignore any locking settings and skip straight to the disconnection process - as in (3) below. Clicking 'Cancel' at this point will cancel both the dialog and the disconnect operation as a whole.

### **3. Do Nothing**

FDT will issue no lock or unlock commands and the device will be left in its current locking state. The disconnection process will happen as normal.

## Chapter 16 Upgrading to FDT3.07

Installing FDT3.07 should not adversely affect your other FDT version installations. The USB drivers may be upgraded by installing FDT3.07, and the Access Rights usernames and passwords will be common, but these are the only shared components. Each version of FDT installs its own set of kernels and keeps its own settings. Installing FDT3.07 will also update the support email addresses for some previous versions of FDT. The most recently installed version of FDT will be the application registered for use with the FDT file extensions. This is so that the most recently installed version will be used when a file is double clicked from Windows® Explorer.

If you are an existing user of FDT, and your existing project needs to be migrated to FDT3.07, the recommendations are as follows:

If the project specifically requires the kernels distributed with an earlier version of FDT (or a customised kernel based upon an older version), it is possible to choose the “Other” option to select the “User Defined” kernel. It is recommended however, that wherever possible, the latest kernels be used. Using kernels from previous versions of FDT may lead to a decrease in functionality (e.g. FDT 3.1 and above kernels include the information that FDT requires to calculate the asynchronous serial error rates; use of kernels distributed prior to FDT 3.1 will result in this feature being disabled).

For projects using FDT versions prior to FDT 3.0, a new workspace and project should be created using the wizard. This will ensure that the new settings are generated as required by FDT and is a very quick process using the Project Wizard. The workspace files used by FDT versions prior to version 3.0 are not compatible with FDT3.07 due to the enhancements since then.

Workspaces from previous versions of FDT (from version 3.0) can be opened with FDT3.07 but should not be opened with the original version again after they have been upgraded. Any errors related to “SyntaxHighlightingFile.dbs” can be ignored in this case. Toolbar positions and states may also be affected by opening old format workspaces. File associations may need setting up once the project has been upgraded to allow double clicking to open the selected file and downloading.

The new style of device image (.ddi based rather than .fpr based) will mean that FDT will attempt to rebuild any project image from a previous version. The old image will not be removed or deleted but a new one will be added and used henceforth. If the old image must be used for download then simple right click on the entry in the Workspace view and choose ‘Download File’ from the context menu.

FDT3.07 has the following new features:-

- Support for Lock Bit commands
- Support for R8C/2x Blank Check embedded function
- BPS (synchronous) selection support for QZ-ROM
- Display Clock Frequency / CKM / CKP in output Window
- SH2A Data Mat Support





---

**Renesas FLASH Microcomputer Programming System  
User's Manual  
Renesas FLASH Development Toolkit 3.07  
(for Windows® 98SE/Me, Windows NT® 4.0, Windows® 2000 and  
Windows® XP)**

Publication Date: Rev.8.00, October 31, 2006  
Published by: Sales Strategic Planning Div.  
Renesas Technology Corp.  
Edited by: Customer Support Department  
Global Strategic Communication Div.  
Renesas Solutions Corp.

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan

---



## RENESAS SALES OFFICES

<http://www.renesas.com>

Refer to "<http://www.renesas.com/en/network>" for the latest and detailed information.

### **Renesas Technology America, Inc.**

450 Holger Way, San Jose, CA 95134-1368, U.S.A  
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

### **Renesas Technology Europe Limited**

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.  
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

### **Renesas Technology (Shanghai) Co., Ltd.**

Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120  
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7898

### **Renesas Technology Hong Kong Ltd.**

7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong  
Tel: <852> 2265-6688, Fax: <852> 2730-6071

### **Renesas Technology Taiwan Co., Ltd.**

10th Floor, No.99, Fushing North Road, Taipei, Taiwan  
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

### **Renesas Technology Singapore Pte. Ltd.**

1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632  
Tel: <65> 6213-0200, Fax: <65> 6278-8001

### **Renesas Technology Korea Co., Ltd.**

Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea  
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

### **Renesas Technology Malaysia Sdn. Bhd**

Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia  
Tel: <603> 7955-9390, Fax: <603> 7955-9510



**Renesas FLASH Development Toolkit 3.07**  
**(for Windows® 98SE/Me, Windows NT® 4.0,**  
**Windows® 2000 and Windows® XP)**

**User's Manual**



**Renesas Electronics Corporation**

1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ10J1131-0900