



Binary Configuration Tool (BCT)

User Guide

June 2013



You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting: <http://www.intel.com/design/literature.htm>.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at <http://www.intel.com/> or from the OEM or retailer.

No computer system can be absolutely secure.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2016, Intel Corporation. All rights reserved.



Contents

1.0	Introduction	6
2.0	Download and Install Instructions	7
2.1	Windows Installation.....	7
2.2	Linux Install.....	8
3.0	Command Line Interface	10
4.0	Quick Start Guide	12
4.1	Changing Binary Configuration Settings	12
4.2	Creating a Final Firmware Image from an ABSF	15
5.0	Rebasing	17
5.1	Rebasing the Load Address of a Binary	17
5.2	Optional Rebasing the Load Address of the Created Firmware.....	20
6.0	GUI Overview	23
6.1	BCT Major Graphical Components.....	23
6.2	Menu Overview	23
6.2.1	File Menu	23
6.2.2	Preferences	24
6.2.3	View Menu	24
6.2.4	Settings Menu	25
6.2.5	Binary Tools Menu.....	26
6.2.6	Help Menu.....	27
6.3	Configuration Setting Value Indicators.....	29
6.4	Controlling View of the Main Panes	30
7.0	License	31
8.0	Glossary	35

Figures

Figure 1.	Basic Inputs and Outputs of the BCT	6
Figure 2.	Intel® Binary Configuration Tool Setup.....	8
Figure 3.	Three Panes on BCT	13
Figure 4.	Select Patch Menu Item.....	13
Figure 5.	Patch Binary File to Create Firmware File	14
Figure 6.	Adding .rom File Extension.....	15
Figure 7.	Dialog Box to Create Firmware Image from an ABSF.....	16



Figure 8.	Rebase Binary File Dialog Box.....	17
Figure 9.	Change Load Address of Binary	18
Figure 10.	Invalid Hexadecimal Entry	19
Figure 11.	Hexadecimal Character Check.....	20
Figure 12.	Rebasing the Load Address – Form 1	21
Figure 13.	Rebasing the Load Address – Form 2	22
Figure 14.	Preferences Dialog.....	24
Figure 15.	Settings Menu.....	25
Figure 16.	Binary Configuration Setting Values	26
Figure 17.	Help Menu.....	27
Figure 18.	About Dialog	28
Figure 19.	About Dialog - Licensing Agreements.....	29

Tables

Table 1.	View Menu Selections.....	25
Table 2.	Configuration Settings	25
Table 3.	Binary Tools Menu	26
Table 4.	Help Menu Selections.....	28
Table 5.	Icons of Configuration Settings.....	29
Table 6.	Glossary	35



Revision History

Date	Revision	Description
June 2013	1.2	Expanded section on the command line interface, updated screen shots, and removed redundancies.
March 2013	1.1	Updated Release package details.
January 2013	1.0	Initial release.

§

1.0 Introduction

The Binary Configuration Tool (BCT) is a utility used to change configuration settings embedded in a binary file. This enables the customer to use the BCT to customize the static FSP configuration parameters which are part of the Intel® FSP binary.

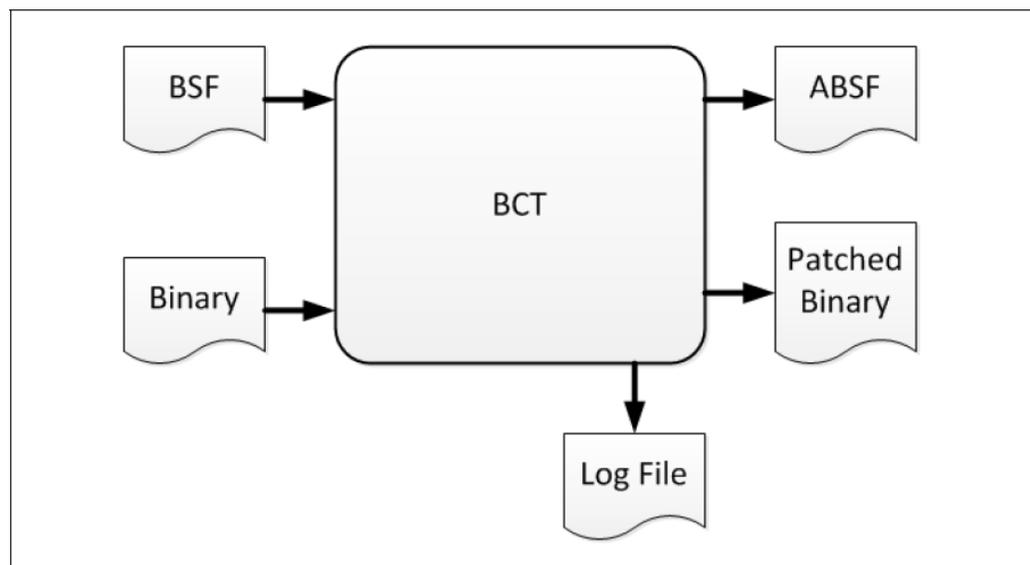
[Figure 1](#) below illustrates the basic input and outputs of the BCT.

The BCT reads a Boot Settings File (BSF) to provide a graphical interface for manipulating a binary's configuration settings. The BCT user can patch a binary with modified configuration settings.

In addition to the patched binary file, the BCT also produces an ABSF which is an augmented BSF that contains the configuration settings used to patch the binary file.

The BCT also produces a log file in the user's home directory named **bct.log**.

Figure 1. Basic Inputs and Outputs of the BCT





2.0 Download and Install Instructions

The tool can be downloaded as three separate packages, depending on the OS of the target system:

- Windows installer with support for both 32- and 64-bit: bct-3.0.2-3-i686.win32.exe
- Linux installer with support for 32-bit: bct-3.0.2-3-i686.fc14.tar.gz
- Linux installer with support for 64-bit: bct-3.0.2-3-x86_64.fc14.tar.gz

Follow install instructions as detailed in the following sections. In each case the documentation is delivered with the package.

- Windows install: The supporting documents are in the installation directory.
- Linux install (both): The supporting documents are in the documentation directory.

Note: Make sure you chose the right package to match your system to download and install. The screen shots in this document show examples for Crown Bay or Bay Trail.

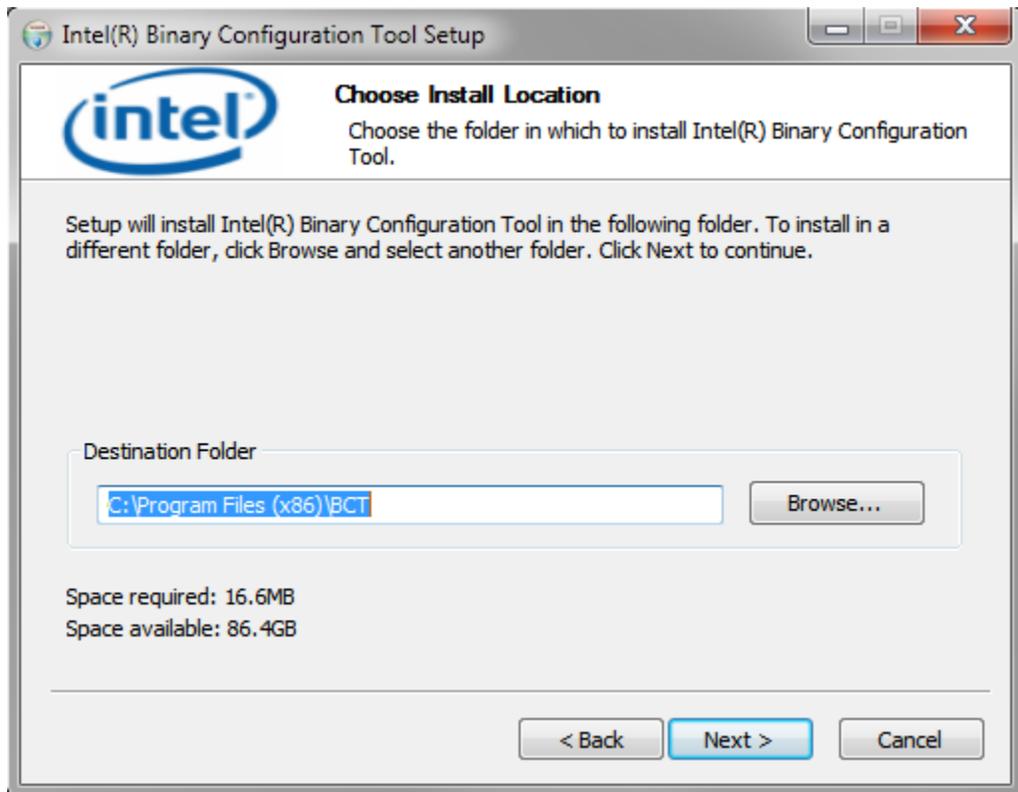
The platform name displayed during your installation will be based on the target platform you are using.

2.1 Windows Installation

The Windows Installer will copy files to `C:\Program Files (x86)\BCT` by default. The Windows UAC will prompt you to grant permission for the installer to start.



Figure 2. Intel® Binary Configuration Tool Setup



When BCT is run for the first time and the user accepts the End User License Agreement, the following files are created in current user's home directory (C:\Users\

- .bctrc
- .bctstyle
- bct.log

2.2 Linux Install

1. Copy the BCT `tar.gz` file to a local directory.
 - a. For 32-bit versions, choose `bct-<current version>.i686.fc14.tar.gz`
 - b. For 64-bit versions, choose `bct-<current version>.x86_64.fc14.tar.gz`
2. From the console terminal, navigate to the folder that contains the BCT `tar.gz` file and issue the following command:
 - a. For 32-bit versions, `>tar -xvf bct-3.0.2-1-i686.fc14.tar.gz`
 - b. For 64-bit versions, `>tar -xvf bct-3.0.2-1-x86_64.fc14.tar.gz`
3. The installation extracts a binary executable, `bct`, and three folders: `help`, `xrc`, and `documentation` inside a folder named `bct`.



4. Run the BCT program by issuing:

```
>cd bct  
>./bct
```

§



3.0 Command Line Interface

The following command line is supported by the BCT:

```
bct [-v] [-h] [-d LEVEL] [-b BSF] [-i BIN] [-a [ABSF]] [-r [ADDR]] [-o BOUT] [-g] [-n]
```

where all arguments are optional and have the following meaning:

- v, --version Emits version number and exits.
- h, --help Show this help message and exits.
- d LEVEL, --debug LEVEL
 Turn on LOGFILE messages at a specific log LEVEL which can be one of the following DEBUG, INFO, WARNING, ERROR, or CRITICAL. The default LEVEL is INFO.
- b BSF, --bsf BSF
 Specifies the Boot Settings File, BSF. If BSF is used on the command line without specifying BIN or ABSF then BCT opens BSF in the GUI.
- i BIN, --bin BIN
 Specifies the Binary input file, BIN, to be configured.
- a [ABSF], --absf [ABSF]
 Specifies the 'As Built' Boot Settings File, ABSF, to create. If ABSF is omitted, an ABSF will be generated in CLI Mode and named after the BSF except with a .absf file name extension. See CLI Mode below.
- r [ADDR], --reloc [ADDR]
 Specifies the relocation load address, ADDR, of BIN which must also be specified via --bin or -i. The ADDR value must be hexadecimal, with or without the 0x prefix. If ADDR is omitted, then the binary's current load address will be emitted.
- o BOUT, --bout BOUT Specifies the binary output file name, BOUT.
- g, --gui Forces the graphical user interface to run.
- n, --info Displays the description table in the binary, if it has one.



The argument values [ABSF] and [ADDR] are optional.

CLI Mode

The following argument signatures will cause BCT to execute in command line mode without a GUI, unless the --gui option is also specified:

(--bin|-i) BIN (--bsf|-b) BSF [(--reloc|-r) ADDR] [(--bout|-o) BOUT]

Patch binary BIN using setting in BSF, optionally rebase with load address ADDR and specify output file BOUT. An ABSF will be created with the same filename as BSF except with a .absf extension. If BOUT is omitted then BCT creates a new binary with the same filename as BIN except with a .rom extension. Please note that if you want to rebase the load address, a valid address must be specified. It is not optional in this case.

(--bin|-i) BIN (--absf|-a) [ABSF] [(--reloc|-r) ADDR] [(--bout|-o) BOUT]

Patch binary BIN using the settings in ABSF, optionally rebase with load address ADDR and specify output file BOUT. If ABSF is omitted, BCT will attempt to use an ABSF file with a path generated from BIN's filename with a .absf extension. If BOUT is omitted then BCT creates a new binary with the same filename as BIN except with a .rom extension. Please note that if you want to rebase the load address, a valid address must be specified. It is not optional in this case.

(--bin|-i) BIN (--reloc|-r) [ADDR] [(--bout|-o) BOUT]

Relocate the base address of the binary specified by BIN and save it as BOUT. If ADDR is omitted, then the current base address of BIN is printed. If BOUT is omitted then BIN will be over-written with the new base address.

(--bsf|-b) BSF (--absf|-a) [ABSF]

Create an ABSF file from a BSF file, using the defaults specified in BSF. If ABSF is omitted, then an ABSF file is created with the name of the BSF with a .absf extension, otherwise BCT saves to ABSF.

(--bin|-i) BIN (--info|-n)

Print the description table that's in BIN, if it has any.



4.0 Quick Start Guide

The following sections document the basic work flow usage model of the BCT and are intended to educate the user as quickly as possible in the basic usage of the BCT. For a comprehensive walk through of the BCT's Graphical User Interface components see Section [6.0 GUI Overview](#).

4.1 Changing Binary Configuration Settings

To modify one or more configuration settings in a binary file, the following BCT usage model should be followed after starting the BCT:

1. Open the BSF associated with the binary file.

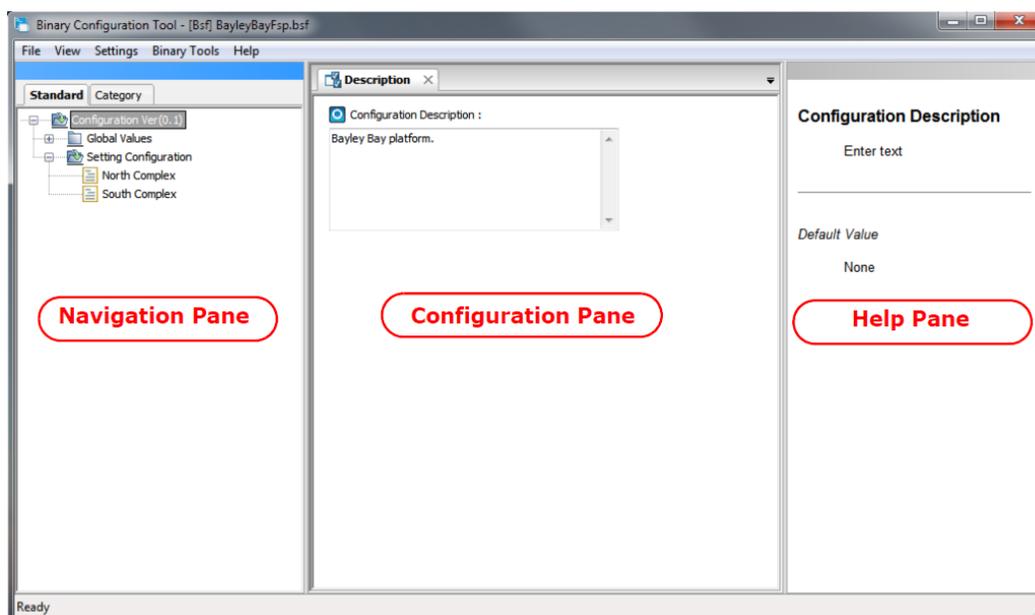
This can be accomplished by doing **one** of four actions below:

- Selecting the File --> Open menu item
- Selecting the Open BSF button
- Selecting the Settings --> Load menu item
- Select a file from the Recently Used Files list. File --> <File>

Configuration settings displayed in the center pane (the **Configuration Pane**) can be modified. The right pane (the **Help Pane**) will change as configuration settings are floated over by the mouse cursor. The left pane (the **Navigation Pane**) can be used to traverse the configuration settings tree under the **Standard or Category** tabs.



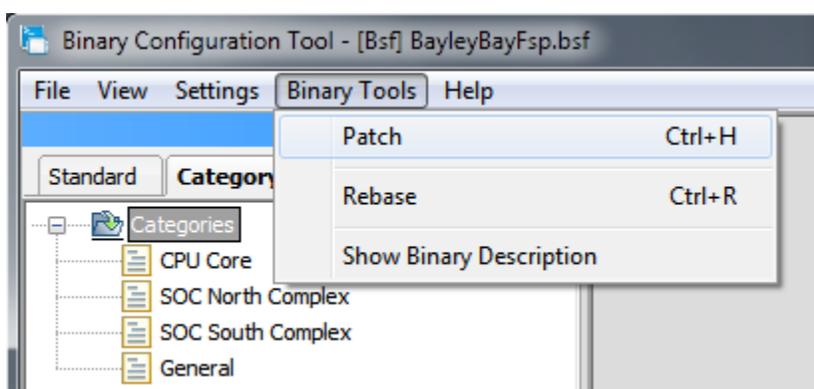
Figure 3. Three Panes on BCT



The **Category** tab offers an alternative way of navigating the configuration settings tree.

2. After you have modified the configuration settings as necessary, patch the binary associated with the BSF by selecting the Binary menu's **Patch** menu item.

Figure 4. Select Patch Menu Item



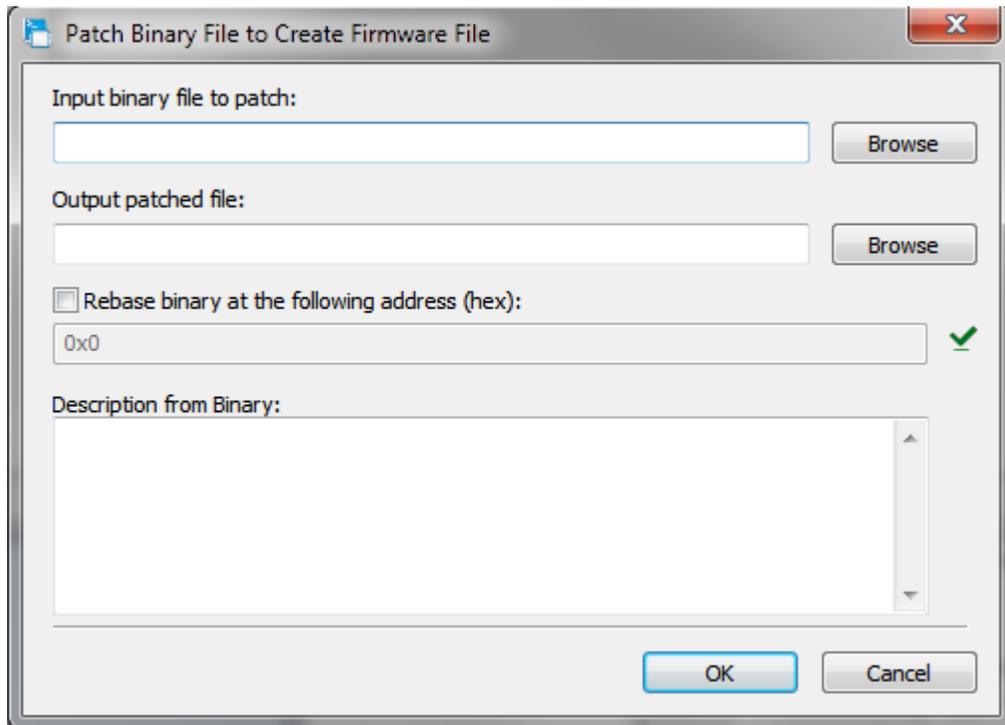
The configuration settings are saved to a ABSF.

Note: The ABSF file name used has the same base name as the BSF file but with a `.absf` file name extension rather than a `.bsf` file name extension.

The ABSF file can be used at a later date to update a binary with its saved configuration. See Section [4.2 Creating a Final Firmware Image from an ABSF](#). After

the configuration settings have been saved to the ABSF, another dialog will be displayed asking the user to browse to the binary file associated with the BSF and to provide a new firmware image file name that will contain the patched configuration settings.

Figure 5. Patch Binary File to Create Firmware File



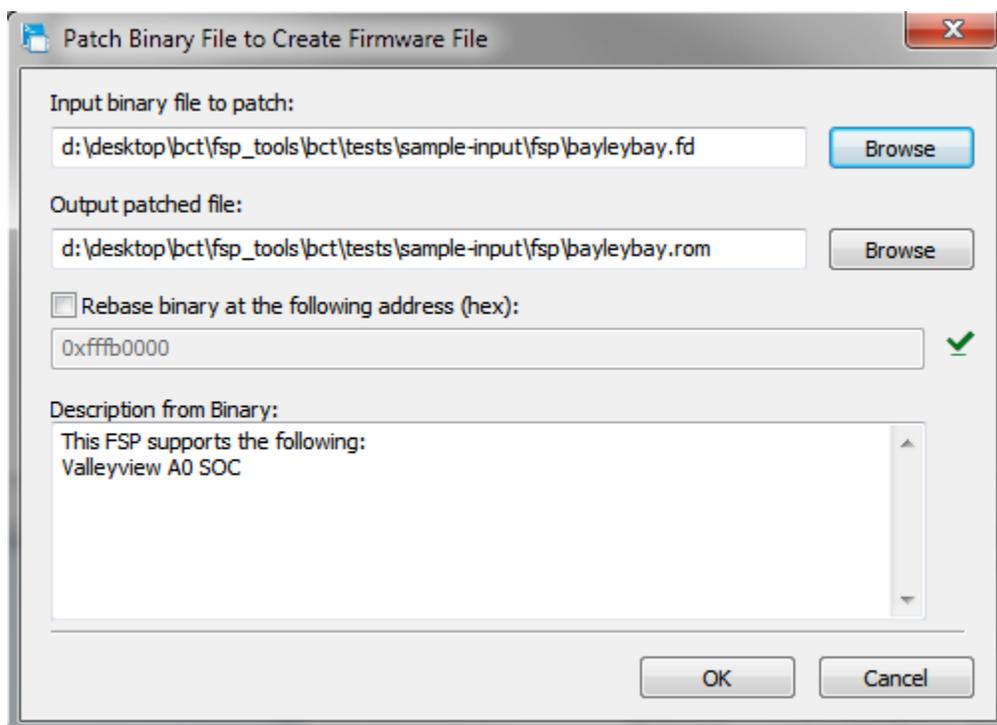
If the binary contains a description table, it will be displayed in the “Description from Binary” area.

3. After both the binary file and the firmware file names have been supplied, select the **OK** button to patch the binary file and create the firmware file specified.

Note: By default the firmware file name is constructed by taking the binary file's base name (its name without a file extension) and adding the file extension `.rom` as shown below.



Figure 6. Adding .rom File Extension



Optionally, the load address of the firmware file can be rebased. See the next section.

Successful creation of a firmware image will display “Firmware file has been created successfully.” in the status bar.

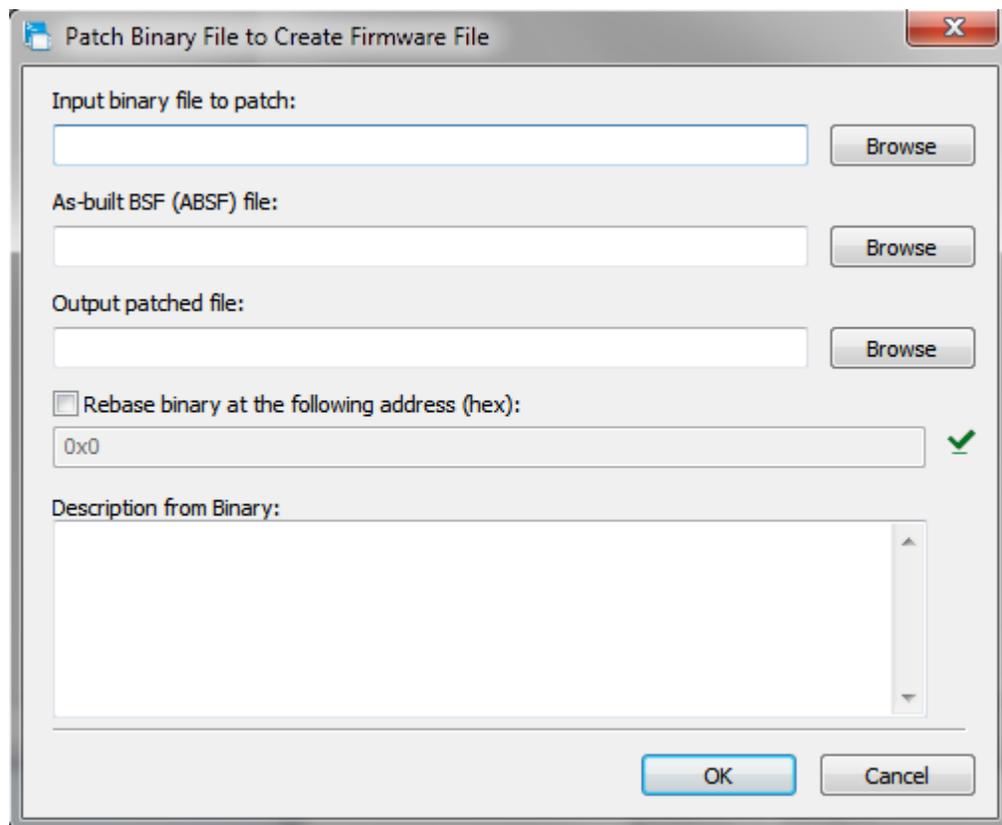
4.2 Creating a Final Firmware Image from an ABSF

When a binary file has its configuration settings changed an ABSF is generated. This **ABSF** file can be used to modify binary files without the user having to load a BSF and manually change the configuration settings as described in the previous section.

To update a binary file with settings saved in an **ABSF**, select the **Binary --> Patch** menu *without* previously loading a **BSF**. Or, click “Patch Binary with ABSF” before loading a BSF.

This action will display the following dialog:

Figure 7. Dialog Box to Create Firmware Image from an ABSF



Use the **Browse** buttons to select the Binary, As built (**ABSF**) file, and the Firmware file to be created.

Optionally, the Firmware file can be rebased. See the next section for details.

§

5.0 Rebasing

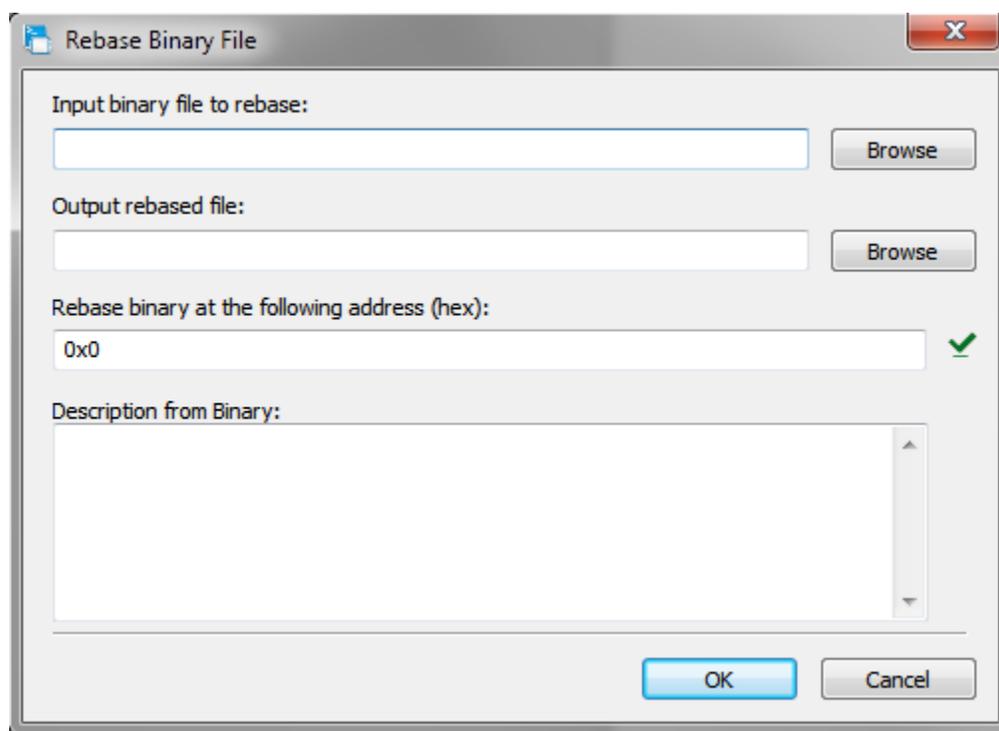
During integration of the Intel® FSP and the host boot loader, you will need to place the Intel® FSP binary at a default physical location based on the instructions provided by the FSP Integration Guide for that platform. If the default address needs to be relocated, you need to use the following Rebase tool to relocate the binary to another location in ROM.

The following section explains the process.

5.1 Rebasing the Load Address of a Binary

A binary file can be rebased without using a BSF by selecting the **Binary --> Rebase** menu item or “Rebase a Binary” before a BSF is loaded. The following dialog is used to input the binary file name to rebase and the rebase address:

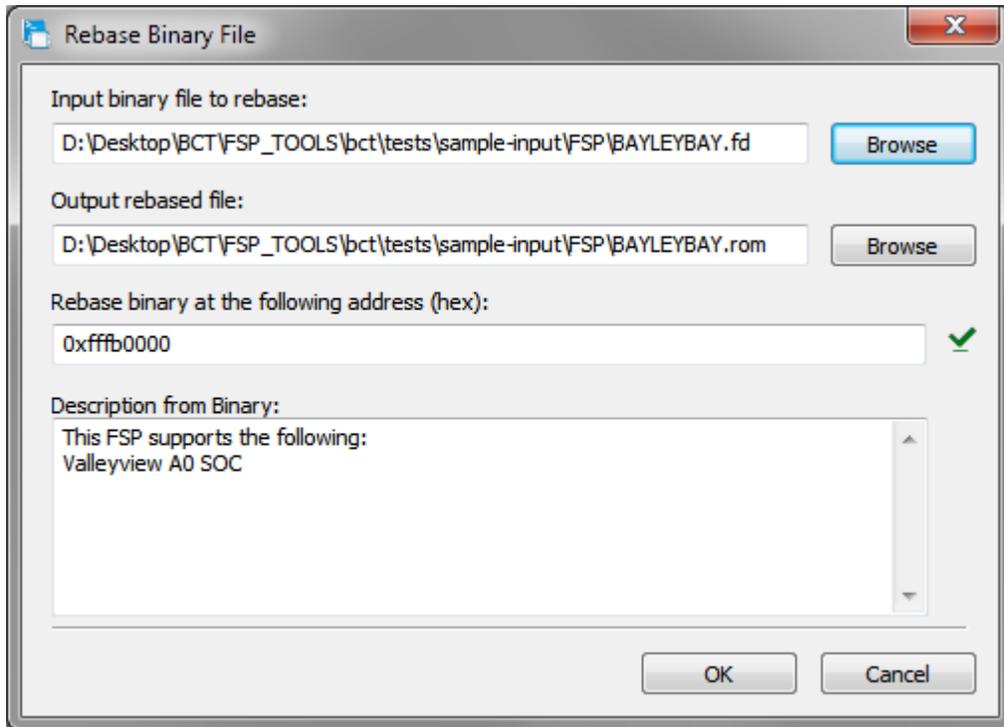
Figure 8. Rebase Binary File Dialog Box



The **Browse** button is used to navigate and select the binary file to be rebased. If the selected binary is a FSP, its current load address will appear in the **Firmware File Rebase HEX Address** input string widget.

To change the load address of the binary check the **Enable Rebase** check box.

Figure 9. Change Load Address of Binary



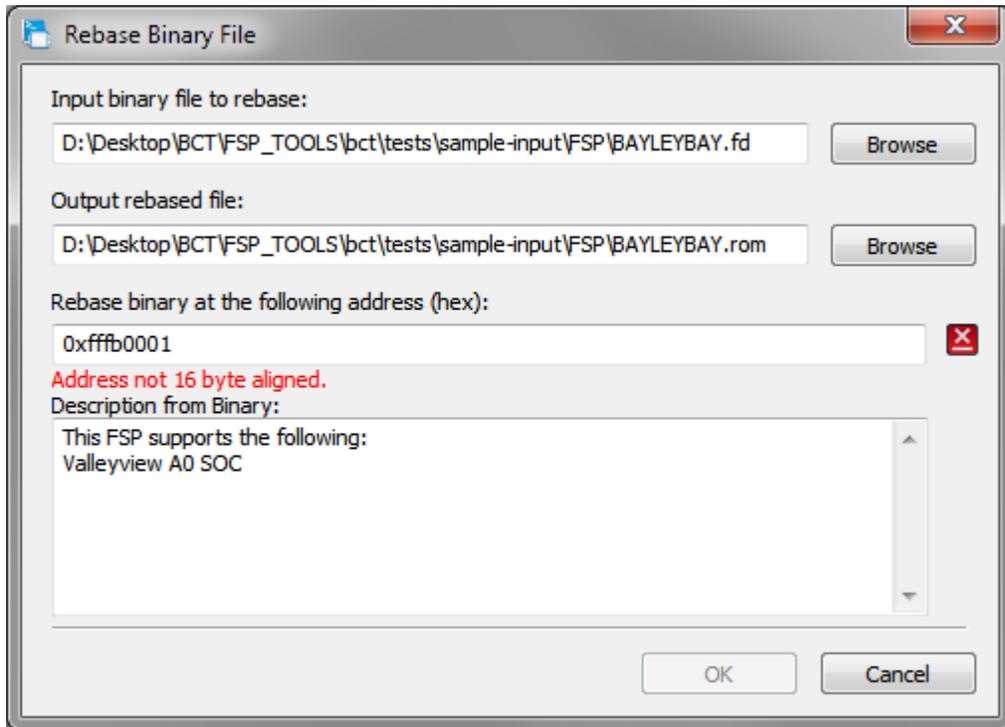
A valid hexadecimal can be entered in the **Firmware File Rebase HEX Address** field. A valid range is 0x00000000 thru 0xFFFFFFFF. The format of the entry can include uppercase or lowercase hex characters and any combination of the following prefixes and suffixes:

- a **Ox** prefix, as in 0xFFFFA000
- a **H** suffix, as in FFFFA000H
- a **h** suffix, as in FFFFA000h
- a **L** suffix, as in FFFFA000L

If the user enters an invalid hexadecimal entry, the green check mark turns to a red X and a descriptive error message will be displayed. Validation tests include a range check, as shown in the following example.

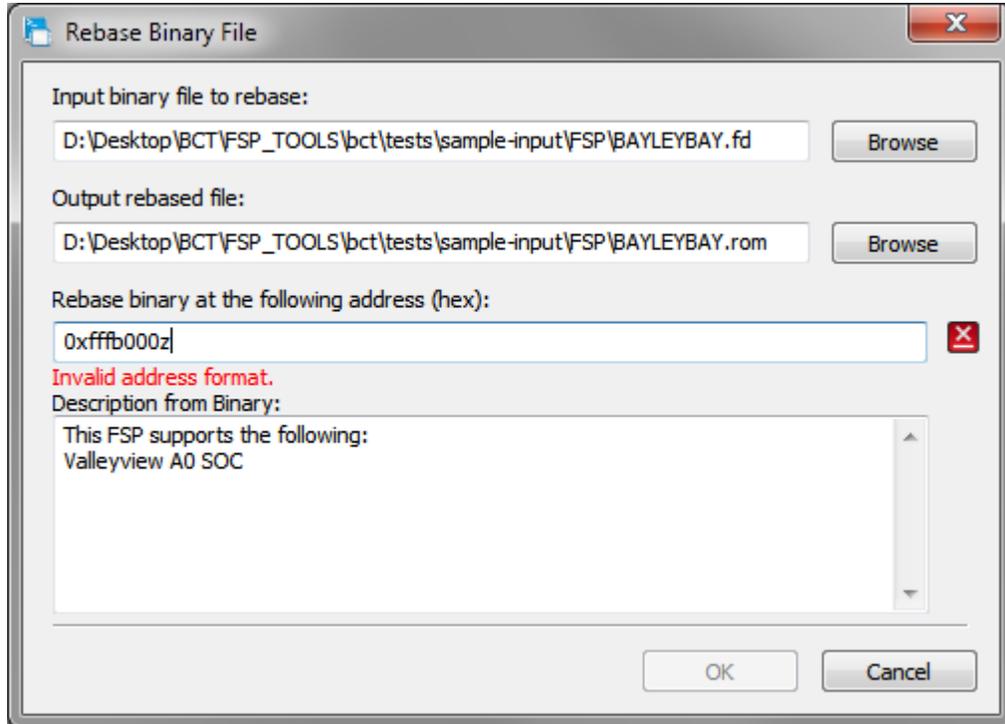


Figure 10. Invalid Hexadecimal Entry



Validation tests include a hexadecimal character check, as shown below:

Figure 11. Hexadecimal Character Check



A successfully validated hexadecimal address will not generate an error dialog.

Selecting **OK** will patch the binary file creating the firmware file specified and then rebase the firmware file.

Note: The rebase process generates its own log file rebase.log.

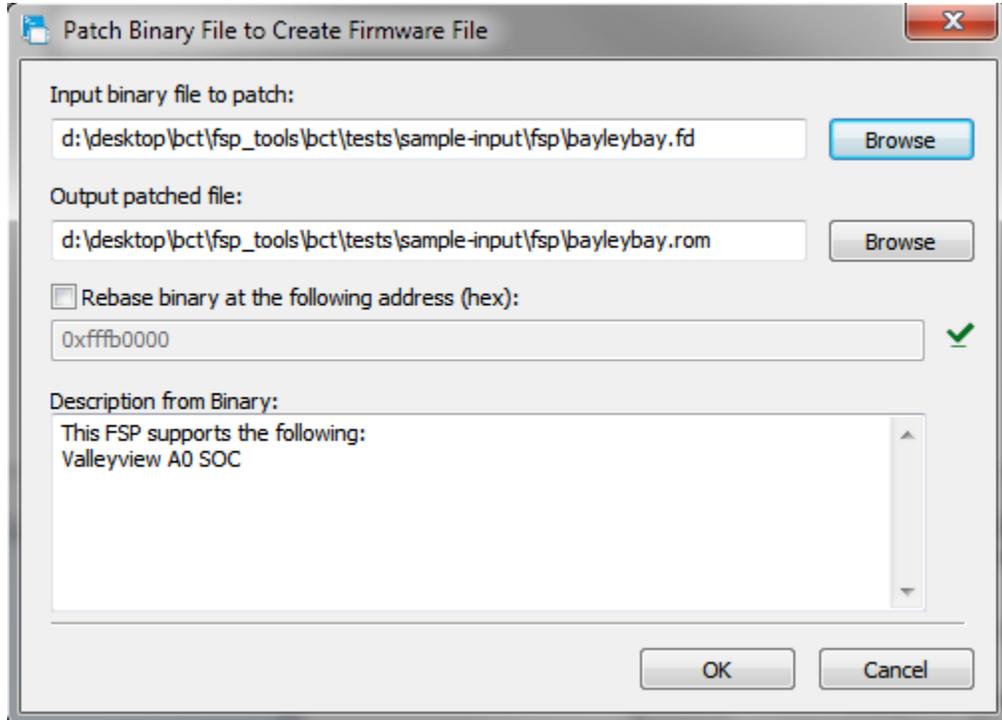
5.2 Optional Rebasing the Load Address of the Created Firmware

Both forms of the dialog titled **Patch Binary File to Create Firmware File** can optionally rebase (change the load address) the created firmware file (*.rom).



In this form:

Figure 12. Rebasing the Load Address – Form 1





Or this form:

Figure 13. Rebasing the Load Address – Form 2

Patch Binary File to Create Firmware File

Input binary file to patch: Browse

As-built BSF (ABSF) file: Browse

Output patched file: Browse

Rebase binary at the following address (hex):
 ✓

Description from Binary:

OK Cancel

§



6.0 GUI Overview

This section provides a brief description of each graphical interface component.

6.1 BCT Major Graphical Components

- The **Menu Bar** is the primary interface through which the user interacts with the BCT to open a BSF and patch the binary file.
- The **Navigation Pane** provides an interface for the user to traverse the configuration settings tree (once a BSF is opened) and change the view provided in the Configuration Pane.
- The **Configuration Pane** provides an interface for the user to manipulate specific configuration settings.
- The **Help Pane** by default is enabled to display additional help information about a specific item selected in the Configuration Pane.
- The **Status Bar** typically displays help messages when a menu or toolbar item is under the mouse cursor.

6.2 Menu Overview

This section provides a quick overview of the various BCT Menus.

6.2.1 File Menu

The File Menu options are:

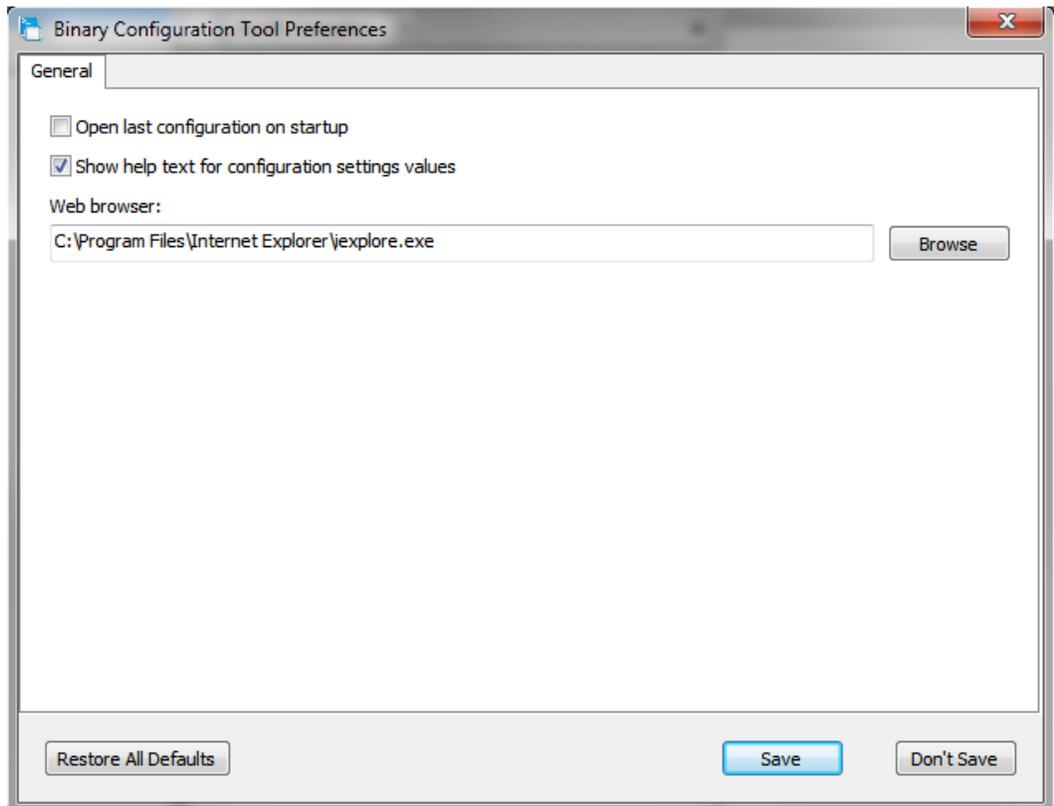
- Open - Open a BSF
- Preferences - Manipulate BCT preferences
- Exit - Exit BCT
- List of the five most recently opened BSFs

Note: The BSF may be opened using the Settings Menu (see Section [6.2.4 Settings Menu](#)).

6.2.2 Preferences

The **File --> Preferences** menu selection opens a Preferences Dialog:

Figure 14. Preferences Dialog



This Preferences dialog allows the user to:

- Choose whether the last configuration should be opened on the next startup
- Choose to suppress configuration setting help messages that appear in the Help Pane
- Define a specific web browser to use to display the help documentation

6.2.3 View Menu

The View Menu provides an interface for the user to manipulate various graphical components.

The View Menu selections have the following meanings.



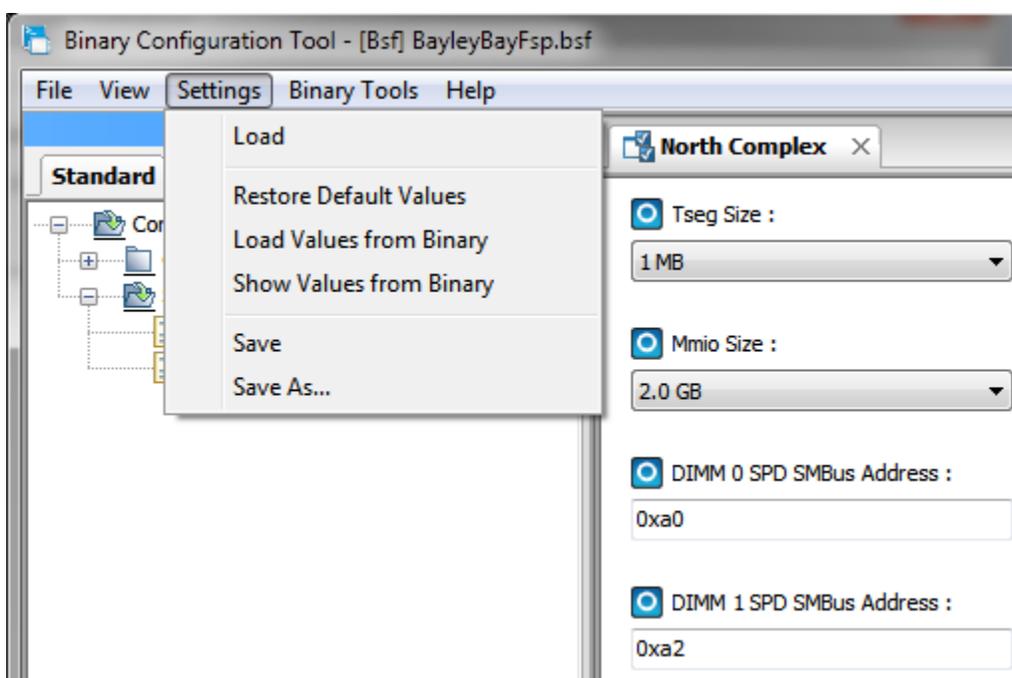
Table 1. View Menu Selections

View Menu Selection	Description
View --> Status Bar	Toggle the display of the Status Bar
View --> Side Windows	Toggle the display of the Navigation, Help, and Output Panes.

6.2.4 Settings Menu

The Settings Menu is show in the figure below.

Figure 15. Settings Menu



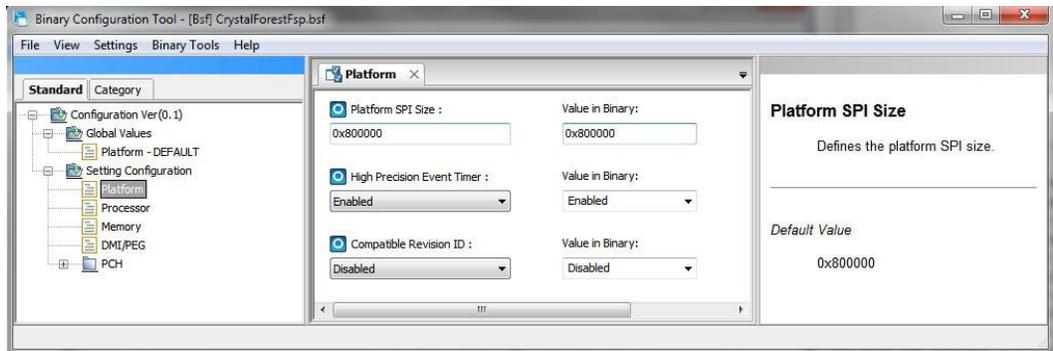
The Settings Menu provides an interface to manipulate configuration settings as defined in the table below.

Table 2. Configuration Settings

Settings Menu Selection	Description
Load	Load a BSF. See also “File Menu”.
Restore Default Values	Re-load default values defined in the BSF. Any configuration setting changes will be restored to default values.

Settings Menu Selection	Description
Load Values from Binary	Load configuration setting values from a binary file. Configuration setting changes will be reset to their values in the binary.
Show Values from Binary	Load configuration setting values from a binary file and display them beside the current configuration settings. See Figure 16 .
Save	Save the current configuration settings to an ABSF that uses the same base name as the BSF, but has a .absf file extension.
Save As...	Save the current configuration settings to an ABSF that the user names.

Figure 16. Binary Configuration Setting Values



6.2.5 Binary Tools Menu

The Binary Tools Menu provides an interface to patch a binary file with the current configuration settings.

Table 3. Binary Tools Menu

Binary Menu Selection	Description
Binary --> Patch	<p>Opens one of two dialogs depending on if a BSF has previously been loaded. If a BSF has previously been loaded, then the dialog will accept:</p> <ul style="list-style-type: none"> • a binary file name to patch • the resulting patched binary (firmware) file name to create <p>See Section 4.1 Changing Binary Configuration Settings</p> <p>If no BSF was previously loaded, then the dialog will accept:</p> <ul style="list-style-type: none"> • a binary file name to patch

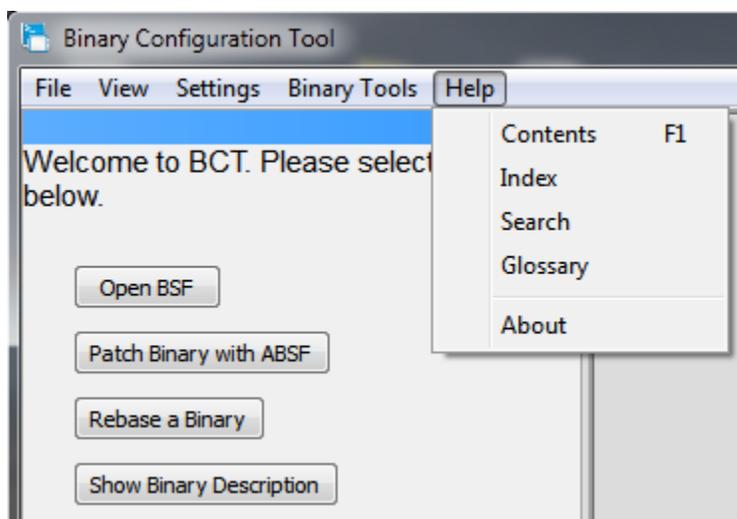


Binary Menu Selection	Description
	<ul style="list-style-type: none"> • an As-built BSF (.absf) file • the resulting patched binary (firmware) file name to create See Creating a Final Firmware Image from an ABSF. Both the above dialogs also contain a rebase check box that will enable rebase address entry and will rebase the firmware file after it has been created.
Binary --> Rebase	Opens a dialog that accepts a binary file name and a rebase hexadecimal address.

6.2.6 Help Menu

The Help Menu provides access to this built-in help documentation and provides access to the About dialog.

Figure 17. Help Menu



The following table defines what each help menu selection does.

Table 4. Help Menu Selections

Help Menu Selection	Description
Help --> Contents	Invokes a web browser to display the Help Table of Contents
Help --> Index	Invokes a web browser to display the Help Index
Help --> Search	Invokes a web browser to display the Help Search page
Help --> Glossary	Invokes a web browser to display the Help Glossary page
Help --> About	Displays the About dialog that identifies the BCT version and provides access to licensing information

6.2.6.1 About Dialog

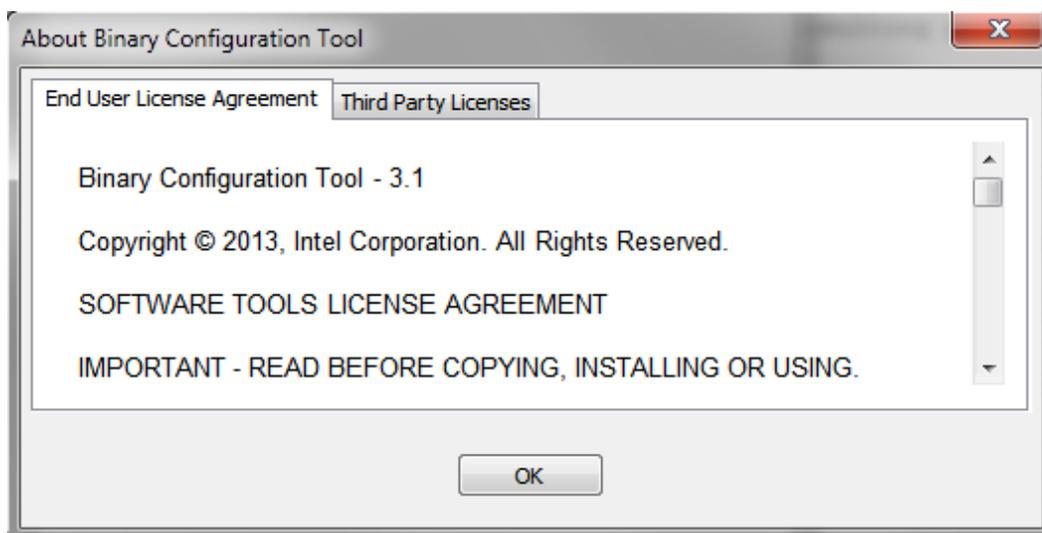
The About dialog displays BCT version information and a License Agreement selection that provides a tabbed interface to read the BCT License Agreement and/or a reference list of third-party licenses associated with the external packages used by BCT.

Figure 18. About Dialog





Figure 19. About Dialog - Licensing Agreements



6.3 Configuration Setting Value Indicators

The following table identifies the various icons that are used to indicate the change or error states of a configuration data value and any help message that might be associated with the state indicated by the icon.

Table 5. Icons of Configuration Settings

Icon	Displayed when	Message
	Normal, unchanged state	None
	Value has been changed, no error detected	None
	An invalid value has been entered	The message will provide information on what type of value is permitted
	Value entered is outside the acceptable range	Entered value is outside acceptable range
	As built value does not match the binary value	None
	A Profile has been selected and the value entered does not match the binary value	None
	Unknown value	The initial value was not set or cannot be determined



Icon	Displayed when	Message
	Two or more values are in conflict	The message will provide information on resolving the conflict

6.4 Controlling View of the Main Panes

The main panes are the **Navigation Pane**, the **Configuration Pane**, and the **Help Pane** (see Section [6.0 GUI Overview](#)).

If you are unable to view the entire width of the screen, you can hide either the right side **Help Pane** or the left side **Navigation Pane**.

Scroll bars will allow you to move vertically and horizontally in the **Configuration Pane**. The **Configuration Pane** cannot be hidden.

§



7.0 License

SOFTWARE TOOLS LICENSE AGREEMENT

IMPORTANT - READ BEFORE COPYING, INSTALLING OR USING. Do not use or load this software and any associated materials (collectively, the "Software") until you have carefully read the following terms and conditions. By loading or using the Software, you agree to the terms of this Agreement. If you do not wish to so agree, do not install or use the Software.

LICENSES: Please Note:

- If you are an end user, the "Single User License" shall apply to you.
- If you are an original equipment manufacturer (OEM), the "OEM License" shall apply to you.
- If you are an Independent BIOS, Operating System or Software Vendor, the "ISV License" below shall apply to you.

DEFINITIONS

Licensed Software means any computing programming code, including Pre-Production Licensed Software, that is provided in connection with, under, or subject to this Agreement, but does not include computing programming code that is (i) delivered with the Licensed Software that is Open Source Software and/or (ii) subject to an agreement, obligation and/or license (in each case whether or not accompanying such computing programming code) intended to supersede or override this Agreement. The file directories, sub-directories and individual File Headers included with the Licensed Software shall contain information to identify the Licensed Software and any Open Source Software.

File Header(s) means the information provided with the Licensed Software and any Open Source Software that identifies (i) Internal Use Software, Distributable As Object Software, Distributable As Sample Source Software, or Open Source Software, and (ii) any notices or additional information about the Licensed Software or Open Source Software.

Open Source Software means any software that requires as a condition of use, modification and/or distribution of such software that such software or other software incorporated into, derived from or distributed with such software (a) be disclosed or distributed in Source Code form; (b) be licensed by the user to third parties for the purpose of making and/or distributing derivative works; or (c) be redistributable at no charge. Open Source Software includes, without limitation, software licensed or distributed under any of the following licenses or distribution models, or licenses or distribution models substantially similar to any of the following: (a) GNU's General Public License (GPL) or Lesser/Library GPL (LGPL), (b) the Artistic License (e.g., PERL), (c) the Mozilla Public License, (d) the Netscape Public License, (e) the Sun Community



Source License (SCSL), (f) the Sun Industry Source License (SISL), and (g) the Common Public License (CPL).

SINGLE USER LICENSE. You may copy the Software onto a single computer for your personal, noncommercial use, and you may make one back-up copy of the Software, subject to these conditions:

1. This Software is licensed for use only in conjunction with Intel component products. Use of the Software in conjunction with non-Intel component products is not licensed hereunder.
2. You may not copy, modify, rent, sell, distribute or transfer any part of the Software except as provided in this Agreement, and you agree to prevent unauthorized copying of the Software.
3. You may not reverse engineer, decompile, or disassemble any part of the Software not provided to you in source code form.
4. You may not sublicense or permit simultaneous use of the Software by more than one user.
5. The Software may include portions offered on terms in addition to those set out here, as set out in a license accompanying those portions.

OEM LICENSE: You may reproduce and distribute the Software only as an integral part of or incorporated in Your product or as a standalone Software maintenance update for existing end users of Your products, excluding any other standalone products, subject to these conditions:

1. This Software is licensed for use only in conjunction with Intel component products. Use of the Software in conjunction with non-Intel component products is not licensed hereunder.
2. You may not copy, modify, rent, sell, distribute or transfer any part of the Software except as provided in this Agreement, and you agree to prevent unauthorized copying of the Software.
3. You may not reverse engineer, decompile, or disassemble any part of the Software not provided to you in source code form.
4. You may only distribute the Software to your customers pursuant to a written license agreement. Such license agreement may be a "break-the-seal" license agreement. At a minimum such license shall safeguard Intel's ownership rights to the Software.
5. The Software may include portions offered on terms in addition to those set out here, as set out in a license accompanying those portions.

ISV LICENSE: You may reproduce, distribute and the Software only as an integral part of or incorporated in Your product or as a standalone Software maintenance update for existing customers of Your products such as your OEM customers, excluding any other standalone products, subject to these conditions:



1. This Software is licensed for use only in conjunction with Intel component products. Use of the Software in conjunction with non-Intel component products is not licensed hereunder.
2. You may not copy, modify, rent, sell, distribute or transfer any part of the Software except as provided in this Agreement, and you agree to prevent unauthorized copying of the Software.
3. You may not reverse engineer, decompile, or disassemble any part of the Software not provided to you in source code form.
4. You may only or distribute the Software to your customers pursuant to a written license agreement. At a minimum such license shall safeguard Intel's ownership rights to the Software.
5. The Software may include portions offered on terms in addition to those set out here, as set out in a license accompanying those portions.

OPEN SOURCE SOFTWARE. In the event Open Source Software is included with Licensed Software, such Open Source Software is licensed pursuant to the applicable Open Source Software license agreement identified in the Open Source Software comments in the applicable source code file(s) and/or File Header provided with Licensed Software. Additional detail may be provided (where applicable) in the accompanying on-line documentation. With respect to the Open Source Software, nothing in this Agreement limits any rights under, or grants rights that supersede, the terms of any applicable Open Source Software license agreement.

NO OTHER RIGHTS. No rights or licenses are granted by Intel to You, expressly or by implication, with respect to any proprietary information or patent, copyright, mask work, trademark, trade secret, or other intellectual property right owned or controlled by Intel, except as expressly provided in this Agreement.

OWNERSHIP OF SOFTWARE AND COPYRIGHTS. Title to all copies of the Software remains with Intel or its suppliers. The Software is copyrighted and protected by the laws of the United States and other countries, and international treaty provisions. You may not remove any copyright notices from the Software. Intel may make changes to the Software, or to items referenced therein, at any time without notice, but is not obligated to support or update the Software. Except as otherwise expressly provided, Intel grants no express or implied right under Intel patents, copyrights, trademarks, or other intellectual property rights. You may transfer the Software only if the recipient agrees to be fully bound by these terms and if you retain no copies of the Software.

LIMITED MEDIA WARRANTY. If the Software has been delivered by Intel on physical media, Intel warrants the media to be free from material physical defects for a period of ninety days after delivery by Intel. If such a defect is found, return the media to Intel for replacement or alternate delivery of the Software as Intel may select.

EXCLUSION OF OTHER WARRANTIES. EXCEPT AS PROVIDED ABOVE, THE SOFTWARE IS PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. Intel does not warrant or assume responsibility for the



accuracy or completeness of any information, text, graphics, links or other items contained within the Software.

LIMITATION OF LIABILITY. IN NO EVENT SHALL INTEL OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, OR LOST INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF INTEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS PROHIBIT EXCLUSION OR LIMITATION OF LIABILITY FOR IMPLIED WARRANTIES OR CONSEQUENTIAL OR INCIDENTAL DAMAGES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU. YOU MAY ALSO HAVE OTHER LEGAL RIGHTS THAT VARY FROM JURISDICTION TO JURISDICTION.

TERMINATION OF THIS AGREEMENT. Intel may terminate this Agreement at any time if you violate its terms. Upon termination, you will immediately destroy the Software or return all copies of the Software to Intel.

APPLICABLE LAWS. Claims arising under this Agreement shall be governed by the laws of California, excluding its principles of conflict of laws and the United Nations Convention on Contracts for the Sale of Goods. You may not export the Software in violation of applicable export laws and regulations. Intel is not obligated under any other agreements unless they are in writing and signed by an authorized representative of Intel.

GOVERNMENT RESTRICTED RIGHTS. The Software is provided with "RESTRICTED RIGHTS." Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor. Use of the Software by the Government constitutes acknowledgment of Intel's proprietary rights therein. Contractor or Manufacturer is Intel 2200 Mission College Blvd., Santa Clara, CA 95052.

§



8.0 Glossary

Table 6. Glossary

ABSF	As-built Boot Settings File Similar to the <i>BSF</i> but also contains the configuration values used to patch the binary file. See the <i>Basics</i> section for contextual information regarding the ABSF.
BCT	Binary Configuration Tool
BSF	<p>Boot Settings File</p> <p>Contains a description of the configurable settings embedded in a binary file. This description includes:</p> <ul style="list-style-type: none"> • default value • help message describing the data • constructs used to represent a graphical interface to manipulate the data • location and size of the data in the binary file <p>See the <i>Basics</i> section for contextual information regarding the BSF.</p>
FSP	<p>Firmware Support Package</p> <p>Provides an interface that initializes silicon components (CPU, chipset, and memory).</p>
GUI	Graphical User Interface
Rebase	To change the binary file's load address