



HT32F49xxx Flash Programmer User Guide

Revision: V1.00 Date: April 10, 2023

www.holtek.com

Table of Contents

1. Introduction	3
1.1 Environmental Requirements	3
1.2 Glossary	3
2. Bootloader Introduction	4
2.1 Enter Bootloader Mode	4
2.2 Hardware Connection Requirements	5
2.3 Peripheral Configuration	5
2.4 Programming Mode Selection	6
3. Installation	7
3.1 Install Programmer	7
3.2 Install USB DFU Driver	8
4. User Interface	13
4.1 Device Connection Page	13
4.2 Flash Status Page	16
4.3 Device Information Page	17
4.4 Operation Configuration Page	19
4.5 Operation Progress Page	27
4.6 SPIM Encryption Download	27

1. Introduction

This user guide describes the HT32F49xxx Flash Programmer. The HT32F49xxx Flash Programmer is a graphical interface application developed to demonstrate the In-System Programming (ISP) function of the HT32F49xxx series of MCU. With this application, users can access the Bootloader through the UART port or USB port to implement program update for the HT32F49xxx series of MCU.

1.1 Environmental Requirements

- Software requirements
Windows 7 and above are required.
For software version below 2.0.04, .Net framework 4.0 is required.
For software version 2.0.04 and above, .Net framework 4.6 is required.
- Hardware requirements
Available serial communication ports (COM).
Available USB communication ports.

1.2 Glossary

- ISP
In-System Programming. Users can directly perform write or erase operations on the device with ISP function.
- UART
Universal Asynchronous Receiver/Transmitter. It is a serial communication port (COM) for full-duplex asynchronous communication.
- USB
Universal Serial Bus. It is an external bus standard used to regulate the connection and communication between computers and external devices.
- DFU
Device Firmware Upgrade. It is a device firmware update protocol based on USB communication.

2. Bootloader Introduction

2.1 Enter Bootloader Mode

The following table describes how to set the HT32F49xxx series to enter the Bootloader mode. Taking the HT32F49395 as an example, during the boot-up process, the hardware determines whether the conditions BOOT0=1, BOOT1=0 and BTOPT=1 are met, if yes, the device will enter the Bootloader mode and receives ISP commands through the communication interface. Note that the method for entering Bootloader mode may vary among different MCU types due to differences in the Boot pin number and the function of the user system data area. Refer to the specific device datasheet for more information.

Part No.	Conditions
HT32F49xxx: Boot Pin×2 + supporting Bank 2	BOOT0=1, BOOT1=0, BTOPT=1
HT32F49xxx: Boot Pin×2	BOOT0=1, BOOT1=0
HT32F49xxx: Boot Pin×1	BOOT0=1, nBOOT1=1

Bootloader Mode

- Note: 1. Refer to the datasheet of the specific device for the corresponding pin locations of BOOT0 and BOOT1.
2. The BTOPT bit is located in the user system data area. For the MCU with Bank 2, it means to boot from Bank 2 when BTOPT is 0.
3. The nBOOT1 bit is mapped in the user system data area and it can be modified by software. Refer to the user manual of the corresponding device for details.

In addition to the methods mentioned in the above table, the Bootloader can be executed by jumping to the Bootloader code area using the user code. Before jumping, all peripherals must be reset and all peripheral clocks, PLL, interrupts must be disabled and pending interrupts signals must be cleared.

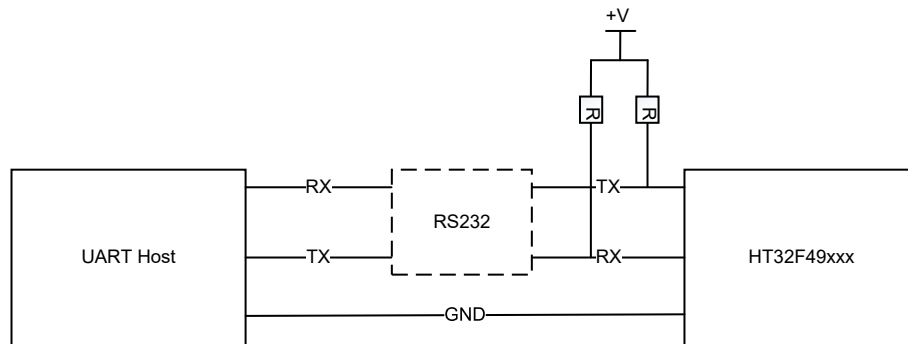
When the Bootloader has entered a programming mode, other programming mode detection will be disabled. For example, the Bootloader supports USART1, USART2 and USB_DFU, when the Bootloader has detected 0x7F on USART1, it will enter the USART1 programming mode and disable the detection of USART2 and USB_DFU.

Note that when using UART for programming, if there is data transmission on the UART RX pin during the baud rate detection, it will enter the corresponding UART programming mode. If the data is not 0x7F, the UART will be configured with an incorrect baud rate and subsequent communication will not proceed properly. Therefore, it is recommended to keep the unused peripheral RX interfaces such as UART_RX at a fixed high or low level when the Bootloader starts. If these pins remain floating or there is data transmission, it may result in access to an unused interface.

Note: It is recommended to wait for 200ms after power-on before sending the Bootloader command.

2.2 Hardware Connection Requirements

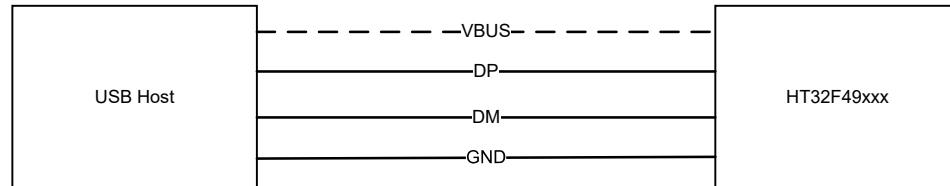
To use the UART Bootloader programming mode, the host must be connected to the corresponding UART RX and TX pins.



Note: The typical voltage for +V is 3.3V and the typical value for R is 100kΩ.

UART Hardware Connection

To use USB DFU, the MCU USB interface should be connected to the USB host.



Note: When the Bootloader starts, it is recommended to keep the unused peripheral RX interfaces such as UART_RX at a fixed high or low level when the Bootloader starts. If these pins remain floating or there is data transmission, it may result in access to an unused interface.

USB Hardware Connection

2.3 Peripheral Configuration

The following tables take HT32F49395 as an example to describe the communication interfaces supported by ISP Bootloader and the pin definition of each interface. Refer to each device datasheet for details.

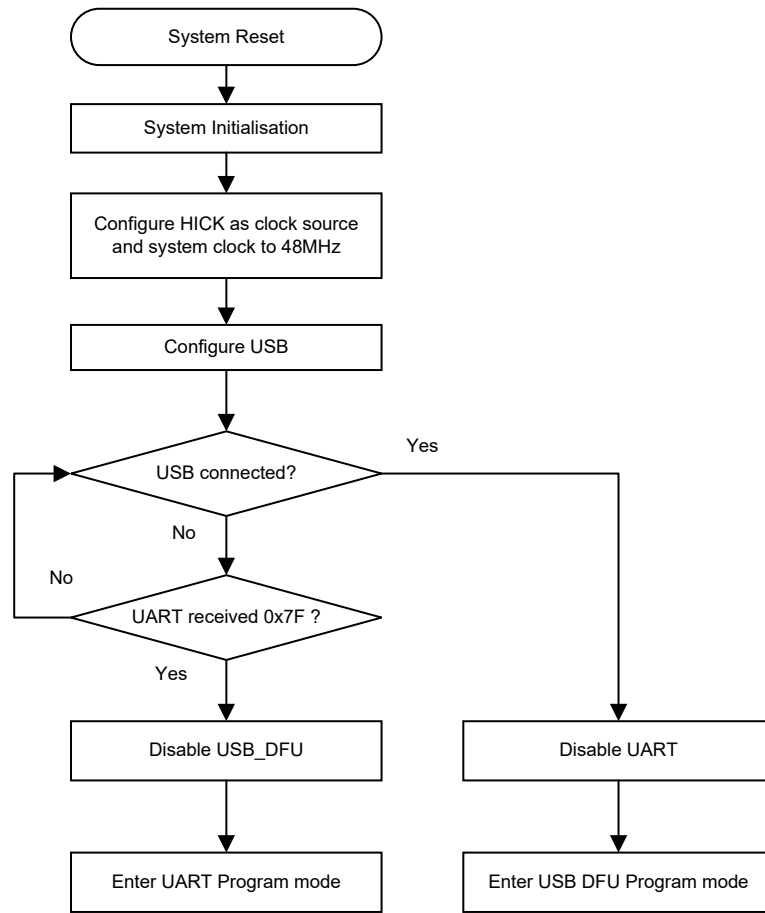
Part No.	Supported Interfaces		
	USART1	USART2	DFU
HT32F49395	Y	Y	Y

HT32F49395 Supported Interfaces

IP	Applicable Devices	TX Pin/DM	RX Pin/DP
USART1	All	PA9	PA10
USART2	HT32F49395 100-pin LQFP	PD5	PD6
	Other	PA2	PA3
USB DFU	All	PA11	PA12

HT32F49395 UART/USB Pin Definition in Bootloader Mode

2.4 Programming Mode Selection



ISP Programming Mode Selection

3. Installation

The hardware environment is set up as follows:

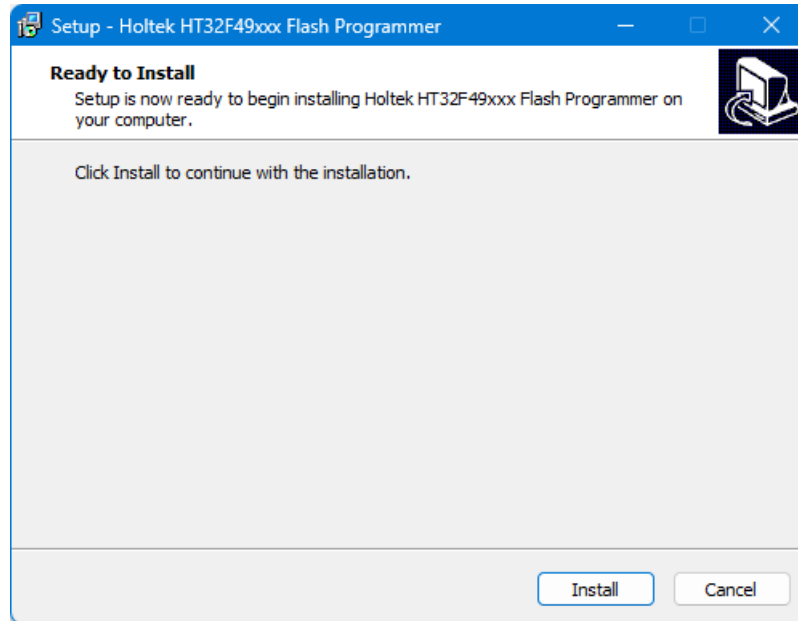
- UART communication: The device must be connected to the available serial communication port (COM) on the computer.
- DFU communication: The device must be connected to the available USB port on the computer.

Note: When a USB has been connected, the UART communication will not be supported. To change to UART mode, it is required to remove the USB and reset the system.

3.1 Install Programmer

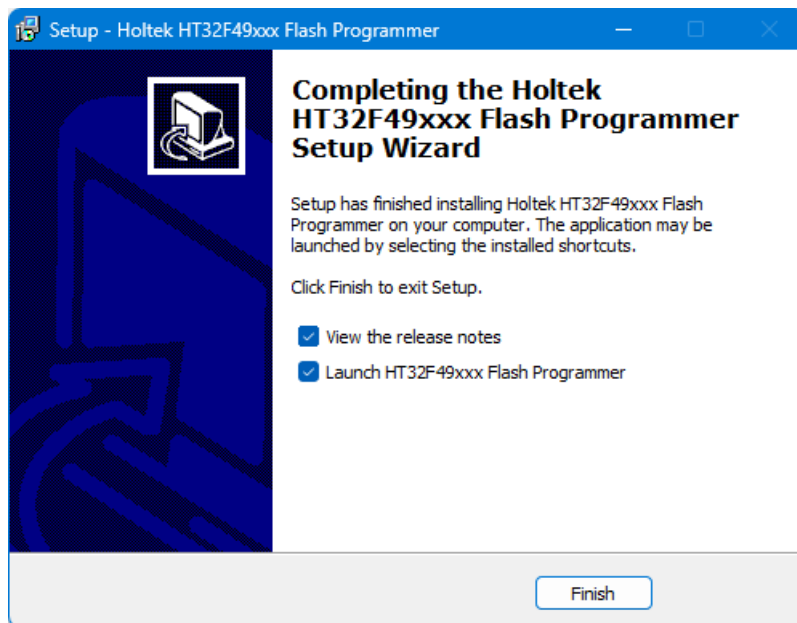
Obtain the latest version of the HT32F49xxx Flash Programmer from the Holtek official website. The installation file is named “HT32F49xxx_Flash_Programmer_Vn.m.r.exe”. The installation steps are as follows:

- Double-click on “HT32F49xxx_Flash_Programmer_Vn.m.r.exe” and click “Install” on the dialog below.



Install Programmer

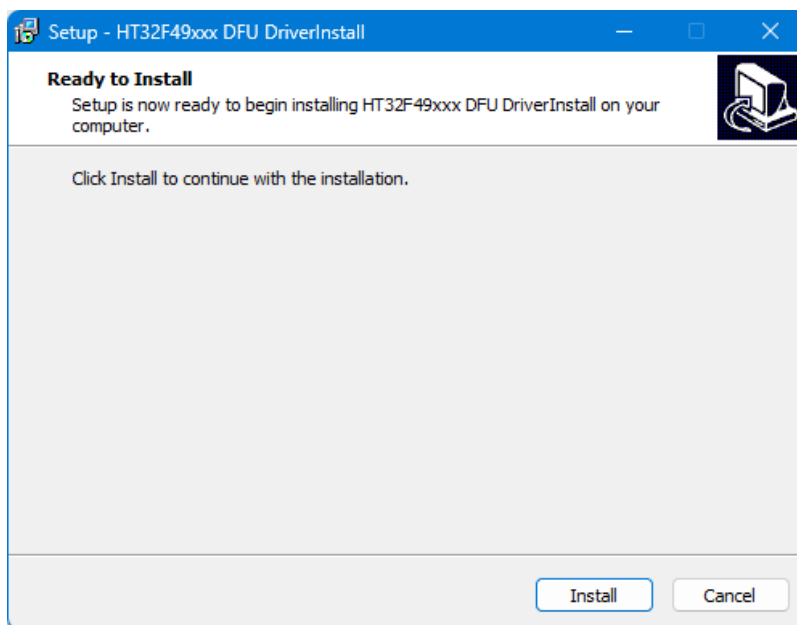
- When the installation has finished, a completion page will appear as shown below. Choose whether or not to view the release notes or to launch the HT32F49xxx Flash Programmer. Click “Finish” to complete the installation.



Finish Programmer Installation

3.2 Install USB DFU Driver

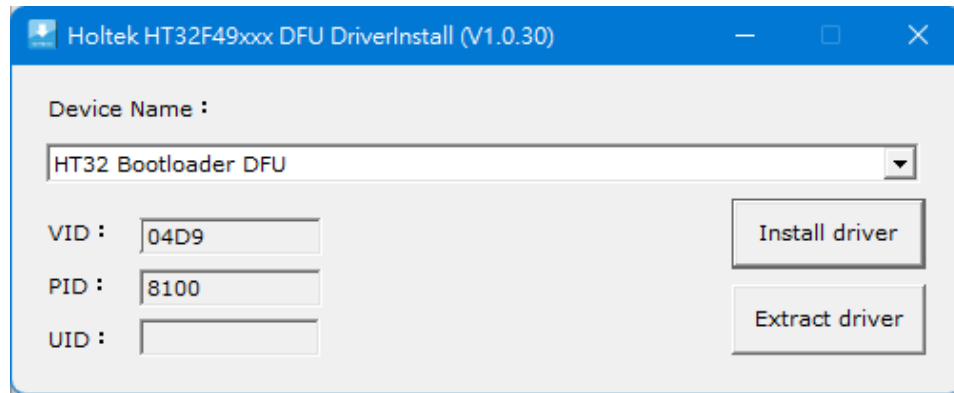
To use the USB DFU communication, it is required to install the USB DFU driver for the used HT32 MCU. The installation file is named “HT32F49xxx_DFU_DriverInstall_Vn.m.r.exe”, double-click on this file and click “Install” on the dialog below.



Install DFU Driver

3.2.1 Automatic Installation

Then enter the driver installation interface. The driver installation program will automatically scan all the “HT32 Bootloader DFU” devices connected to the computer. When the devices have been connected, the “VID”, “PID” and “UID” of each device can be displayed respectively.

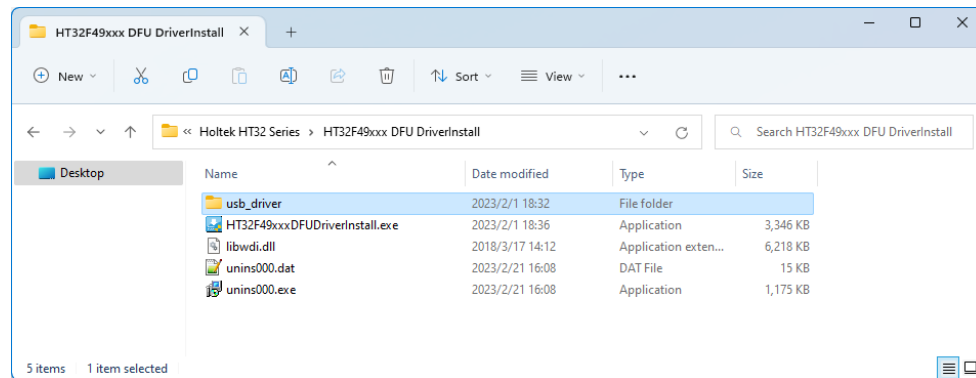


DFU Driver Installation

Click on the “Install driver” button to start the automatic installation of the driver. If the installation is successful, a successful installation message will be displayed. If the installation is failed, an error message will be displayed.

3.2.2 Manual Installation

When the automatic installation has failed or the user needs to install the driver manually, click on “Extract driver”, a driver installation package, the “usb_driver” folder, will be generated in the installation path.

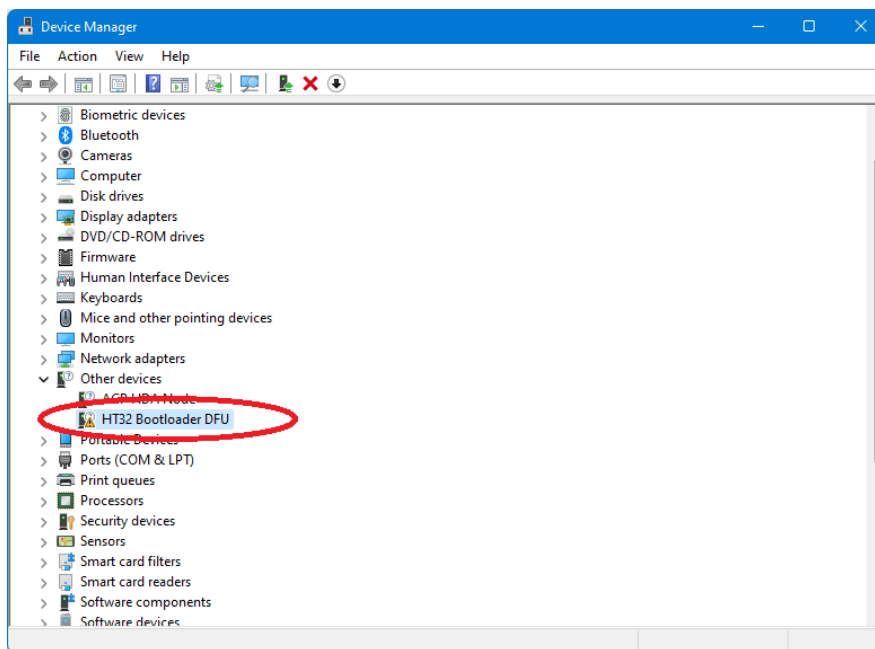


Driver Installation Package Location

This installation package is only available for the currently running operating system. If it is applied to other operating systems, the installation may be failed.

The procedures of manual installation are as follows, which takes the Windows 11 as an example.

- Open “Device Manager”.

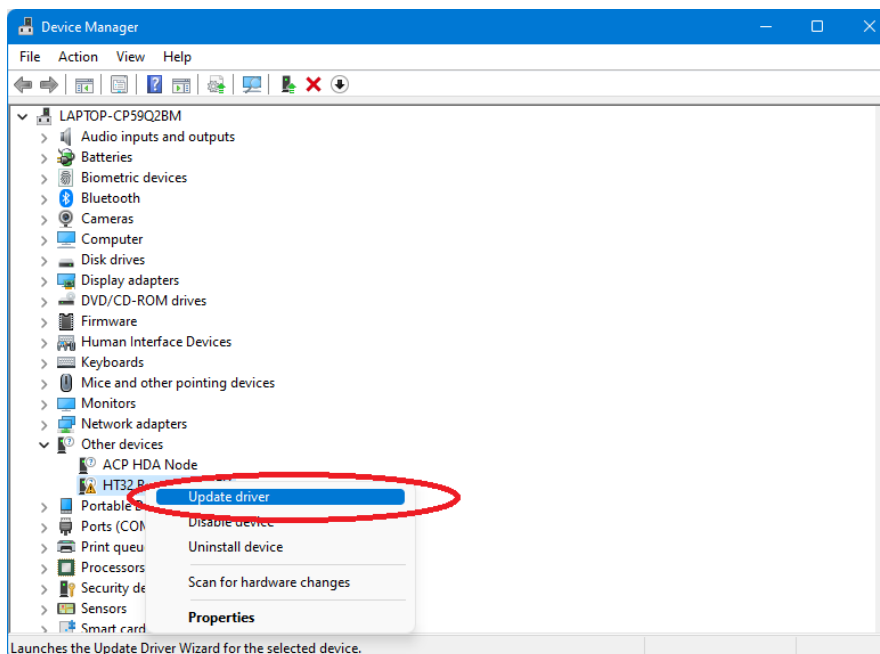


Device Manager

First make sure that the “HT32 Bootloader DFU” device has been correctly connected to the computer. After this, the “Device Manager” will scan the device “HT32 Bootloader DFU” without driver installed.

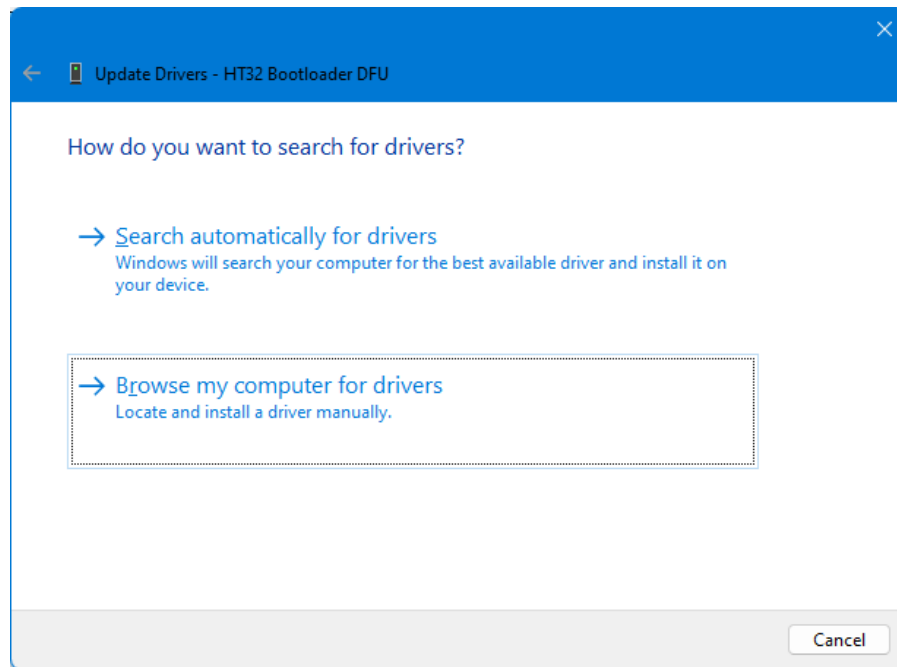
If the device “HT32 Bootloader DFU” is not found, rescan it. Click on the “Device Manager”- “Action” menu and select “Scan for hardware changes”.

- Right-click on the “HT32 Bootloader DFU” to select “Update driver”.



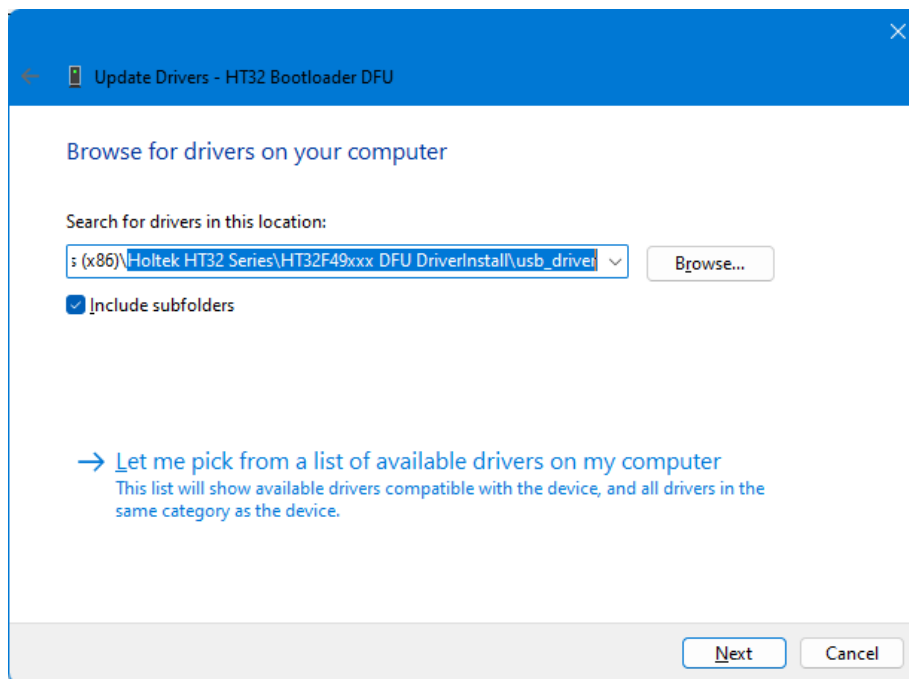
Update Driver

- Select “Browse my computer for drivers”.



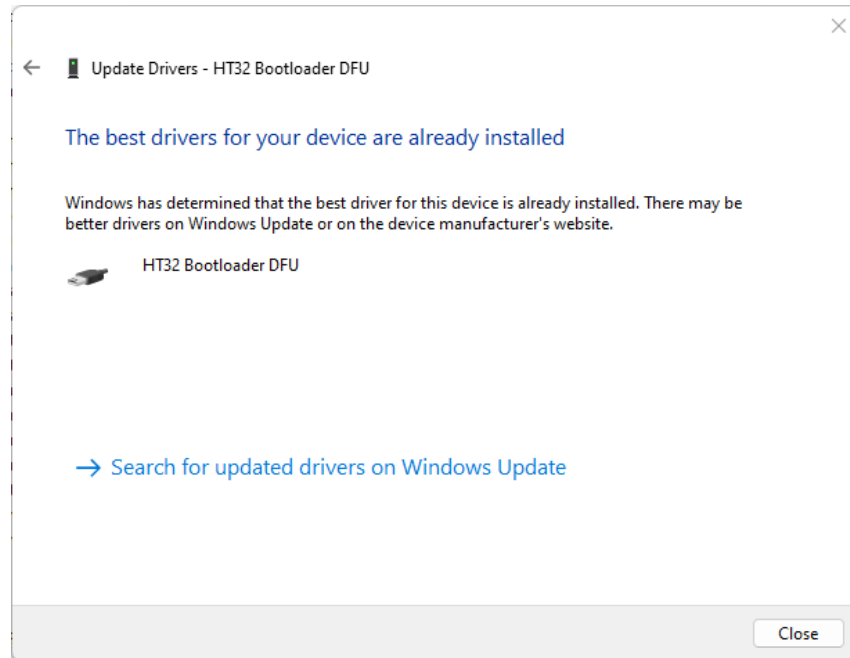
Browse My Computer for Drivers

- Correctly select the driver location. Click on “Extract driver” to generate a driver installation package, the “usb_driver” folder. Then click on “Next”.



Select Driver

- Wait for the driver installation to complete. When the installation has completed, click on “Close”.
The manual installation of the driver has completed.



Complete Installation

4. User Interface

4.1 Device Connection Page

In this section, users can select a corresponding connection method, UART or DFU.

4.1.1 UART Connection

After selecting the UART connection, select a serial port and complete relevant settings, as shown in the following figure. Ensure that the device to be operated is correctly connected to the selected serial port.

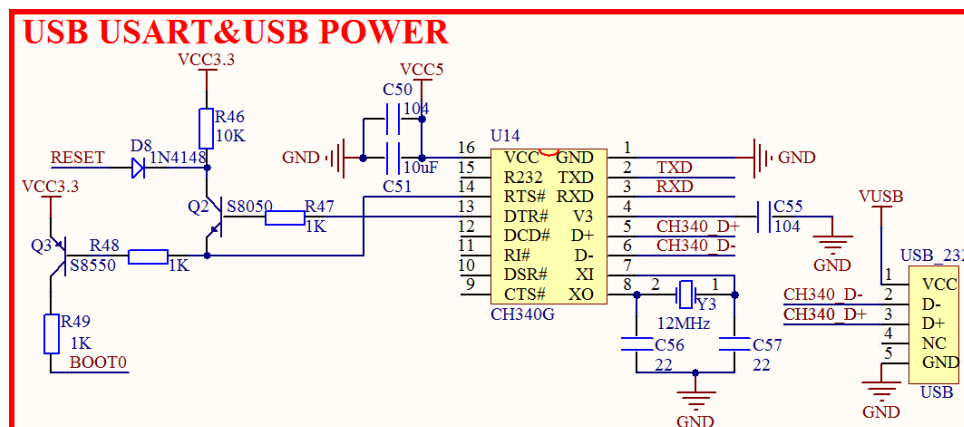


UART Connection Interface

When the “Boot Switch” is set to “Manual”, the device should be reset manually to restart the “BootLoader” program in device. If the device supports automatic connection circuitry, the device can control the reset by controlling the DTR and RTS signals. The supported control mode of the current device can be selected in “Boot Option”.

When the settings have completed, click on “Next”. If the connection is successful, the software will jump to the next page. If the connection is failed, an error message will be displayed.

The USB serial port automatic connection circuit can be designed with reference to the following figure.



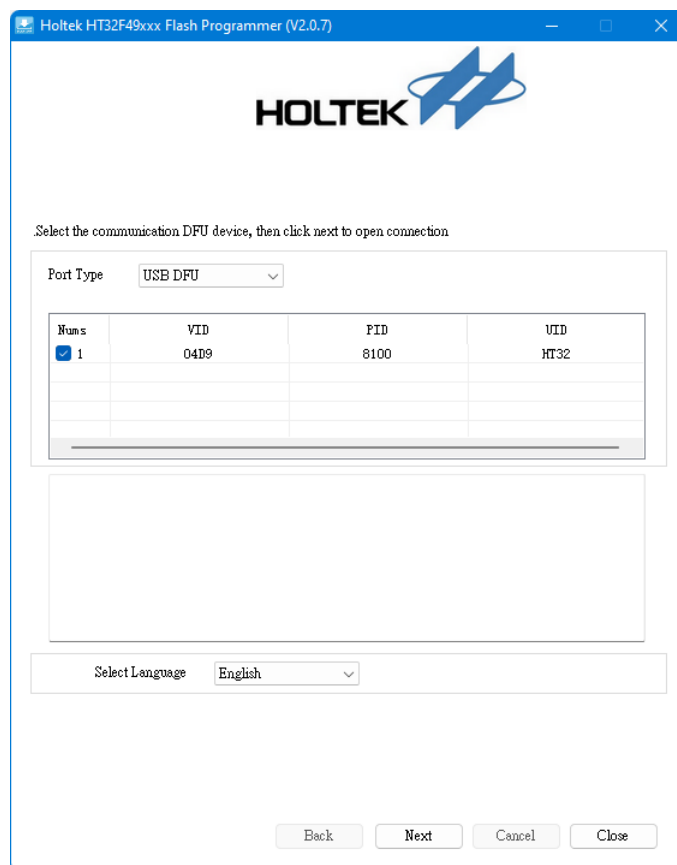
USB Serial Port Auto Connection Diagram

The combination of Q2 and Q3 constitutes the automatic connection circuit of the development board, which only requires to set DTR low and set RTS high into the Bootloader using the ISP software. In this case, auto connection can be implemented without manually setting B0 and pressing RESET key. Here the RESET is the reset signal of the development board, whereas BOOT0 is B0 signal of the boot mode.

The following is the implementation process of automatic connection circuit when BOOT1 is low. First, the DTR is controlled to output a low level using ISP, DTR# output is high, then set RTS high, RTS# output is low. In this case, Q3 is turned on to pull high the BOOT0 (BOOT0=1) and Q2 is also turned on. This causes the device reset pin to be pulled low to implement reset. Then, after a delay of 100ms, the DTR is controlled to output a high level using ISP, DTR# output is low level, and RTS maintains high, RTS# continues to be low level. In this case, Q2 is no longer on which makes the device reset pin turning high to end reset. However BOOT0 will remain high and the system will enter the Bootloader mode, then ISP starts to connect and download the code.

4.1.2 DFU Connection

After selecting the DFU connection, select a DFU device to be connected, as shown in the following figure. Ensure that the device to be operated is correctly connected to the corresponding USB port of PC.



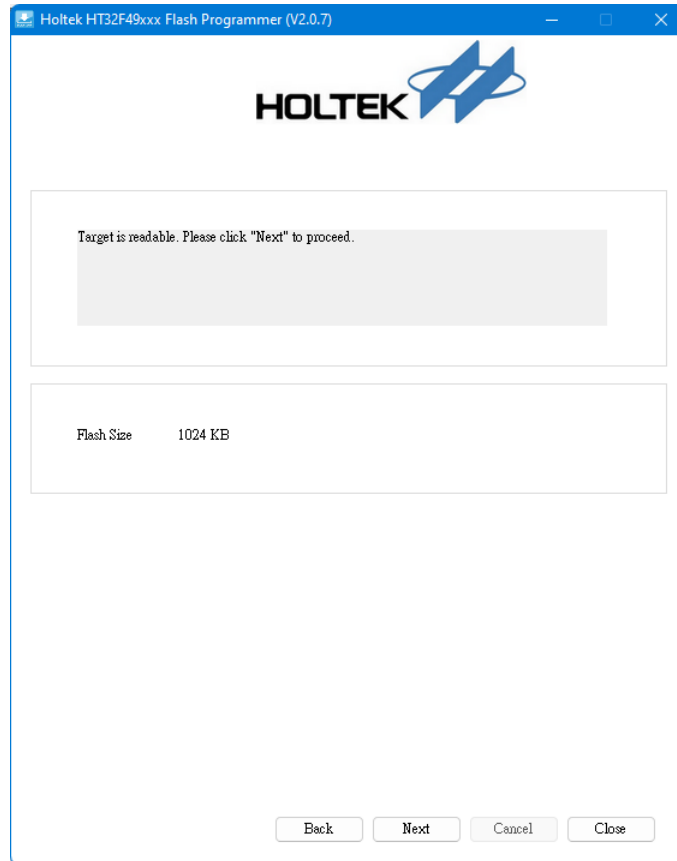
DFU Connection Interface

The software will automatically obtain and display the relevant information of the DFU device, including vendor ID (VID), product ID (PID) and product serial number (UID).

After selecting the connected DFU device, click on “Next”. If the connection is successful, the software will jump to the next page. If failed, an error message will be displayed.

4.2 Flash Status Page

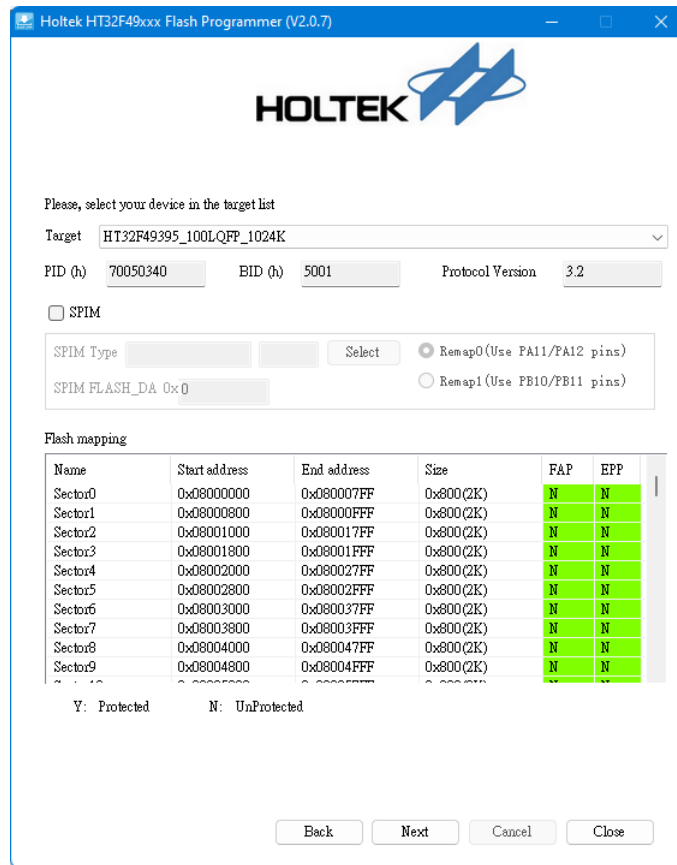
When the connection has been established, the Flash status will be displayed on this page, as shown in the following figure. If the “Access protection” function is enabled, the device will restrict the use of some functions, it is only allowed to use Firmware CRC function, Flash CRC and disable “Access protection” function.



Flash Status Interface

4.3 Device Information Page

This page shows device information such as target device, PID, BID, protocol version, Flash mapping and Flash protection status.



Please, select your device in the target list

Target:

PID (h): BID (h): Protocol Version:

☐ SPIM

SPIM Type: ☒ Remap0 (Use PA11/PA12 pins) ☐ Remap1 (Use PB10/PB11 pins)

SPIM FLASH_DA:

Flash mapping

Name	Start address	End address	Size	FAP	EPP
Sector0	0x08000000	0x080007FF	0x800 (2K)	N	N
Sector1	0x08000800	0x08000FFF	0x800 (2K)	N	N
Sector2	0x08001000	0x080017FF	0x800 (2K)	N	N
Sector3	0x08001800	0x08001FFF	0x800 (2K)	N	N
Sector4	0x08002000	0x080027FF	0x800 (2K)	N	N
Sector5	0x08002800	0x08002FFF	0x800 (2K)	N	N
Sector6	0x08003000	0x080037FF	0x800 (2K)	N	N
Sector7	0x08003800	0x08003FFF	0x800 (2K)	N	N
Sector8	0x08004000	0x080047FF	0x800 (2K)	N	N
Sector9	0x08004800	0x08004FFF	0x800 (2K)	N	N

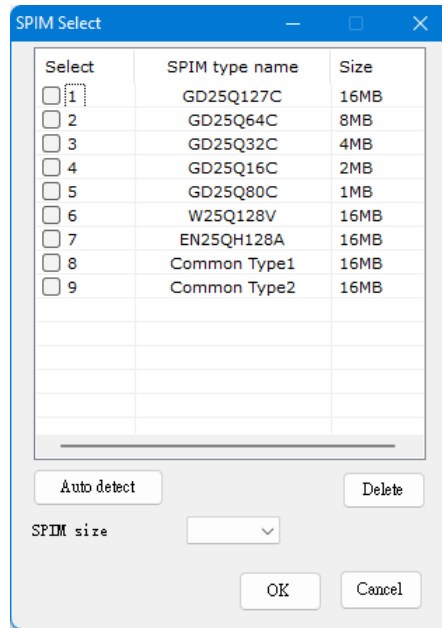
Y: Protected N: UnProtected

Device Information Interface

If an SPIM is connected, check “SPIM” and select “SPIM Type”. The SPIM size depends on the “SPIM Type”. If the SPIM encryption is required, set “SPIM FLASH_DA”.

Then all sectors of the main Flash and SPIM will be automatically displayed in the Flash mapping area.

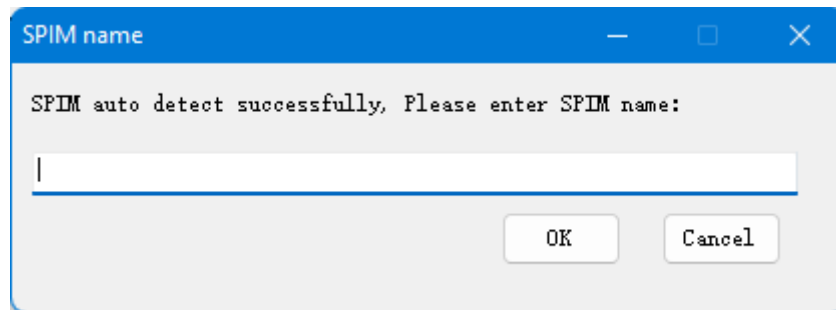
- Check “SPIM”
If an SPIM is connected, check “SPIM” before the software allows SPIM operations.
- Uncheck “SPIM”
The software will not allow the SPIM operations.
- SPIM Type
The SPIM type can be selected by the “Select” button.
Click on the “Select” button, the following dialog box will pop up.



SPIM Selection Interface

Auto detect: It will automatically detect whether the SPIM meets the operation requirements of this software. Auto detect will overwrite some data of SPIM, use it with caution.

When the detection is successful, a dialog box to enter SPIM name will pop up.

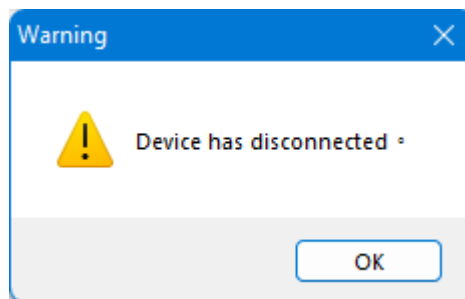


Enter SPIM Name

Click on “OK” to add the detected SPIM to the SPIM list.

Click on “Cancel” to cancel this auto detect.

If the detection is failed, a failure dialog box will pop up.



SPIM Detection Failed

4.4.1 Erase

- Click on “All” to erase the whole memory including SPIM.
- Click on “Sectors” to customize the sectors to be erased. Click on “...” to select the sectors to be erased in the pop-up dialog box.

Memory Mapping

Name	Start address	End address	Size	FAP	EPP
<input type="checkbox"/> Sector0	0x08000000	0x080007FF	0x800(2K)	N	N
<input type="checkbox"/> Sector1	0x08000800	0x08000FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector2	0x08001000	0x080017FF	0x800(2K)	N	N
<input type="checkbox"/> Sector3	0x08001800	0x08001FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector4	0x08002000	0x080027FF	0x800(2K)	N	N
<input type="checkbox"/> Sector5	0x08002800	0x08002FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector6	0x08003000	0x080037FF	0x800(2K)	N	N
<input type="checkbox"/> Sector7	0x08003800	0x08003FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector8	0x08004000	0x080047FF	0x800(2K)	N	N
<input type="checkbox"/> Sector9	0x08004800	0x08004FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector10	0x08005000	0x080057FF	0x800(2K)	N	N
<input type="checkbox"/> Sector11	0x08005800	0x08005FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector12	0x08006000	0x080067FF	0x800(2K)	N	N
<input type="checkbox"/> Sector13	0x08006800	0x08006FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector14	0x08007000	0x080077FF	0x800(2K)	N	N
<input type="checkbox"/> Sector15	0x08007800	0x08007FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector16	0x08008000	0x080087FF	0x800(2K)	N	N
<input type="checkbox"/> Sector17	0x08008800	0x08008FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector18	0x08009000	0x080097FF	0x800(2K)	N	N
<input type="checkbox"/> Sector19	0x08009800	0x08009FFF	0x800(2K)	N	N

☐ Select All

OK

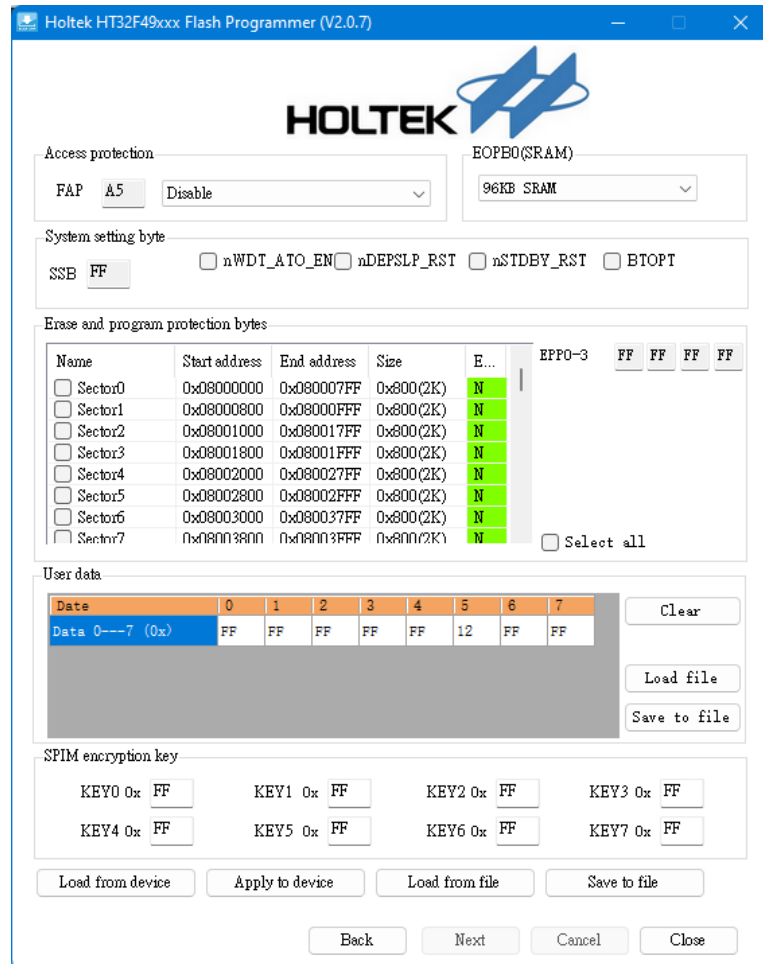
Cancel

Sector Erase Selection

4.4.2 Edit User System Data

Select “Edit User system data” and click on “Next” to jump to the “User system data” configuration page.

On this page, users can configure the “User system data” using the graphical interface.



HOLTEK

Access protection: FAP

System setting byte: SSB ☐ nWDT_ATO_EN ☐ nDEPSLP_RST ☐ nSTDBY_RST ☐ BTOPT

Erase and program protection bytes:

Name	Start address	End address	Size	E...	EPP0-3
<input type="checkbox"/> Sector0	0x08000000	0x080007FF	0x800(2K)	N	FF FF FF FF
<input type="checkbox"/> Sector1	0x08000800	0x08000FFF	0x800(2K)	N	
<input type="checkbox"/> Sector2	0x08001000	0x080017FF	0x800(2K)	N	
<input type="checkbox"/> Sector3	0x08001800	0x08001FFF	0x800(2K)	N	
<input type="checkbox"/> Sector4	0x08002000	0x080027FF	0x800(2K)	N	
<input type="checkbox"/> Sector5	0x08002800	0x08002FFF	0x800(2K)	N	
<input type="checkbox"/> Sector6	0x08003000	0x080037FF	0x800(2K)	N	
<input type="checkbox"/> Sector7	0x08003800	0x08003FFF	0x800(2K)	N	

☐ Select all

User data:

Date	0	1	2	3	4	5	6	7
Data 0---7 (0x)	FF	FF	FF	FF	FF	12	FF	FF

SPIM encryption key:

KEY0 0x	FF	KEY1 0x	FF	KEY2 0x	FF	KEY3 0x	FF
KEY4 0x	FF	KEY5 0x	FF	KEY6 0x	FF	KEY7 0x	FF

User System Data Interface

The software supports obtaining the “User system data” value from the device or file and displaying the value. After editing, apply to device or save to file.

- Access protection

The access protection status is displayed. The access protection of the memory cannot be set here.

Enabled: FAP is 0xFF.

Disabled: FAP is 0xA5.

When the access protection is enabled, neither the Flash memory or user system data can be read, unless the access protection is disabled. When the access protection is disabled, both the main Flash memory and user system data will be erased.

- System setting byte

nWDT_ATO_EN:

Checked – Watchdog self-enabled is enabled.

Unchecked – Watchdog self-enabled is disabled.

nDEPSLP_RST:

Checked – Reset occurs when entering Deep Sleep mode.

Unchecked – No reset occurs when entering Deep Sleep mode.

nSTDBY_RST:

Checked – Reset occurs when entering Standby mode.

Unchecked – No reset occurs when entering Standby mode.

BTOPT:

Checked – when the device is set to boot from Flash memory bank 1 or bank 2. The device will boot from bank 1 if there is no startup program in bank 2, otherwise, it will boot from bank 2.

Unchecked – when the device is set to boot from Flash memory (default value), it starts from bank 1.

- EOPB0 (SRAM)

224KB SRAM – SRAM 224KB.

96KB SRAM – SRAM 96KB.

- Erase and program protection bytes

Choose which sectors need to be protected from erase and program.

Erase and program protection bytes

Name	Start address	End address	Size	E...
<input type="checkbox"/> Sector0	0x08000000	0x080007FF	0x800(2K)	N
<input type="checkbox"/> Sector1	0x08000800	0x08000FFF	0x800(2K)	N
<input type="checkbox"/> Sector2	0x08001000	0x080017FF	0x800(2K)	N
<input type="checkbox"/> Sector3	0x08001800	0x08001FFF	0x800(2K)	N
<input type="checkbox"/> Sector4	0x08002000	0x080027FF	0x800(2K)	N
<input type="checkbox"/> Sector5	0x08002800	0x08002FFF	0x800(2K)	N
<input type="checkbox"/> Sector6	0x08003000	0x080037FF	0x800(2K)	N
<input type="checkbox"/> Sector7	0x08003800	0x08003FFF	0x800(2K)	N

EPP0-3

☐ Select all

Erase and Program Protection Bytes

EPP0:

Controls the erase and program protection of sectors ranging from Flash 1K to 32K.

EPP1:

Controls the erase and program protection of sectors ranging from Flash 33K to 64K.

EPP2:

Controls the erase and program protection of sectors ranging from Flash 65K to 96K.

EPP3:

Bits 0~6 control the erase and program protection of sectors ranging from Flash 97K to 124K;

Bit 7 controls the erase and program protection of all sectors after Flash 124K, including SPIM.

- User data

User data

Date	0	1	2	3	4	5	6	7
Data 0---7 (0x)	FF	FF	FF	FF	FF	FF	FF	FF

User Data

Clear: Clear all user system data to 0xFF.

Load file: Load the user system data file to the table.

Save to file: Save the user system data in the table to the file.

- SPIM encryption key

Users can set the SPIM encryption key value.

SPIM encryption key

KEY0 0x	FF	KEY1 0x	FF	KEY2 0x	FF	KEY3 0x	FF
KEY4 0x	FF	KEY5 0x	FF	KEY6 0x	FF	KEY7 0x	FF

SPIM Encryption Key

- Load from device

Read the user system data from the device and update it to the interface for display.

- Apply to device

Save the user system data settings to the device.

- Load from file

Read the content of user system data from user system data file and update it to the interface for display.

- Save to file

Save the user system data settings to a file.

4.4.3 Download to Device

sLib Status: DISABLE Start sector

Remaining usage times: 253 DATA start sector

Password 0x End sector

No.	File Name	File Size	Address Range(0x)

Erase option
☐ Enable slib before download

☐ Optimize(Remove some FFs) ☐ Verify after download

☐ Write user serial number ☐ Jump to the user program

Address 0x Current SN 0x Increase step 0x

☐ Apply User system data ...

☐ Enable Access protection after Download

Download to Device

- sLib settings
 - ♦ sLib Status
Status of the sLib connected to the device, “DISABLE” or “ENABLE”.
 - ♦ Remaining usage times
It means the remaining usage times of sLib. It can be used up to 256 times and will be reduced after each usage. When the remaining usage times is 0, the sLib function will not be available.
 - ♦ Password
Enter the enable password when the sLib function is enabled. Enter the disable password when the sLib function is disabled.
 - ♦ Start sector
The start sector of the sLib area. The instruction area is from the “Start sector” to the “DATA start sector” (not including the “DATA start sector”). When the sLib is enabled, the data in this area cannot be erased, written or read.
 - ♦ DATA start sector
The start sector of the sLib data area. This data area is from “DATA start sector” to “End sector” (including the “End sector”). After the sLib is enabled, the data in this area cannot be erased and written, but can be read. When the “DATA start sector” is set to “none”, it means no data area.
 - ♦ End sector
The end sector of the sLib area.
- Other download settings
Three file types are supported: bin (binary), hex (hexadecimal) and s19/srec (Motorola S file).

No.	File Name	File Size	Address Range(0x)	Add
1	led_toggle.bin	2692	08000000—08000A83	Delete

Download File Selection

If adding a bin file, a download address can be chosen.

If adding a hex or s19/srec file, the download address is obtained from the loaded file.

- ♦ Select “Erase the sectors of file size” to erase sectors where the downloaded file is located before download.
Select “No Erase”, no erase operation will be performed before download.
Select “Global Erase” to erase the whole memory including SPIM before download.
- ♦ Check “Enable sLib before download” to enable sLib before download. Users need to enter the password, start sector, data start sector and end sector to enable sLib. Refer to the sLib settings on the previous page for details.
- ♦ Check “Verify after download” to run the verify program after downloading to verify whether the download data is correct.
- ♦ Check “Jump to the user program” to run the program directly after the download is completed.
- ♦ Check “Optimize (Remove some FFs)” to optimize the download process, skip the 0xFF field of the file and speed up the download.
- ♦ Check “Write user serial number” and download the serial number to the device after download.
- ♦ Address: The address where the serial number is programmed into the memory.

- ♦ Current SN: The serial number for the current programming.
- ♦ Increase step: This is the amount added to the current serial number after each serial number is programmed.
- ♦ Check “Apply User system data” to load the user system data file after download and set the values to the device.
- ♦ Check “Enable Access protection after Download” to enable access protection after download.

4.4.4 Disable sLib

To disable the sLib, enter the disable password which is the enable password when the sLib was last enabled.

☐ Download to device
 ☒ **Disable sLib**

sLib Status: DISABLE	Start sector	<input type="text"/>
Remaining usage times: 253	DATA start sector	<input type="text"/>
Password 0 <input type="text"/>	End sector	<input type="text"/>

Disable sLib

When the sLib is disabled successfully, the whole device will be erased.

4.4.5 Upload from Device

Three file types are supported: bin (binary), hex (hexadecimal) and s19/src (Motorola S file). Select the upload sectors.

Memory Mapping

Name	Start address	End address	Size	FAP	EPP
<input checked="" type="checkbox"/> Sector0	0x08000000	0x080007FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector1	0x08000800	0x08000FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector2	0x08001000	0x080017FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector3	0x08001800	0x08001FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector4	0x08002000	0x080027FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector5	0x08002800	0x08002FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector6	0x08003000	0x080037FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector7	0x08003800	0x08003FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector8	0x08004000	0x080047FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector9	0x08004800	0x08004FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector10	0x08005000	0x080057FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector11	0x08005800	0x08005FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector12	0x08006000	0x080067FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector13	0x08006800	0x08006FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector14	0x08007000	0x080077FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector15	0x08007800	0x08007FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector16	0x08008000	0x080087FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector17	0x08008800	0x08008FFF	0x800 (2K)	N	N
<input type="checkbox"/> Sector18	0x08009000	0x080097FF	0x800 (2K)	N	N
<input type="checkbox"/> Sector19	0x08009800	0x08009FFF	0x800 (2K)	N	N

☐ Select All

Upload from Device

4.4.6 Firmware CRC

This function is used to calculate the CRC code and compare it with the loaded file to confirm the correctness of the downloaded files. This function can be used in the Flash access protection state.

First select the file to be compared.

No.	File Name	File Size	Address Range(0x)	Add
1	led_toggle.bin	2692	08000000—08000A83	Delete

Firmware CRC

“Sector fill”: The Firmware CRC is performed in unit of sectors. What is filled in here is the download data which is not filled in the sector part. Generally, it is “FF”.

4.4.7 Flash CRC

This function is used to calculate the Flash CRC value, including main Flash memory and SPIM. This function can be used in the Flash access protection state.

Flash CRC

Start sector: End sector:

Flash CRC Sector Selection

The start sector and end sector of the Flash memory must be set up.

4.4.8 Protection

- Select “ENABLE” - “Access protection” to enable the Flash memory access protection. The whole Flash memory will be protected from access when executed correctly.
- Select “DISABLE” - “Access protection” to disable the access protection of the whole Flash memory.
- Select “ENABLE” - “Erase and program protection” and click “...”, select the sectors to enable erase and program protection in the dialog box that is popped up.

Memory Mapping

Name	Start address	End address	Size	FAP	EPP
<input type="checkbox"/> Sector0	0x08000000	0x080007FF	0x800(2K)	N	N
<input type="checkbox"/> Sector1	0x08000800	0x08000FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector2	0x08001000	0x080017FF	0x800(2K)	N	N
<input type="checkbox"/> Sector3	0x08001800	0x08001FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector4	0x08002000	0x080027FF	0x800(2K)	N	N
<input type="checkbox"/> Sector5	0x08002800	0x08002FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector6	0x08003000	0x080037FF	0x800(2K)	N	N
<input type="checkbox"/> Sector7	0x08003800	0x08003FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector8	0x08004000	0x080047FF	0x800(2K)	N	N
<input type="checkbox"/> Sector9	0x08004800	0x08004FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector10	0x08005000	0x080057FF	0x800(2K)	N	N
<input type="checkbox"/> Sector11	0x08005800	0x08005FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector12	0x08006000	0x080067FF	0x800(2K)	N	N
<input type="checkbox"/> Sector13	0x08006800	0x08006FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector14	0x08007000	0x080077FF	0x800(2K)	N	N
<input type="checkbox"/> Sector15	0x08007800	0x08007FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector16	0x08008000	0x080087FF	0x800(2K)	N	N
<input type="checkbox"/> Sector17	0x08008800	0x08008FFF	0x800(2K)	N	N
<input type="checkbox"/> Sector18	0x08009000	0x080097FF	0x800(2K)	N	N
<input type="checkbox"/> Sector19	0x08009800	0x08009FFF	0x800(2K)	N	N

☐ Select All

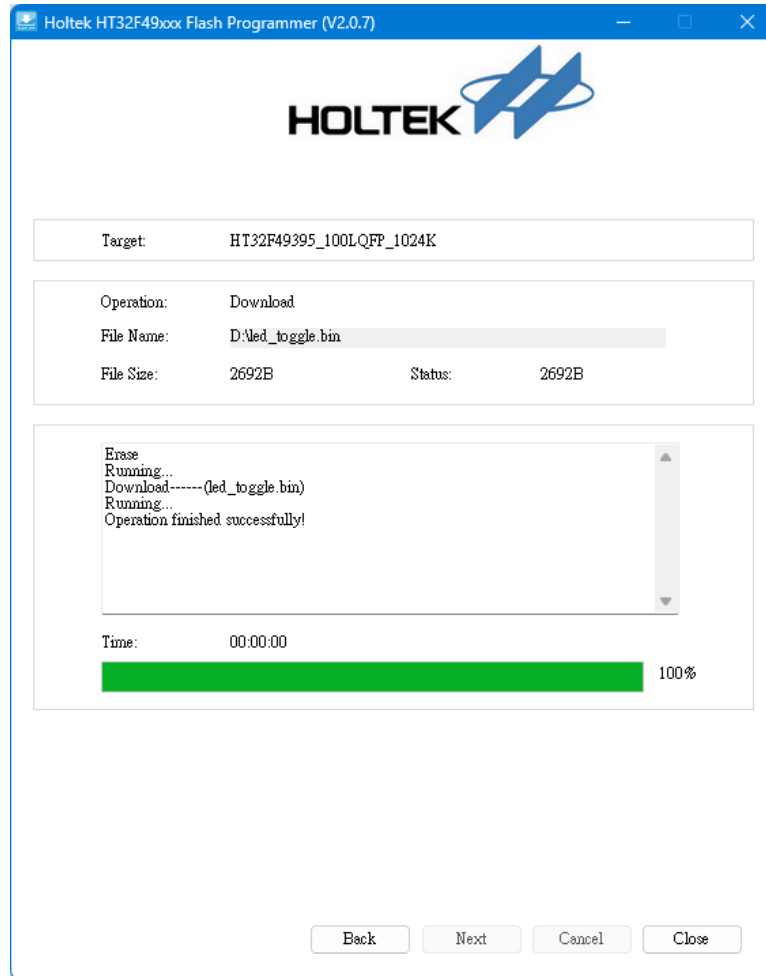
OKCancel

Enable Erase and Program Protection

- Select “DISABLE” - “Erase and program protection” to disable the erase and program protection of the whole Flash memory.

4.5 Operation Progress Page

This page shows information related to the operation progress.



Operation Progress Interface

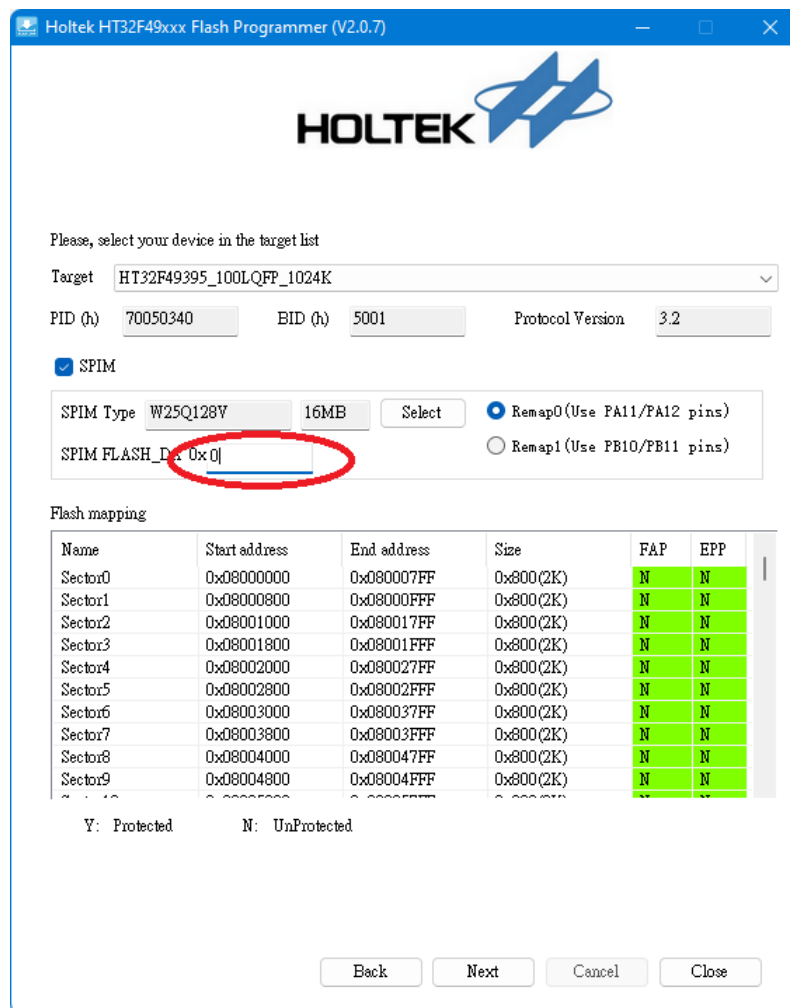
4.6 SPIM Encryption Download

SPIM encryption principle:

When the SPIM encrypted download is required, users must first configure the SPIM FLASH_DA and SPIM encryption key and the key is set in the user system data interface, then perform download operation. In this case, the MCU will encrypt the download original data according to SPIM FLASH_DA and encryption key as well as MCU internal algorithm, then write the encrypted data to the SPIM.

When users want to read the encrypted data in the SPIM, users also need to configure the SPIM FLASH_DA and encryption key. The MCU will decrypt the encrypted data and restore it to the correct original data according to SPIM FLASH_DA and encryption key as well as MCU internal algorithm. When downloading files to SPIM, the following steps can be set to encrypt the download contents.

Step 1: Set the SPIM FLASH_DA.



Please, select your device in the target list

Target: HT32F49395_100LQFP_1024K

PID (h): 70050340 BID (h): 5001 Protocol Version: 3.2

☒ SPIM

SPIM Type: W25Q128V 16MB Select ☒ Remap0 (Use PA11/PA12 pins) ☐ Remap1 (Use PB10/PB11 pins)

SPIM FLASH_DA: 0x0

Flash mapping

Name	Start address	End address	Size	FAP	EPP
Sector0	0x08000000	0x080007FF	0x800(2K)	N	N
Sector1	0x08000800	0x08000FFF	0x800(2K)	N	N
Sector2	0x08001000	0x080017FF	0x800(2K)	N	N
Sector3	0x08001800	0x08001FFF	0x800(2K)	N	N
Sector4	0x08002000	0x080027FF	0x800(2K)	N	N
Sector5	0x08002800	0x08002FFF	0x800(2K)	N	N
Sector6	0x08003000	0x080037FF	0x800(2K)	N	N
Sector7	0x08003800	0x08003FFF	0x800(2K)	N	N
Sector8	0x08004000	0x080047FF	0x800(2K)	N	N
Sector9	0x08004800	0x08004FFF	0x800(2K)	N	N

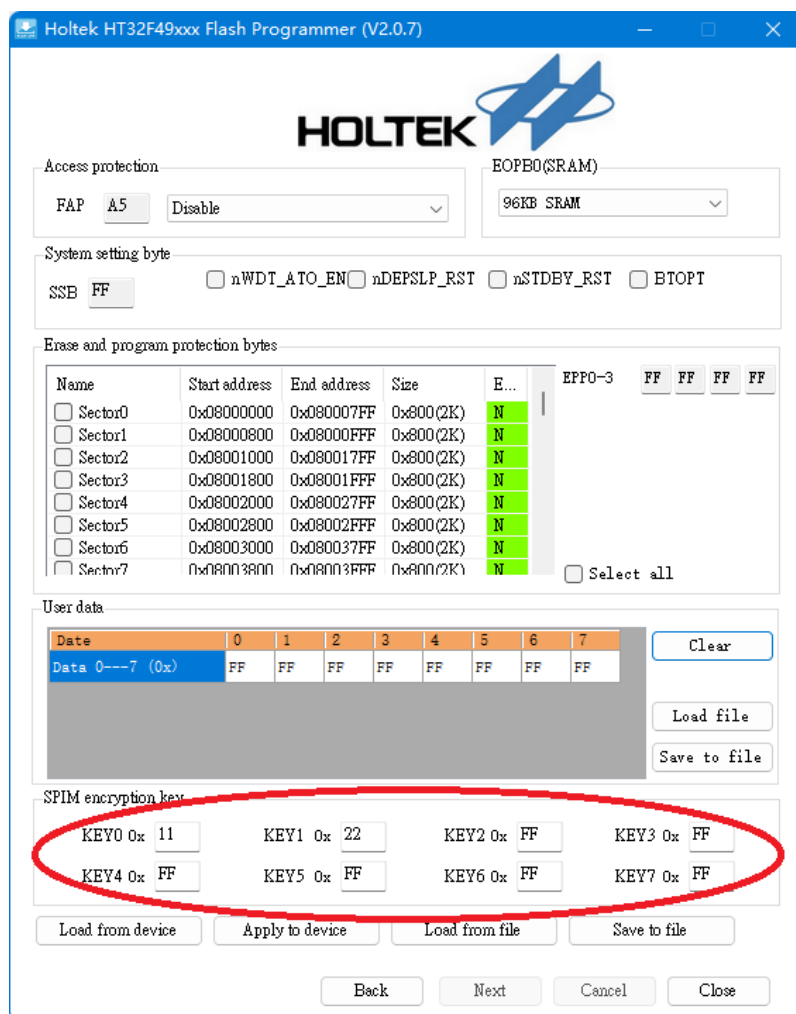
Y: Protected N: UnProtected

Back Next Cancel Close

Encryption Range Configuration

Start address 0x08400000 plus the set “SPIM FLASH_DA” forms the encryption area. If the encryption is not required, it is set to 0.

Step 2: Set the SPIM encryption key on the “User system data” page.



HOLTEK

Access protection
FAP A5 Disable

System setting byte
SSB FF ☐ nWDT_ATO_EN ☐ nDEPSLP_RST ☐ nSTDBY_RST ☐ BTOPT

Erase and program protection bytes

Name	Start address	End address	Size	E...
<input type="checkbox"/> Sector0	0x08000000	0x080007FF	0x800(2K)	N
<input type="checkbox"/> Sector1	0x08000800	0x08000FFF	0x800(2K)	N
<input type="checkbox"/> Sector2	0x08001000	0x080017FF	0x800(2K)	N
<input type="checkbox"/> Sector3	0x08001800	0x08001FFF	0x800(2K)	N
<input type="checkbox"/> Sector4	0x08002000	0x080027FF	0x800(2K)	N
<input type="checkbox"/> Sector5	0x08002800	0x08002FFF	0x800(2K)	N
<input type="checkbox"/> Sector6	0x08003000	0x080037FF	0x800(2K)	N
<input type="checkbox"/> Sector7	0x08003800	0x08003FFF	0x800(2K)	N

EPFO-3 FF FF FF FF

☐ Select all

User data

Date	0	1	2	3	4	5	6	7
Data 0---7 (0x)	FF	FF	FF	FF	FF	FF	FF	FF

Clear

Load file

Save to file

SPIM encryption key

KEY0 0x 11	KEY1 0x 22	KEY2 0x FF	KEY3 0x FF
KEY4 0x FF	KEY5 0x FF	KEY6 0x FF	KEY7 0x FF

Load from device Apply to device Load from file Save to file

Back Next Cancel Close

SPIM Encryption Key Configuration

This is the encryption/decryption key for downloading and reading data in the encryption range of SPIM. When the access protection is disabled, the key will be erased.

Step 3: Download the files to SPIM to implement encryption download.

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.