

H2OFFT™ (Flash Firmware Tool)

User Guide for EFI Version

Rev. 1.3 Approved
Prepared by DR-1 Team
June 07, 2018
Insyde Software Corp.



Copyright © 1998-2018, All Rights Reserved.
Insyde Software Corp.


No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form, or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of Insyde Software Corp.

Disclaimer

Insyde Software Corp. provides this document and the programs "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose.

This document could contain technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in future revisions of this document. Insyde Software Corp. is under no obligation to notify any person of the changes.

The following trademarks are used in this document:

 and InsydeH2O are registered trademarks of Insyde Software Corp.

All other trademarks or trade names are property of their respective holders.

Revision History

Rev.	Date	Author	Approver	Summary
1.1o	05/25/2012	Kevin Liang	<Approver>	Initial version
1.2d	08/10/2012	Kevin Liang	<Approver>	Update command & add platform.ini & ifdpacker
1.2g	09/26/2012	Kevin Liang	<Approver>	Update command & add platform.ini & ifdpacker
1.2h	10/16/2012	Kevin Liang	<Approver>	Update command & ifdpacker & support PFAT image update
1.2i	01/15/2013	Kevin Liang	<Approver>	Update command & platform.ini & ifdpacker version
1.2k	02/20/2013	Kevin Liang	<Approver>	Update ifdpacker version
1.2l	05/09/2014	Jerry Yen		Update commands
1.3	06/07/2018	Daniel Chang		Update Commands, platform.ini and ifdpacker

Table of Contents

1.Introduction	6
1.1.Overview	6
1.2.Support Functionality	6
1.3.Support Requirements.....	6
1.4.Member	6
2.Using H2OFFT-S	8
2.1.Option /h — Print help.....	9
2.2.Option /ab — Check battery life percent.....	9
2.3.Option /ac — Do not check AC plug in	9
2.4.Option /acb — Check AC plug in & battery percentage	9
2.5.Option /all — Flash all	9
2.6.Flash BIOS protect area.....	9
2.7.Flash which region that describe in descriptor table	10
2.8.Option /dc — Disable comparison in normal flash process.....	10
2.9.Option /di — Disable ID display.....	10
2.10.Option /e: — Update fix size from file offset to physical address	10
2.11.Option /ecp — Flash Non-share EC with percentage	10
2.12.Option /ecbp — Flash Non-share EC with BIOS and show percentage.....	10
2.13.Option /edt — updates customize data.....	11
2.14.Option /eob — Only flash BIOS when EC and BIOS are merged in the same file	11
2.15.Option /extec: — Update external EC region.....	11
2.16.Option /nativeme: — Update ME region with native ME file.....	11
2.17.Option /g — Save current BIOS to file (from IHISI).....	11
2.18.Option /mc — Skip all platform model check	12
2.19.Option /mfg — Manufacture mode	12
2.20.Option /OemCus — OEM customize token	12
2.21.Do which action after flash	12
2.22.Option /pbi — Flash BIOS protect region	12
2.23.Option /pi — Dump BVDT protection MAP (Need BIOS support)	12
2.24.Option /pq — Query ROM protection MAP in current ROM	12
2.25.Option /pr — Query region MAP in current ROM.....	13
2.26.Option /priv — Query ROM private MP in current ROM.....	13
2.27.Option /pw — Query whole ROM MAP.....	13
2.28.Option /rt — When SMI error, retry how many times	13
2.29.Option /u — Show confirmation message	13

2.30.Option /vrt — When verifying error, retry how many times.....	13
3.Customizing H2OFFT-S	14
3.1.[AC_Adapter]	15
3.2.[Bios_Version_Check]	16
3.3.[BIOSVersionFormat]	17
3.4.[CommonFlash]	18
3.5.[FlashComplete]	19
3.6.[FlashSecureBIOSOverride]	20
3.7.[ForceFlash]	21
3.8.[Log_file]	22
3.9.[MessageStringTable]	23
3.10.[MULTI_FD]	24
3.11.[Others]	26
3.12.[Platform_Check]	27
3.13.[PlatformVersion]	28
3.14.[Region]	29
3.15.[SecureUpdate]	30
3.16.[UI]	31
3.17.[UpdateEC]	32
3.18.[UpdateOEMME]	33
3.19.[PassToBios]	34
4.Using iFDPacker	35
4.1.Pack H2OFFT	35
4.1.1 Pack H2OFFT steps.....	35
4.1.2 Packer support command list	36
5.Support for BIOS Guard (PFAT) image update	37
5.1.How to sign a BIOS Guard (PFAT) BIOS image?	37
6.FAQ	38
6.1 Which configure file (platform.ini) will be referenced if the configure file built in capsule image and also exist external folder?	38

1. Introduction

1.1. Overview

H2OFFT-S is a flash utility provided by Insyde Software. It serves as a powerful and intelligent tool for updating and maintaining the computer system BIOS under the EFI Shell environment. H2OFFT-S also features a friendly user interface for saving, loading and updating the BIOS, as well as displaying the system BIOS information. This chapter provides a quick introduction on H2OFFT-S, its key features, and what it can do for you.

1.2. Support Functionality

H2OFFT-S offers you the following functionality:

- Allows to easily update the system BIOS under EFI Shell environment (Legacy Flash) and secure flash.
- Allows to verify the system BIOS to ensure system reliability.
- Auto detects hardware settings to help determine H2OFFT-S system compatibility.

1.3. Support Requirements

Installing H2OFFT-S is quick and easy. However, you need to be aware of its system requirements before using the program:

- EFI Shell environment.
- InsydeH2O® BIOS-compatible motherboard (H2OFFT-S will auto detect the BIOS on motherboard for compatibility).
- InsydeH2O® BIOS that supports Insyde IHISI.
- On secure boot supported system, the shell flash utility (H2OFFT-Sx64.efi) must be signed via iEFIFlashSigner.exe. Please refer “ReadMe.txt” in security flash package to know how to sign by iEFIFlashSigner.exe.

1.4. Member

File	Description
H2OFFT-S.efi	The main execution file that supports x86 systems.
H2OFFT-Sx64.efi	The main execution file that supports x64 systems.
platform.ini	Flash utility option configuration file.
Doc\ReleaseNote.txt	Release history.
Doc\H2OFFT_UserGuide_EFI_VX.X.pdf	User guide of this utility.

Tools\iFdPacker.exe	Single-execution package tool. It can package the H2OFFT-Sx64.efi/H2OFFT-S.efi and BIOS binary file into single file to run.
Tools\PlatformIniEditor.exe	The editor for platofrm.ini.
Tools\platform.xml	The configuration file of PlatformIniEditor.

2. Using H2OFFT-S

Option	Description
-h	This flash utility help.
-ab	Check battery life percentage.
-ac	Do not check AC plug in.
-acb	Check AC plug in & battery percentage.
-all	Force flash all protected regions.
-b	Flash PEI volume.
-bios	Flash BIOS region.
-dc	Disable comparison in normal flash process.
-desc	Flash DESC region.
-di	Disable ID display.
-e: Offset, Size, Address	Update fix size from file offset to physical address.
-ecbp	Flash EC with BIOS with percentage.
-ecp	Flash EC with percentage.
-edt#@:"VALUE"	Update customizes data (such as logo with signature) by IHISI.
-eob	Only flash BIOS when EC and BIOS are merged in the same file.
-exec:FILE	Update external EC region with input file.
-fd	Force flash protected DXE region.
-fe	Force flash protected share EC region.
-fl	Force flash protected Logo region.
-fm	Force flash protected CPU Microcode region.
-fn	Force flash protected OEM NVS region.
-fp	Force flash protected password region.
-ft:TYPE	Force flash protected OEM special type region.
-fv	Force flash protected Variable region.
-g	Save current BIOS to file (from IHISI).
-gbe	Flash GBE region.
-iv	Show tool and bios support IHISI version.
-lg:GUID	Input GUID.
-mc	Skip all platform model check (model name and BIOS version check).
-me	Flash ME region.
-mfg	Run in manufacture mode.
-n	Do not reboot after flash.
-nativeme:FILE	Update Intel native ME with input file.
-OemCus	Passing specific string to BIOS at program start to do specific action in BIOS.
-oemid:GUID	OEM ID for update Intel native ME, it only valid when –nativeme: used.
-pbi:Type(Hex)	Flash BIOS protect region. Type is the protected region type.
-pdr	Flash platform data region.
-pi	Dump BVDI protection MAP.
-pq	Query ROM protection MAP in current ROM.
-pr	Query external region MAP in current ROM.
-priv	Query ROM private MAP in current ROM.

-pw	Query whole ROM MAP.
-r	Reboot after flash.
-rt:TIMES,DELAY	When SMI error, retry times and delay milliseconds between each retry.
-s	Shutdown after flash.
-u	Show confirm message.
-vrt:Times	When verify error, retry how many times

2.1. Option /h — Print help

H2OFFT-S -h

All of the tools contain help descriptions to provide guidance in how to use them. This feature will list all options in this tool on screen, allowing users to search through it.

2.2. Option /ab — Check battery life percent

H2OFFT-S (filename) -ac -ab

This feature needs to use the option “ac” to skip AC check when the AC is not plugged in (only use when AC is not plugged in and with -ac). This feature will check the battery percentage remaining. It must be more than the value that is set in BIOS.

2.3. Option /ac — Do not check AC plug in

H2OFFT-S (filename) -ac

Before flashing the BIOS, the tool will check if the AC power is plugged in. If it is not, the tool will not continue to flash and pop a message to inform the user. If you do not want to check it, please use this feature.

2.4. Option /acb — Check AC plug in & battery percentage

H2OFFT-S (filename) -acb

Before flashing BIOS, the tool will check AC plug-in & battery percentage.

2.5. Option /all — Flash all

H2OFFT-S (filename) -all

BIOS will report to the tool about which areas will be protected in the BIOS region. User can use this command to flash all BIOS regions.

2.6. Flash BIOS protect area

Since there are many protecting areas in the BIOS region, you can utilize these options to force to update these protecting areas.

H2OFFT-S.efi (filename) /b	Force to update the PEI area.
/fd	Force to update the DXE area.
/fe	Force to update the EC area.
/fl	Force to update the Logo area.

/fm	Force to update the CPU Microcode area.
/fp	Force to update the password area.
/fn	Force to update the OEM NVS area.
/ft:TYPE	Force to update the specific type area.
/fv	Force to update the variable area.

2.7. Flash which region that describe in descriptor table

H2OFFT-S (filename) -(desc, me, ec, gbe, bios, pdr)

In descriptor mode, there may have more than two regions of the ROM. These commands will provide you the ability to flash the regions that you want to flash.

Note: These commands support on Intel platform only.

2.8. Option /dc — Disable comparison in normal flash process

H2OFFT-S (filename) -dc

During flash, BIOS tool will read the current block from BIOS and new block from file, and then compare them. If they are the same, tool will skip and will not flash those blocks. This command will skip this compare feature.

2.9. Option /di — Disable ID display

H2OFFT-S (filename) -di

Tool will show the name of this platform, and this feature will disable the name display.

2.10. Option /e: — Update fix size from file offset to physical address

H2OFFT-S (filename) -e:offset(Hex),size(Hex),address(Hex)

This command will update fix size from file offset to physical address.

2.11. Option /ecp — Flash Non-share EC with percentage

H2OFFT-S (filename) -ecp

This command will flash EC with percentage.

2.12. Option /ecbp — Flash Non-share EC with BIOS and show percentage

H2OFFT-S (filename) -ecbp

Flash non-share ROM EC with new BIOS file, so “filename” is the EC and the BIOS merges the filename.

Because this feature sends SMI many times, this EC is not flashed by SMI one time. To learn more about this SMI, please read IHISI 1.8.0 or the highest version of the IHISI spec.

2.13. Option /edt — updates customize data

H2OFFT-S (filename) -edt:#@:"value"

You can use -edt#@:"Value" for updating customized data (such as logo with signature) by IHISI.

- ❑ # — from 4 ~ C.
- ❑ @ — F, S, W, DW
 - ❑ F — means file
 - ❑ S — means string
 - ❑ W — means word value
 - ❑ DW — means double word value

Example:

Update type 4 data, the source is file.

And update type 5 data, the source is string.

- -edt4f:logo.jpg -edt5s:"Input string"

Update a type 9 data, the source is WORD.

- -edt9w:"0x1234"

Update a type C data, the source is DWORD

- -edtcdw:"0x12345678"

2.14. Option /eob — Only flash BIOS when EC and BIOS are merged in the same file

H2OFFT-S (filename) -eob

The file is merged by EC and BIOS. This feature will flash BIOS and skip EC.

2.15. Option /extec: — Update external EC region

H2OFFT-S -extec:(filename)

Update external EC region with input file.

Note: This command supports on Intel platform only.

2.16. Option /nativeme: — Update ME region with native ME file

H2OFFT-S -nativeme:FILENAME [-oemid:GUID]

Update Intel ME region with input native ME file.

It allow to input an OEM ID via -oemid: command.

Note: This command supports on Intel platform only.

2.17. Option /g — Save current BIOS to file (from IHISI)

H2OFFT-S (filename) -g

This feature allows you to read the BIOS from IHISI to a file.

2.18. Option /mc — Skip all platform model check

H2OFFT-S (filename) -mc

This tool will check the platform model for the version and the platform name that are reported from BIOS. If there do not match, the tool will show an error message and leave. If you don't want to use this feature, you can use this command to do it. As a result, the tool will not show the name and version.

2.19. Option /mfg — Manufacture mode

H2OFFT-S (filename) -mfg

Application notifies BIOS that current system is in manufacturing mode. BIOS can do some special process while in manufacturing mode.

2.20. Option /OemCus — OEM customize token

H2OFFT-S (filename) -oemcus CUSXXX

This command can pass the following string token to BIOS at program start. BIOS can use it to do some specific action.

The CUSXXX means the string start with “CUS”, for example CUSV1, CUSDEBUG, CUSSKU2.

Note: One command only can follow one token.

When there is more than one token need to pass to BIOS, you can use “-oemcus CUSTOKEN1 -oemcus CUSTOKEN2”.

2.21. Do which action after flash

H2OFFT-S (filename) -(r, s, n)

After flash, the BIOS tool will call ec to do the following three things; “reboot”, “shutdown”, and “nothing to do”. The default is to reboot. If this is not your desired behavior, you can use “-n” to prevent system reboot or use “-s” to tell tool to shutdown after flash.

2.22. Option /pbi — Flash BIOS protect region

H2OFFT-S (filename) -pbi:Type(Hex)

You can update partial region base on protecting map. Type the protected region type(Hex).

2.23. Option /pi — Dump BVDT protection MAP (Need BIOS support)

H2OFFT-S (filename) -pi

Tool will dump the BVDT protection map in new BIOS and current BIOS.

2.24. Option /pq — Query ROM protection MAP in current ROM

H2OFFT-S -pq

Dump the protection map of BIOS region in current ROM.

2.25. Option /pr — Query region MAP in current ROM

H2OFFT-S -pr

Dump the region map of BIOS region in current ROM.

2.26. Option /priv — Query ROM private MP in current ROM

H2OFFT-S -priv

Dump the private map of BIOS region in current ROM.

2.27. Option /pw — Query whole ROM MAP

H2OFFT-S -pw

Dump the whole region map which defined in current ROM.

2.28. Option /rt — When SMI error, retry how many times

H2OFFT-S -rt:Times,Delay

When SMI error occurs, this lets you set the number of retry attempts and the delay in milliseconds between each retry attempt.

2.29. Option /u — Show confirmation message

H2OFFT-S (filename) -u

Application will ask if you want to flash before performing it.

2.30. Option /vrt — When verifying error, retry how many times

H2OFFT-S -vrt:Times

When verifying error, retry the set amount of times.

3. Customizing H2OFFT-S

Supported Configuration Section List

- | | |
|-----|---------------------------|
| 1. | [AC_Adapter] |
| 2. | [Bios_Version_Check] |
| 3. | [BIOSVersionFormat] |
| 4. | [CommonFlash] |
| 5. | [FlashComplete] |
| 6. | [FlashSecureBIOSOverride] |
| 7. | [ForceFlash] |
| 8. | [Log_file] |
| 9. | [MessageStringTable] |
| 10. | [MULTI_FD] |
| 11. | [Others] |
| 12. | [Platform_Check] |
| 13. | [PlatformVersion] |
| 14. | [Region] |
| 15. | [SecureUpdate] |
| 16. | [UI] |
| 17. | [UpdateEC] |
| 18. | [UpdateOEMME] |
| 19. | [PassToBios] |

3.1. [AC_Adapter]

To do AC/DC check before firmware update.

Flag	(default) 0: 1:	Don't check AC. Check AC.
BatteryCheck	(default) 0: 1:	Don't check battery. Check battery.
BatteryBound	1~100 (default) 20	Low battery boundary (percentage). When BatteryCheck=1 this value will be referenced. And only when the battery life percentage is bigger than inputted value, it can do flash.
LauncherAcWarning	(default) A02	A key name which list in [MessageStringTable].
SecurityAcWarning	(default) A02	A key name which list in [MessageStringTable].

3.2. [Bios_Version_Check]

To do firmware version check before update firmware.

Flag	0: Don't check rom file version. 1: Check BIOS version. When rom file version is older than BIOS, it will displa a warning message and close application. (default) 2: Depend on BIOS report.
CheckByBios	(default) 0: Normal process. 1: It will pass version by IHISI to BIOS and check by BIOS. When BIOS return not allow to flash, it will be terminated the process. When BIOS allow to flash, it will go ahead and do its normal process. When this flag is enable but BIOS not support, it will skip version check and assume allow to flash.
CheckByBiosErrorMessage	"This BIOS file is not allow to flash. The flash process will be terminated." (default) String: User defined error message when BIOS is not allow to flash this version image.

3.3. [BIOSVersionFormat]

The below configuration for firmware version format is used to define version format check.

BIOSVFEEnable	(default) 0: 1:	Disabled. Enabled.
VersionFormat	[X] [N] [T] [.] [D]	The field is masked. It will NOT be compared. The digit field can be ASCII, case-sensitive. It's the same definition with N. But T is a case-insensitive. Dot is also a mask. It will NOT be compared. Don't care field. It will NOT be compared. This field can be empty. It only allow to put at start or end of the version format. For example: Onboard version 1.21B flash to 1.22, VersionFormat must be N.NND N.NND means the valid format is N.NN and N.NNX, the 5th character will be ignore in version compare.

3.4. [CommonFlash]

This feature is only available for specific ODM.

Flag	(default) empty	A switch flag setting string. Ex: "CPVER:[1] ACEN DCEN FHRST" Detail parameter please reference following table.	
		Parameter	Description
		PTEN	All protection enable.
		PTDIS	All protection disable.
		ACEN	AC protect checking enable.
		ACDIS	AC protect checking disable.
		DCEN	DC & Gangue protect checking enable.
		DCDIS	DC & Gangue protect checking disable.
		RESSEN	BIOS Regression enable.
		RESSDIS	BIOS Regression disable.
		PJMDEN	Project Model string protect checking enable.
		PJMDDIS	Project Model string protect checking disable.
		FHOS	System back to OS after flash BIOS completely.
		FHST	System directly shutdown after flash BIOS completely.
		FHRST	System directly reboot after flash BIOS completely.
		CPVER:[Num]	Common Flash Version information ex: [Num] is decimal and start from 1.
ErrorMsg00	(default) empty	No error message.	
ErrorMsg01	(default) empty	AC error message.	
ErrorMsg02	(default) empty	DC error message.	
ErrorMsg03	(default) empty	DC gas gauge under xx% message.	
ErrorMsg04	(default) empty	BIOS version error message.	
ErrorMsg05	(default) empty	Model name error message.	
ErrorMsg10	(default) empty	No support this version of Flash Common Interface message.	
ErrorMsg##	(default) empty	The number ## is in hex.	

3.5. [FlashComplete]

Below configuration is to override default action and to apply optional action in end of flash when launching in normal flash mode.

Action	0: Do nothing. 1: Shutdown. (default) 2: Reboot.
Pause	(default) 0: Disable to pause after flash complete. 1: Enable to pause after flash complete.
PauseWarning	default: messagestringA03 A key name which list in [MessageStringTable].

3.6. [FlashSecureBIOSOverride]

EnableFlashSecureBIOSOverride	(default) 0: Disable action override. Use the action which returned from BIOS. 1: Enable the action override when flashing secure BIOS in OS.
Action	0: S3 (default) 1: Reboot. 2: Shutdown. 3: Do nothing.

3.7. [ForceFlash]

ALL	(default) 0: 1:	Reserve all protected areas. Flash all protected areas.
BB_PEI CPU_Microcode Variable DXE EC Password OEM_NVS Logo	(default) 0: 1:	Protect this area. Force flash this area. They are the predefined types each one indicate a type value. BB_PEI = Type#00 CPU_Microcode = Type#01 Variable = Type#02 DXE = Type#03 EC = Type#04 Password = Type#0F OEM_NVS = Type#10 Logo = Type#05
Type#NN	(default) 0: 1:	Protect this area. Force flash this area. Type#NN is an extended method for those didn't have predefine type. The NN is a number in Hex. For example: If BIOS report to protect type 13h from IHISI, and you want to flash this protect area. Then the setting in platform.ini can add Type#13=1.

3.8. [Log_file]

Enable it will generate a log file for debugging purposes.

Flag	(default) 0: 1:	Don't log to file. Utility will log to specify file.
FileName	(default) H2OFFT.log	Log file name string.

3.9. [MessageStringTable]

Provide message string table to define customize message.

messagestring1 messagestringA00 messagestringA01 messagestringA02 messagestringA03 messagestringA04	(default) empty	The message string must as following format messagestring#="Your message here." The # is a number in Decimal or Hex. If a multi-line message is required, you can use "\n" in message string for new line.
--	------------------------	--

3.10. [MULTI_FD]

Some of flash package would contain multiply firmware image for different SKU. Following setting can be configured what condition is to detect firmware image.

Flag	(default) 0: Disable 1: Enable
FD#XX	FD#01 ~ FD#99 XX is decimal number from 01 to 99. The setting is a string with following format: condition type, condition

<u>condition of IO :</u> IO ,[Offset], [Mask], [Value], [File Name], [ME File Name], [INI File Name]	Offset in hex. IO type supports BYTE, WORD and DWORD in hex. IO type supports BYTE, WORD and DWORD in hex. File name of FD. ME File name of FD. If it exists, utility will run OEMME flash feature. INI File name for overwrite.
---	---

<u>condition of PCI :</u> PCI ,[Bus], [Device], [Function], [Offset], [Mask], [Value], [File Name], [ME File Name], [INI File Name]	Bus number Device number Function number Offset in hex. PCI type supports DWORD in hex only. PCI type supports DWORD in hex only. File name of FD. ME File name of FD. If it exists, utility will run OEMME flash feature. INI File name for overwrite.
---	---

<u>condition of ID :</u> ID ,[Model Name], [File Name], [ME File Name], [INI File Name]	The platform ID, model name string. File name of FD. ME File name of FD. INI File name for overwrite.
--	--

<u>condition of OS :</u> OS ,[OS Version], [File Name], [ME File Name], [INI File Name]	32bit or 64bit OS. 32 for 32bit OS, 64 for 64bit OS. File name of FD. ME File name of FD. INI File name for overwrite.
--	---

<u>condition of MEMORY :</u> MEMORY ,[Physical Address], [Mask], [Value], [File Name], [ME File Name], [INI File Name]	A DWORD value in hex. MEMORY type supports BYTE, WORD and DWORD in hex. MEMORY type supports BYTE, WORD and DWORD in hex. File name of FD. ME File name of FD. INI File name for overwrite.
---	--

<u>condition of MPCIO :</u> MPCIO ,[Condition Number],	Condition type number
--	-----------------------

PCI-[Bus]-[Device]-[Function]-[Offset]-[Mask]-[Value], IO-[Offset]-[Mask]-[Value], [File Name], [ME File Name], [INI File Name]	Condition of PCI Condition of IO File name of FD. ME File name of FD. INI File name for overwrite.
---	--

3.11. [Others]

The below setting is to detect whether to automatically or manually start to enter flash process.

DisableSecureCapsuleFlash	(default) 0: 1:	Enable flash secure BIOS on normal platform. Disable flash secure BIOS on normal platform.
----------------------------------	---------------------------	---

3.12. [Platform_Check]

Perform platform name check before firmware update.

Flag	0: Don't check project ID. 1: Check project ID of new file. If ID is different with current BIOS, the utility will close. 2: Utility will compare current platform ID with the 20 platform IDs. If anyone is match, it will go ahead, otherwise utility will close. (default) 3: Depends on BIOS report.
PlatformName1~20	(default) empty String: If ROM file do not contain correct ID, user can define ID here.

3.13. [PlatformVersion]

This flag only available when the [Platform_Check] is enable to compare the 20 platform IDs.

The Version is pair with the PlatformName.

For example: When the platform ID matches with PlatformName2, the Version2 will be used.

Flag	(default) 0: 1:	Don't use multi version. Use the version in the list instead of the version in file.
Version1~20	(default) empty	If ROM file do not contain correct version, user can define version here.

3.14. [Region]

This section is used to update region of Intel firmware.

Default is flash all regions when the values all set to 0.

If any one of the regions set to 1, it will only flash specific regions.

If the BIOS is built without additional Intel firmware as like ME, GbE and Descriptor or BIOS is an AMD firmware which does not support ME, please ignore this section.

DESC	(default) 0: Don't flash. 1: Flash Descriptor region.
GbE	(default) 0: Don't flash. 1: Flash GbE region.
ME	(default) 0: Don't flash. 1: Flash ME region.
EC	(default) 0: Don't flash. 1: Flash EC region.
BIOS	(default) 0: Don't flash. 1: Flash BIOS region.
Platform_Data	(default) 0: Don't flash. 1: Flash Platform Data region.

3.15. [SecureUpdate]

In secure flash mode, we need somewhere to temporarily save the secure flash capsule. The below flag is to decide whether the capsule is put in ESP or default is put memory space.

viaESP	(default) 0: 1:	Disable. Write the capsule to ESP (EFI system partition).
DeviceOrder	(Default)	eMMC, NVMe, SATA, ATAPI, USB The FAT device detection sequence of secure flash via ESP feature. Now we support eMMC, NVMe, SATA, ATAPI, USB.

3.16. [UI]

Below configuration is provided to user decide what information is present and what action is applied in user interface.

Elapse	(default) 0: 1:	Disable to show elapse time during progressing BIOS update. Enable to show elapse time during progressing BIOS update.
--------	--------------------	---

3.17. [UpdateEC]

This configuration is configured for EC update.

EC_DockWarning		messagestringA04 (default) A key name which list in [MessageStringTable].
----------------	--	--

3.18. [UpdateOEMME]

The Intel firmware update tool (FWUpdLcl.exe) is dependent on each chipset generation, and the tool in release flash package is a sample and may be not suitable for your project.

Please remember to replace the FWUpdLcl.exe with right version before you will utilize function to update Intel firmware (ME or TXE).

MEFileName		<p>empty (default)</p> <p>If this file name or Multi-FD ME file name exist tool will run this case to flash ME.</p>
Command		<p>empty (default)</p> <p>When this field is empty and don't want to check ME version, utility will use "-f %filename -generic -allowsv" as default command. The %filename is a keyword which will be replaced with the value in MEFileName within this section or the filename in MULTI_FD section.</p>

3.19. [PassToBios]

The settings in this section will by pass to BIOS. Tool won't do any action on them.

ClearTXE	(default) 0: 1:	Disable. Tell BIOS to clear TXE at this flash.
----------	--------------------	---

4. Using iFDPacker

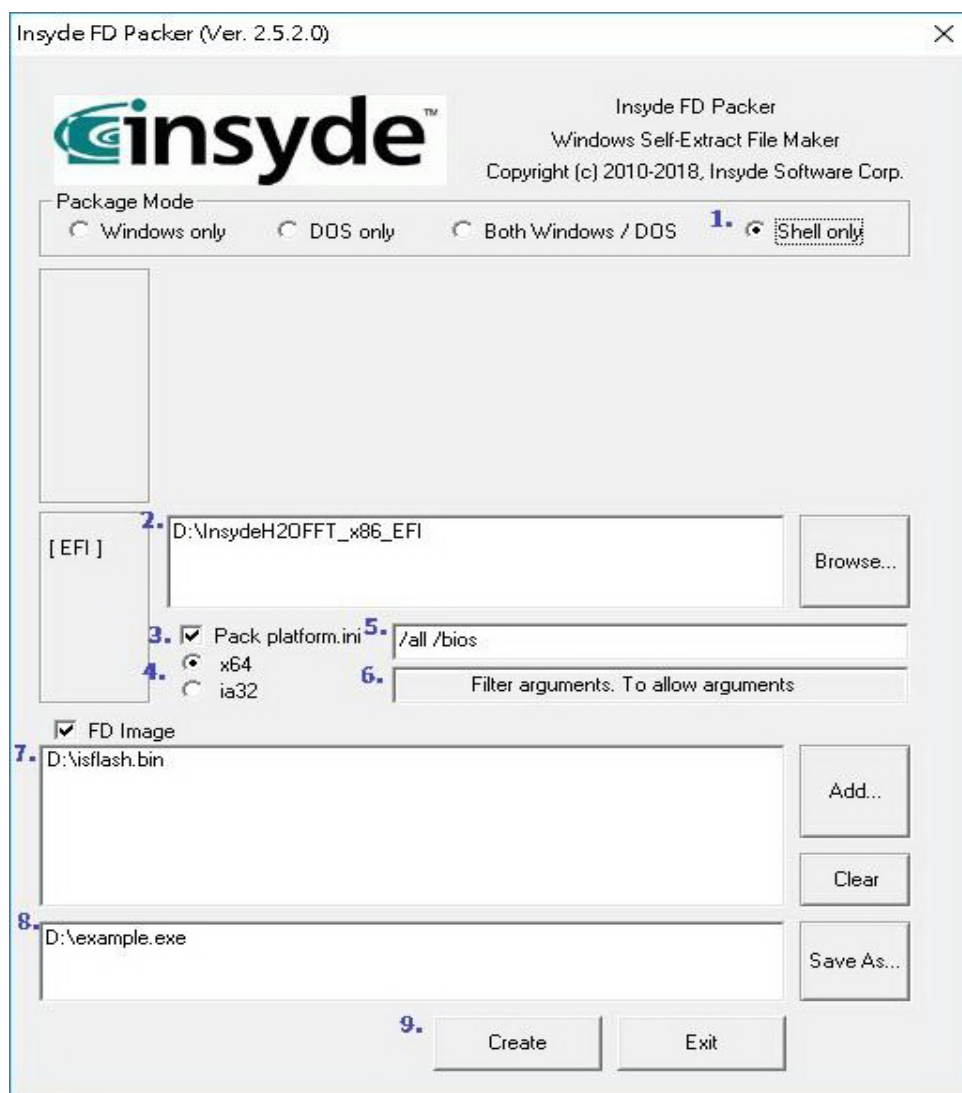
4.1. Pack H2OFFT

4.1.1 Pack H2OFFT steps

Locate “iFdPacker.exe” in the Packer folder. You can use this tool to package H2OFFT-S and binary files with the setting in Platform.ini.

In addition to the H2OFFT-S utility, BIOS ROM files, etc., the “iFdPacker.exe” is the InsydeFdPacker utility for you to generate an executable file.

Additionally, you can use InsydeFdPacker to pack the entire H2OFFT-S folder into an executable file. To use InsydeFdPacker, follow these procedures:



1. Select Package mode item.
2. Click [Browse] to select directory of H2OFFT-S package.
3. Checked the check box if using external configure file.

4. Select EFI mode.
5. Input default argument which will auto be used when packaged execution file launched.
6. Input filter argument which will allow to be used with packaged execution file.
7. Click [Add] to select BIOS image.
8. Click [Save As] to select output file folder and specify a file name.
9. Click [Create] to pack.

4.1.2 Packer support command list

You can use "iFdPacker -h" to see the usage

Command	Description
-winsrc PATH	The path of WinFlash
-dossrc PATH	The path of DosFlash
-shlsrc PATH	The path of ShellFlash
-b [3264 32 64]	The WinFlash Build Type 3264 - 32bit Ap runs on 32/64bit OS 32 - 32bit Ap runs on 32bit OS 64 - 64bit Ap runs on 64bit OS
-winini	Windows pack platform.ini
-dosini	Dos pack platform.ini
-shlini	Shell pack platform.ini
-protect	Protect packed package that will not allow to unzip or modify with third-party program.
-winarg "flag"	Windows argument with quotation marks
-arg "flag"	Dos/Shell argument with quotation marks
-argfilter "flag"	Dos/Shell argument filter with quotation marks
-fv FILE	The path of firmware file
-output FILE	The single package file
-h	The usage message

Example:

```
iFdPacker.exe -shlsrc D:\InsydeH2OFFT_x86_EFI
               -shlini
               -b 64
               -fv d:\isflash.bin
               -arg "-bios -all"
               -argfilter "-ac"
               -output D:\output.efi
```


5. Support for BIOS Guard (PFAT) image update

5.1. How to sign a BIOS Guard (PFAT) BIOS image?

Support for PFAT image update requires BIOS version: SharkBay 03.72.37.0018.

1. System Requirements:
 - Microsoft Windows 7 or later.
 - Microsoft "SignTool.exe". (Included in the Microsoft Windows SDK package. v6.2.8229.0 in Win8 SDK or later).
 - Make sure the SignTool.exe is in the System Environment Variable "path".
2. Please install QA Certificate in your system. (Reference QA Certificate Installation Guide) You need QA.pfx file and double-click to install it into your system.
3. Use iEFIFlashSigner.exe to sign PFAT BIOS image in command mode. (Output file name: isflash.bin)

Example:

Sign PFAT bios image only command:

```
iEFIFlashSigner.exe -n "QA Certificate." -bios BIOS_PFAT.fd
```

(PS: BIOS_PFAT.fd is PFAT BIOS image file name)

6. FAQ

6.1 Which configure file (platform.ini) will be referenced if the configure file built in capsule image and also exist external folder?

External configure file is higher priority,

- **In secure update**

H2OFFT will reference the platform.ini in the same folder instead of configuration of secure FD (isflash.bin).

- **In package mode**

H2OFFT will reference the external configure file, if click “pack platform.ini”.

If there is platform.ini in the same folder with package file (package01.exe), configure file will not be referenced.