



HT32 Flash Commander User Guide

Revision: V1.10 Date: October 16, 2023

www.holtek.com

Table of Contents

Introduction	3
Functional Description.....	3
Usage Description	4
Installation	4
Start Test	6
Example Test.....	7
Command Description	8
Command Summary.....	9
Command	10
Connection Command	10
General Command.....	11
ISP Command	11
IAP Command.....	12
Engineering Command	13
Return Value.....	13
Examples of Use	14

Introduction

The HT32 Flash Commander is a command prompt tool for firmware updates, allowing users to read/write/erase the HT32 MCU internal Flash through command inputs. The HT32 Flash Commander supports two update modes: ISP (In-System Programming) mode and IAP (In-Application Programming) mode. Each update mode supports two communication interfaces, USB and COM (UART) ports.

The full range of the Holtek HT32 MCU internal Flash memory stores an un-erasable Image, called the Bootloader*^{Note}, which has the ability to control the internal Flash. When the user issues a request through the specified USB/UART port, the Bootloader will update the internal Flash according to the corresponding request. Updating the firmware using this Bootloader is what is known as the ISP mode.

In most applications, the main program has the ability to update the firmware. These application programs general have two Images stored in two different blocks. The first block has the ability to implement firmware updates, called the IAP block. The second block is used to store the application program, called the AP block. Using the IAP to update the AP is what is known as the IAP mode.

Note: The Bootloader storage address starts from 0x1F000000. The Bootloader has been programmed into the MCU during manufacture and cannot be written/erased.

Functional Description

- Supports four firmware update methods:
 - ♦ ISP USB Mode
 - ♦ ISP UART Mode
 - ♦ IAP USB Mode
 - ♦ IAP UART Mode
- Read/Write Flash
- Memory protection – ISP Mode
- Image file is compatible with Hex and Binary formats
- Parameters are compatible with Hexadecimal and Decimal formats
- Supports multi-command functions

Usage Description

Installation

- Step 1. Obtain the latest version of the HT32 Flash Commander from Holtek's official website, the installation file name is "HT32_Flash_Commander_Vnnn.exe", where "nnn" represents the version number.
- Step 2. Double-click on "HT32_Flash_Commander_Vnnn.exe". The default installation path is "C:\Programs(x86)\Holtek HT32 Series\HT32 Flash Commander", to select a different folder, click "Browse" to change the installation path, and then click "Next" to continue the installation.

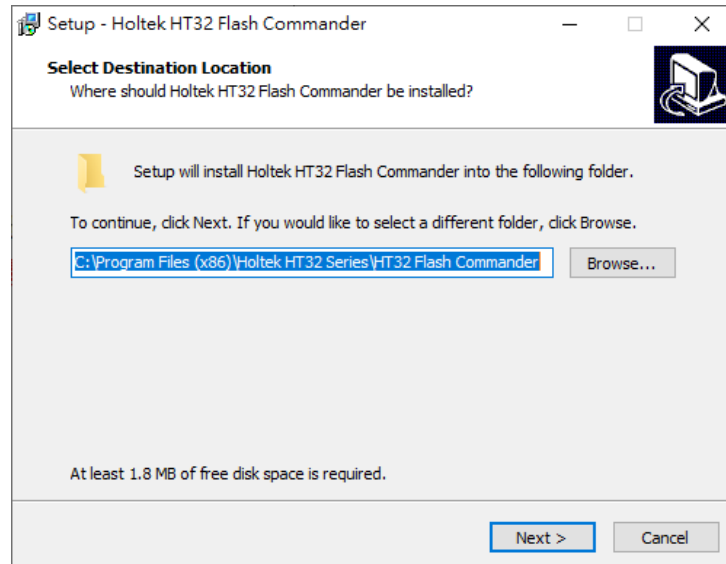


Figure 1. HT32 Flash Commander Installation – Select the Installation Path

Step 3. Choose whether or not to add environmental variables. After completion, click the “Next” button to continue the installation. It is important to note that if environmental variables are added, this function will not take effect until the computer is manually rebooted after installation.

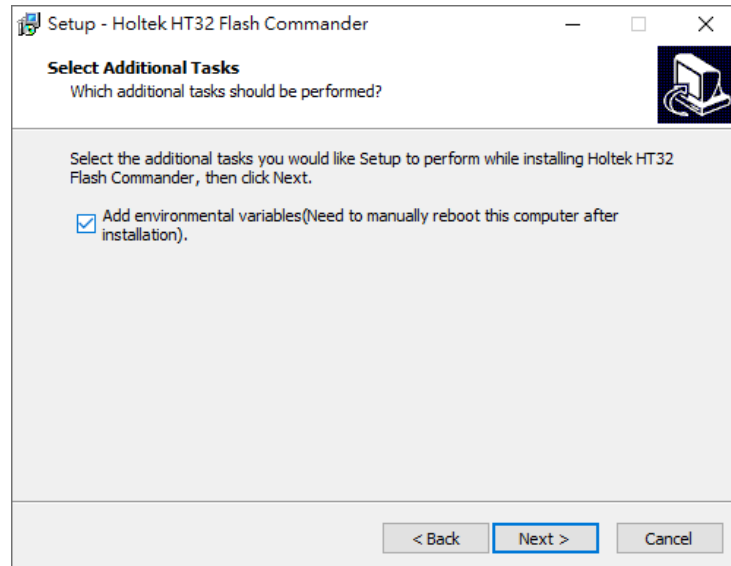


Figure 2. HT32 Flash Commander Installation-Add Environmental Variables

Step 4. When the “Ready to Install” screen appears, click the “Install” button to start the installation.

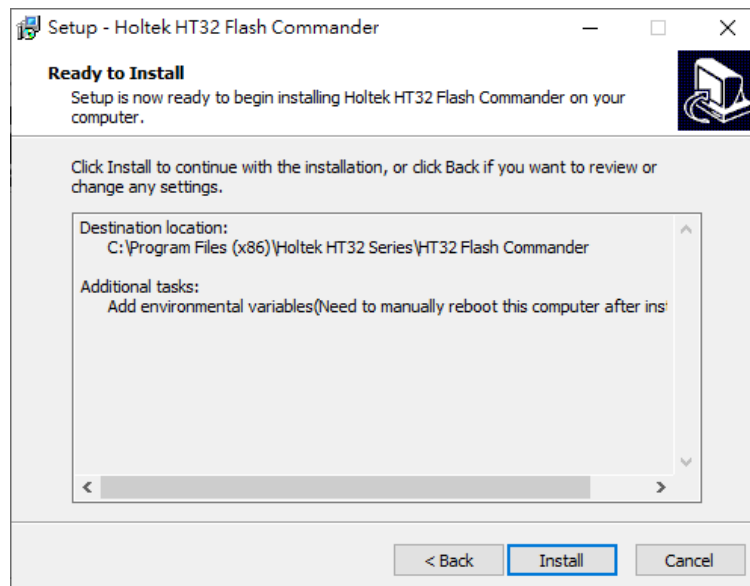


Figure 3. HT32 Flash Commander Installation – Ready to Install Page

Step 5. Once the installation has finished, a completion page will appear as shown below. Select whether to view the user manual and whether to launch HT32 Flash Commander. Finally click the “Finish” button to complete the installation process.

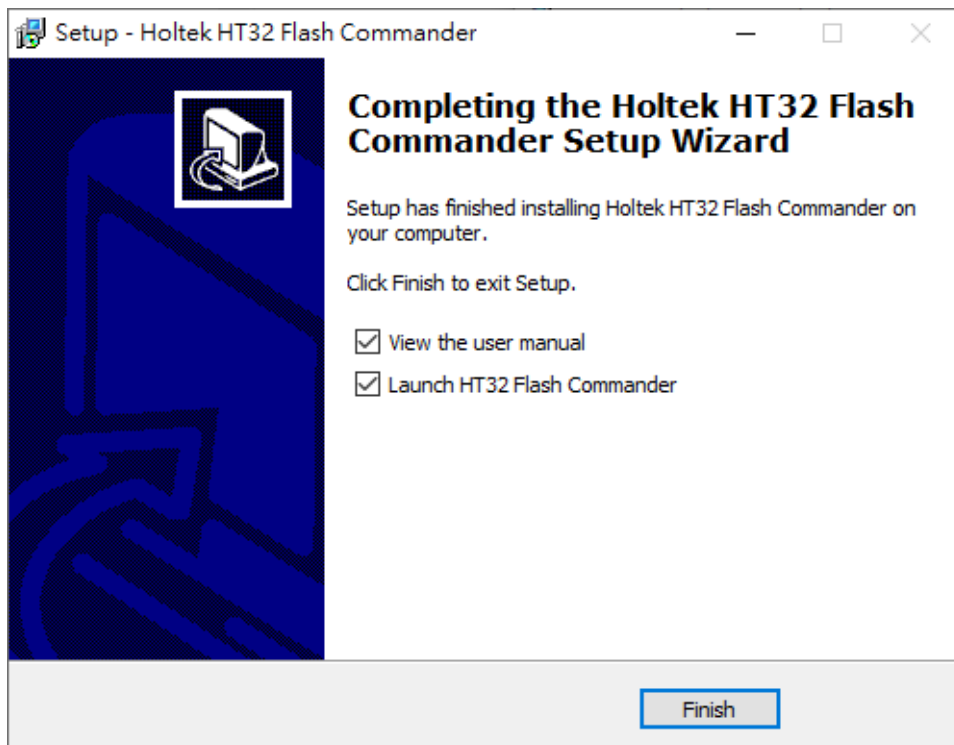


Figure 4. HT32 Flash Commander Installation-Completion

Start Test

Here the ISP mode is taken as an example, follow the steps below to complete the firmware update.

Step 1. Prepare a HT32 development board (Starter kit) and select the communication interface according to the target MCU specifications.

- (1) UART interface: connect the USB cable to the USB connector of the e-Link32 Lite on the development board.
- (2) USB interface: connect the USB cable to the USB connector under the board.

Step 2. Ensure that the BOOT pin is setup as follows.

M0+: BOOT=0.

M3: BOOT0=1, BOOT1=0.

When the configuration has finished, press the “Reset” button. After the MCU has reset, the ISP update can be executed.

Step 3. Place the pre-updated image on the HT32 Flash Commander installation path. The Image file supports Hex or Binary format.

Step 4. Turn on the command prompt character and jump to the installation path of the HT32 Flash Commander using the DOS command, the default path is “C:\Program Files (x86)\Holtek HT32 Series\HT32 Flash Commander” .

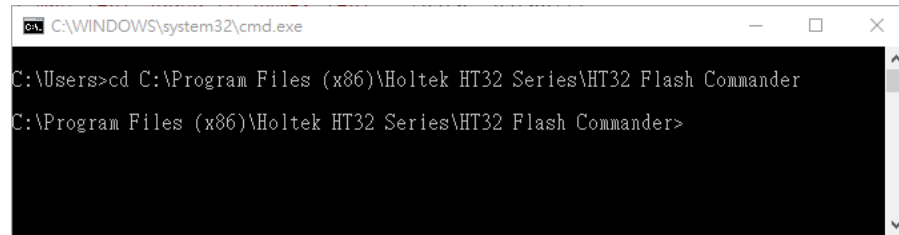


Figure 5. HT32 Flash Commander Installation Path

Step 5. Enter the command to start the firmware update.

- (1) UART interface: COM33, the Image name is HT32.hex, enter the command “ht32flashcmd -isp -cc 33 -a HT32.hex fpar=OPV”, the finished screen is shown below.

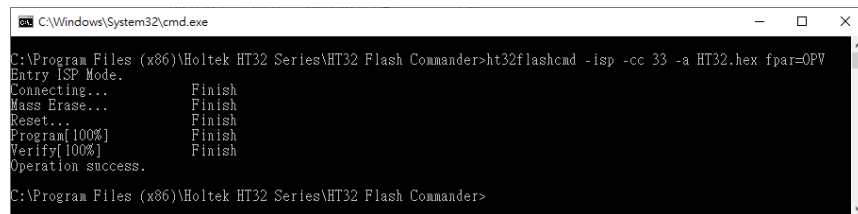


Figure 6. UART Interface Update Finished Screen

- (2) USB Interface: The Image name is “HT32.hex”, enter the command “ht32flashcmd -isp -cu -a HT32.hex fpar=OPV”, the finished screen is shown below.

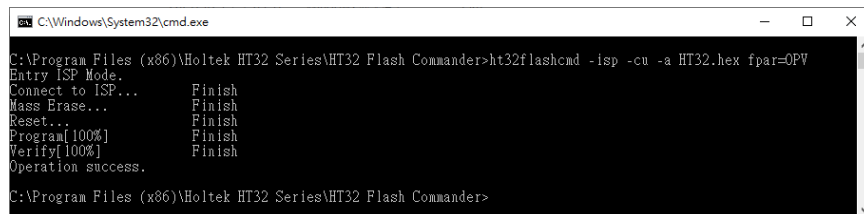


Figure 7. USB Interface Update Finished Screen

Example Test

There are four examples in the installation path: IAP_UART_Example, IAP_USB_Example, ISP_UART_Example and ISP_USB_Example. The following description explains how to update the firmware using the ISP example.

Step 1. Manually force the MCU to enter the Bootloader. Refer to Step1 and Step2 of the “Start Test”.

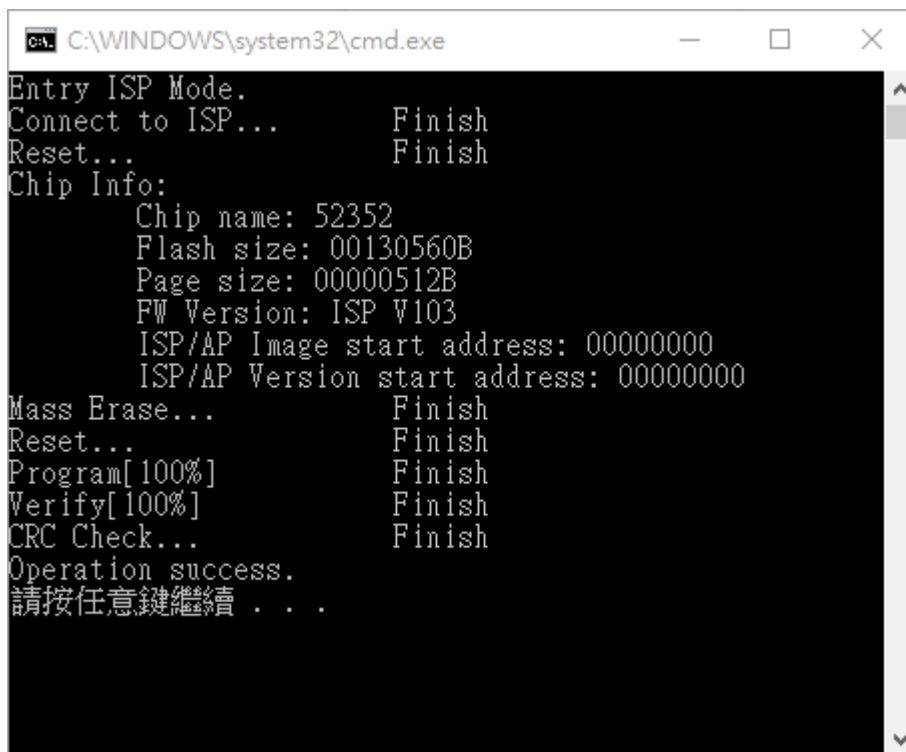
Step 2. Open the corresponding example folder according to the communication interface. The path is as follows.

- (1) UART example path: “C:\Program Files (x86)\Holtek HT32 Series\HT32 Flash Commander\ISP_UART_Example”.
- (2) USB example path: “C:\Program Files (x86)\Holtek HT32 Series\HT32 Flash Commander\ISP_USB_Example”.

Step 3. Place the pre-updated image into the example folder. The Image file supports Hex or Binary formats.

Step 4. Modify the “config.bat” file using the text editor. The modifications include Image file name and COM port number (UART interface).

Step 5. Execute “Start.bat”. This bat file can update the target Image file to the MCU via ht32flashcmd. The following figure uses the ESK-30501 (HT32F52352) as an example.



```

C:\WINDOWS\system32\cmd.exe
Entry ISP Mode.
Connect to ISP...      Finish
Reset...              Finish
Chip Info:
  Chip name: 52352
  Flash size: 00130560B
  Page size: 00000512B
  FW Version: ISP V103
  ISP/AP Image start address: 00000000
  ISP/AP Version start address: 00000000
Mass Erase...         Finish
Reset...              Finish
Program[100%]         Finish
Verify[100%]          Finish
CRC Check...          Finish
Operation success.
請按任意鍵繼續 . . .
  
```

Figure 8. ISP Example Update Completion Page

Command Description

The ht32flashcmd supports multiple commands. Each execution of the program can contain multiple commands and parameters. The command syntax is as follows:

“ht32flashcmd [Cmd0 <Para0> <Para1> ...] [Cmd1 <Para0> <Para1> ...]...”

Note: To clearly identify the space bar, replace it with “ ”.

Parameter	Description
ht32flashcmd	The execution file name
Cmdn (n=0, 1, 2, ...)	The ht32flashcmd completes the corresponding command according to the “Cmd” entered by the user.
Paran (n=0, 1, 2, ...)	Para is a parameter, its usage method and format vary depending on the “Cmd”, therefore it is necessary to fill in the correct Para according to different “Cmd”.

Command Summary

Connection Command		
Command and Parameter	Description	Example
-h help ? /?	Display the usage method.	C:\ht32flashcmd -h C:\ht32flashcmd help C:\ht32flashcmd ? C:\ht32flashcmd /?
-isp	Select ISP mode.	ISP USB mode:
-iap	Select IAP mode.	C:\ht32flashcmd -isp -cu ISP UART mode:
-rsti	After the MCU has successfully connected, reset the MCU and jump to the IAP/Bootloader to execute.	C:\ht32flashcmd -isp -cc 8 IAP USB mode:
-cu [<APVID>□<APPID>□ <APINF>□<IAPVID>□ <IAPPID>□<IAPINF>□<DC>]	Connect to the MCU via the USB.	C:\ht32flashcmd -iap -rsti -cu 0x1234 2 0 0x1234 1 0 0 -ck Command_key.bin IAP UART mode:
-cc□<CPN>□[]	Connect to the MCU via the COM port (UART)	C:\ht32flashcmd -iap -rsti -cc 8 -ck Command_key.bin
-ck□<Filename>	Load the command key (only IAP mode support).	
General Command		
-a□ <Filename>□ [fpar=N M OBPVC]	Execute a firmware update.	C:\ht32flashcmd ... -a HT.hex C:\ht32flashcmd ... -a AP.bin fpar=OBPC C:\ht32flashcmd ... -a HT.hex fpar=MPV
-i	Obtain the MCU information.	C:\ht32flashcmd ... -i
-r32□ <Address>□<NumWords>□ <Filename>	Read the internal Flash data.	C:\ht32flashcmd ... -r32 0 256
-dump□ <Address>□<NumWords>□ <Filename>	Read the internal Flash data and write it into the file.	C:\ht32flashcmd ... -dump 0 256 Data.bin
-run	After all commands have completed, reset the MCU and jump to the AP block to execute.	C:\ht32flashcmd ... -run
-list	List all COM ports.	C:\ht32flashcmd -list
-q	Quiet mode.	C:\ht32flashcmd -q
-eng	Enter the engineering mode.	C:\ ht32flashcmd -eng
-tsn□<TargetChipSN>	Check the serial number.	C:\ht32flashcmd ... -tsn 52352 -a HT.hex
-tout□<CommandTimeout>□ <PageEraseTimeout>	Set command and page erase timeout in milliseconds. It should be multiples of 100.	C:\ht32flashcmd ... -tout 200 500
ISP Command		
-ob□ [<cp0=CP0>]□[<cp1=CP1>]□ [<pp0=PP0>]□[<pp1=PP1>]□ [<pp2=PP2>]□[<pp3=PP3>]□ [<wdt=WDT>]	Set memory protection.	C:\ht32flashcmd ... -ob cp0=1 C:\ht32flashcmd ... -ob pp0=0xFFFF cp1=1 C:\ht32flashcmd ... -ob cp0=1 pp0=123 pp1=0x5a5a5a
-rob	Display memory protection settings.	C:\ht32flashcmd ... -rob
IAP Command		
-ae□ <Filename>□<Start Addr>	Update the data to the external Flash.	C:\ht32flashcmd ... -ae AP.bin 0 C:\ht32flashcmd ... -ae AP.bin 0x2800
-dg□<Filename>	Set “Digest” according to the “bin” file.	C:\ht32flashcmd ... -dg AP_Digest.bin
-dgt□<DGT>	Set “Digest” according to the parameters <DGT>.	C:\ht32flashcmd ... -dgt 0x5A21

-sv□<VersionString>	Set the AP version.	C:\ht32flashcmd ... -sv v110
-v	Read the AP version.	C:\ht32flashcmd ... -v
Engineering Command		
-exit	Exit the engineering mode command	FlashCommander>-exit
-w□ <Filename>□<Start Addr>	Write data into the Flash.	FlashCommander>-w AP.bin 0 FlashCommander>-w AP.bin 0x2800
-se□ <Start Addr>□<EndAddr>	Erase the specified Flash block	FlashCommander>-se 0 1023 FlashCommander>-se 0x2800 0x2BFF FlashCommander>-se 0x2C00 0x77FF
-me	Erase all the internal Flash	FlashCommander>-me
-bk□ <Start Addr>□<EndAddr>	Bank check the specific internal Flash block	FlashCommander>-bk 0 1023 FlashCommander>-bk 0x2800 0x2BFF
-crc□ <Start Addr>□ <EndAddr>□[<DGT>]	Obtain the internal Flash CRC value and transmit the Digest Key to the MCU	FlashCommander>-crc 0 0x3FF FlashCommander>-crc 0 0x577 FlashCommander>-crc 0 0x1053 0x5261
-rst□<RESET_MODE>	Reset the MCU and jump to AP or to the IAP/Bootloader	FlashCommander>-rst 0 FlashCommander>-rst 1

Command

Connection Command

- h | help | ? | /?: Display the usage of each command and the description of each response value, allowing users to quickly familiarise themselves with the rules for each command. This command must be used separately and cannot be used together with other commands, such as C:\ht32flashcmd help.
- isp: Use the ISP communication protocol to update the firmware, and in this mode ht32flashcmd will support ISP commands.
- iap: Use the IAP communication protocol to update the firmware, and in this mode ht32flashcmd will support IAP commands.
- rsti: After the MCU has successfully connected, the MCU will reset and jump to the IAP/Bootloader for execution. This command is required in most IAP applications, because the firmware update can only be performed when the MCU jumps to the IAP.
- cu□[<APVID>□<APPID>□<APINF>□<IAPVID>□<IAPPID>□<IAPINF>□<DC>]: For the USB connection. All parameters below are for the IAP mode only.
 - APVID: the VID of the AP, the range of the APVID is 0x0 ~ 0xFFFF.
 - APPID: the PID of the AP, the range of the APPID is 0x0 ~ 0xFFFF.
 - APINF: the INF of the AP, the range of the APINF is 0x0 ~ 0xFF.
 - IAPVID: the VID of the IAP, the range of the IAPVID is 0x0 ~ 0xFFFF.
 - IAPPID: the PID of the IAP, the range of the IAPPID is 0x0 ~ 0xFFFF.
 - IAPINF: the INF of the IAP, the range of the IAPINF is 0x0 ~ 0xFF.
 - DC: Data confusion control
 - 0: Disable
 - 1: Enable
- cc□<CPN>□[
]: Use COM port for connection – UART mode.
 - CPN: COM port number, the range of which is 1~255.
 - BR: Set baud rate. This parameter can be omitted. If this parameter setting is omitted, the baud rate has the default value of 115200bps.

7. -ck□<Filename>: Set the Command Key according to the binary <Filename>. This command only supports the IAP mode.

General Command

1. -a□<Filename>□[fpar=N|M|OBPVC]: Update the firmware. The supported files include Binary or Hex formats.

Filename: The Image file to be updated.

fpar: Update behavior options. This parameter can be omitted and only supports the ISP mode. If this parameter is omitted, ht32flashcmd uses the default value of “fpar=NPV”.

N|M|O: Erase the internal Flash.

N: Only erase the required pages

M: Erase the entire MCU.

O: Erase the entire MCU, after the erasing has finished, reset the MCU*^{Note}.

B: Blank check

P: Programming the program.

V: Verify the program.

C: CRC verification

Note: After the specific HT32 MCU option byte has been erased, memory protection can be removed when the MCU is reset.

2. -i: Obtain the MCU relevant information and display the results.
3. -r32□<Address>□<NumWords>: Read the internal Flash in 32-bit units from the beginning of the <Address> and display the results.
4. -dump□<Address>□<NumWords>□<Filename>: Read the internal Flash in 32-bit units from the beginning of the <Address> and store the results into <Filename>. The <Filename> format is binary.
5. -run: After all commands have been executed, an MCU reset will occur and the MCU will jump to the AP to execute. After the ISP mode is reset, the MCU will jump to the internal Main Flash to execute. If the ISP mode is required to update again, it must manually enter Bootloader.
6. -list: List all COM ports and display the results.
7. -q: Quiet mode, the ht32flashcmd will not display any information when executing each command.
8. -eng: Enter the engineering mode, where the engineering commands can be executed. The ht32flashcmd does not end after the command has executed, so users can send the commands in a batch.
9. -tsn□<TargetChipSN>: Check the serial number of the target chip before executing “-a” update the firmware.
10. -tout□<CommandTimeout>□<PageEraseTimeout>: Set <CommandTimeout> and <PageEraseTimeout> in milliseconds, with adjustment values required to be multiples of 100.

ISP Command

The ISP commands only support the ISP mode. If the following commands are not executed in the ISP mode, the ht32flashcmd will not be able to judge and report the error.

1. -ob□[<cp0=CP0>]□[<cp1=CP1>]□[<pp0=PP0>]□[<pp1=PP1>]□[<pp2=PP2>]□[<pp3=PP3>]□[<wdt=WDT>]: Set memory protection. This command will not preserve the original settings, and

during executing the “-ob” command, the ht32flashcmd will erase the original settings and then reset, so the unset parameters are cleared to 0 (disable).

cp0=CP0, CP0 is 0 or 1.

0: Disable memory protection

1: Enable memory protection

cp1=CP1, CP1 is 0 or 1.

0: Disable Option Byte write/erase protection

1: Enable Option Byte write/erase protection

pp0=PP0, <value> is from 0 to 0xFFFFFFFF.

pp1=PP1, <value> is from 0 to 0xFFFFFFFF.

pp2=PP2, <value> is from 0 to 0xFFFFFFFF.

pp3=PP3, <value> is from 0 to 0xFFFFFFFF.

There are 128 bits from pp0 to pp3, which can be expressed as pp[127:0] bits.

pp[n]=0, Sector n, disable write/erase protection

pp[n]=1, Sector n, enable write/erase protection

wdt=WDT, WDT is 0 or 1. The AUTO-WDT function is controlled by the WDT value. Part of the HT32 MCUs do not support this function. If the AUTO-WDT function is enabled, after the MCU is reset, the WDT will be enabled automatically by the hardware.

0: Disable AUTO-WDT function

1: Enable AUTO-WDT function

2. -rob: Display memory protection setting.

IAP Command

The IAP command only supports the IAP mode. If the following commands are not executed in the IAP mode, the ht32flashcmd will not be able to judge and report the error.

1. -ae□<Filename>□<Start Addr>: Update the external Flash. Use the <Filename> start erasing/writing/verification the external Flash from <Start Addr>. <Filename> only supports a Binary format.
2. -dg□<Filename>: Set the Digest Key^{*Note(1)} via the <Filename>. The <Filename> format is Binary.
3. -dgt□<DGT>: Set the Digest Key^{*Note(1)} via the <DGT>. The <DGT> range is 0~0xFFFF.
4. -sv□<VersionString>: Set the AP firmware version^{*Note(2)}. Space characters are not allowed in <VersionString>, and the length supported by the version string is adjusted according to the F/W definition.
5. -v: Read the AP firmware version string^{*Note(2)}.

Notes:

- (1) The Digest Key must be correct to complete the “-a” for program update (when Digest Key is error, ht32flashcmd will not report an error, but the Image file will not be programmed into the AP).
- (2) “-sv” must be executed after the “-a”, and “-sv” is executed successfully means that the AP update has completed. The confirmation method is to read the AP version via “-v”, if empty then this represents an update failure.

Engineering Command

The “-eng” engineering command can only be used in the engineering mode. If the ht32flashcmd does not enter engineering mode, it will not be able to judge the engineering command thus reporting errors.

1. -exit: Exit the engineering mode.
2. -w<Filename><Start Addr>: Write <Filename> from <Start Addr> into Flash. <Filename> only supports a Binary format.
3. -se<Start Addr><End Addr>: Erase the Flash from <Start Addr> to <End Addr>.
4. -me: Erase the entire MCU.
5. -bk<Start Addr><End Addr>: Blank check internal Flash from <Start Addr> to <End Addr>.
6. -crc<Start Addr><End Addr>[<DGT>]: Calculate the CRC code from <Start Addr> to <End Addr> and return the results. The <DGT> can be omitted and valid only in the IAP.
7. -rst<RESET_MODE>: Reset MCU and jump to the Bootloader/IAP or AP to execute.

RESET_MODE: Reset mode

0: After reset, the MCU jumps to the AP block to execute.

1: After reset, the MCU jumps to the Bootloader/IAP block to execute.

Return Value

Use the environment variable “%ERRORLEVEL%” to obtain the Return Code. 0 represents a success, other codes stand for an error condition.

Return Code	Support Command	Description
0	all	Operation successful.
1	all	Invalid command parameter. Check the parameter rules for each command.
2	-cc, -cu	An error occurred when connecting to the target MCU. Possible reasons are: 1. Target MCU is not powered on. 2. MCU is not connected to the computer.
3	all	The command does not apply to the Commander. Confirm that the command is correct and confirm that the IAP or ISP update mode supports this command.
4	all	Un-connection. Execute “-cu” or “-cc”.
5	all	The update mode is not set. The ht32flashcmd is unable to determine if the update mode is IAP or ISP, execute “-iap” or “-isp”.
6	all	Response timeout. There is no response from the device after the command is issued.
11	-i, -a, -ob, -rob, -sv, -v	Failed to obtain the MCU information.
12	-a, -ae, -sv, -v, -w	Failed to update the program. Possible reasons are: 1. Whether the target Image file is larger than the Flash configuration. 2. This memory has been Page Erase/Program protected, please check the OB_PP bit of the Option Byte . The page protection of Option Byte can only be disabled by a mass erase operation. Example: “-a xxx.bin fpar=OPV”.
13	-a, -ae, -sv, -v, -w	Verification failed. Possible reasons are: 1. Whether the target Image file is larger than the Flash configuration. 2. The data in Flash memory does not match with the file data.

Return Code	Support Command	Description
14	-r32, -dump, -rob, -ob	Failed to read the internal Flash.
15	-cc, -cu, -a, -run, -rst	Failed to reset to the AP/IAP/Bootloader.
16	-a, -ae, -sv, -ob, -se, -me	Failed to erase the internal Flash. Possible reasons are: 1. Whether the target Image file is larger than the Flash configuration. 2. This memory has been Page Erase/Program protected, please check the OB_PP bit of the Option Byte. The page protection of Option Byte can only be disabled by a mass erase operation. Example: "-a xxx.bin fpar=OPV".
17	-a, -bk	Blank check failed. Confirm that the blank check range has been erased.
18	-a	CRC check failed. Possible reasons are: 1. The target Image file is larger than the Flash configuration. 2. The <DGT> digest value does not match with the device's digest.
19	-crc	Unable to obtain the CRC check code.
22	-a + -tsn	This target chip doesn't match with the serial number.
31	-a, -ob	Option Byte setting error. Erase the entire MCU, e.g. "-a xxx.bin fpar=OPV".
32	-ob	The internal memory of the MCU has been securely locked, e.g. "-a xxx.bin fpar=OPV".
33	-ob	The MCU Option Byte is protected by the memory. Erase the entire MCU, e.g. "-a xxx.bin fpar=OPV".
34	-ob	This MCU does not support AUTO-WDT function.
35	-a, -ob	Option Byte is repeatedly set. Both Hex file and "-ob" are set with Option Byte. Select a method to set.
40	-a, -ae, -dump, -ck, -dg, -w	File access failed. Confirm the file path and that the file exists.

Examples of Use

Case1: ISP USB mode, for firmware update:

```
ht32flashcmd -isp -cu -a HT32.bin
```

Case2: ISP USB mode, for firmware update + Security Lock:

```
ht32flashcmd -isp -cc 8 -a HT32.hex -ob cp0=1
```

Case3: ISP UART mode (COM1, the default baud rate is of 115200bps), for firmware update:

```
ht32flashcmd -isp -cc 1 -a HT32.hex
```

Case4: IAP USB mode:

```
ht32flashcmd -iap -rsti -cu 0x1234 0x0002 0x00 0x1234 0x0001 0x00 0 -ck Command_Key.  
bin -dgt 0x5643 -a AP.bin -sv V001 -run
```

Case5: IAP USB mode, obtain the AP version:

```
ht32flashcmd -iap -rsti -cu 0x1234 0x0002 0x00 0x1234 0x0001 0x00 0 -ck Command_Key.  
bin -v -run
```

Case6: IAP UART Mode (COM8, Baud rate 204800):

```
ht32flashcmd -iap -rsti -cc 8 204800 -ck Command_Key.bin -dgt 0x5643 -a AP.bin -sv V002 -run
```

Case7: ISP UART mode (COM20, the default baud rate is of 115200bps), In engineering mode, write the data at 0x0 and exit after execution:

```
ht32flashcmd -isp -cc 20 -eng -w HT32.bin 0x0 -exit
```

Copyright© 2023 by HOLTEK SEMICONDUCTOR INC. All Rights Reserved.

The information provided in this document has been produced with reasonable care and attention before publication, however, HOLTEK does not guarantee that the information is completely accurate. The information contained in this publication is provided for reference only and may be superseded by updates. HOLTEK disclaims any expressed, implied or statutory warranties, including but not limited to suitability for commercialization, satisfactory quality, specifications, characteristics, functions, fitness for a particular purpose, and non-infringement of any third-party's rights. HOLTEK disclaims all liability arising from the information and its application. In addition, HOLTEK does not recommend the use of HOLTEK's products where there is a risk of personal hazard due to malfunction or other reasons. HOLTEK hereby declares that it does not authorise the use of these products in life-saving, life-sustaining or safety critical components. Any use of HOLTEK's products in life-saving/sustaining or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold HOLTEK harmless from any damages, claims, suits, or expenses resulting from such use. The information provided in this document, including but not limited to the content, data, examples, materials, graphs, and trademarks, is the intellectual property of HOLTEK (and its licensors, where applicable) and is protected by copyright law and other intellectual property laws. No license, express or implied, to any intellectual property right, is granted by HOLTEK herein. HOLTEK reserves the right to revise the information described in the document at any time without prior notice. For the latest information, please contact us.