**GigaDevice Semiconductor Inc.**

**GD32Fxxx**

**ARM® Cortex™-M3 32-bit MCU**

**USB IAP User Mannual**

(Jan. 2019)

# Table of Contents

[Table of Contents 2](#_Toc874494)

[1. Introduction 3](#_Toc874495)

[2. USB IAP Firmware Library File Structure 3](#_Toc874496)

[3. IAP implementation environment 3](#_Toc874497)

[4. Introduction to IAP 4](#_Toc874498)

[5. DFU for IAP implementation steps 5](#_Toc874499)

[5.1. DFU tool installation 5](#_Toc874500)

[5.2. IAP driven firmware generation and loading 5](#_Toc874501)

[5.3. User APP generation 6](#_Toc874502)

[5.4. User APP upgrade 8](#_Toc874503)

[6. Custom HID for IAP implementation steps 9](#_Toc874504)

[6.1. HID IAP tool installation 9](#_Toc874505)

[6.2. IAP driven firmware generation and loading 9](#_Toc874506)

[6.3. User APP generation 9](#_Toc874507)

[6.4. User APP upgrade 9](#_Toc874508)

[7. MSC (USB HOST) for IAP implementation steps 10](#_Toc874509)

[7.1. IAP driven firmware generation and loading 10](#_Toc874510)

[7.2. User APP generation 10](#_Toc874511)

[7.3. User APP upgrade 10](#_Toc874512)

[8. MSC (USB Device) for IAP implementation steps 11](#_Toc874513)

[8.1. IAP driven firmware generation and loading 11](#_Toc874514)

[8.2. User APP generation 11](#_Toc874515)

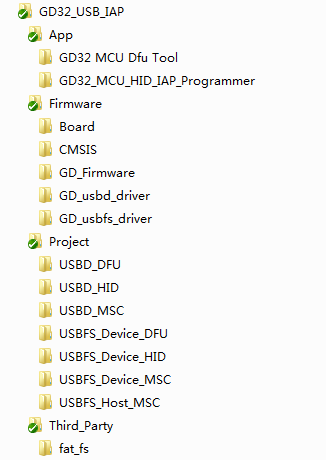
[8.3. User APP upgrade 11](#_Toc874516)

[9. Revision history 12](#_Toc874517)

# Introduction

With the increasing demand for firmware upgrade, GD32 provides a flexible upgrade mode: IAP (In Application Program). It can be implemented by USB DFU class, USB Custom HID class or USB MSC class with relevant demo software tools.

# USB IAP Firmware Library File Structure



**②**

**④**

**③**

**①**

The tree structure of USB IAP firmware library folder is shown in the figure above. In this figure, PC-side upper computer program is in App folder, including DFU and HID IAP upper computer, as shown in ① of this figure. The driver layer code of USB IAP firmware library is in Firmware folder, including Board (bottom driver of development board), GD\_Firmware (general peripheral driver), GD\_usbd\_driver (USBD peripheral driver), GD\_usbfs\_driver (USBFS peripheral driver), etc., as shown in ② of this Figure. The application layer code is in Project folder. As shown in ③ of this figure. The third-party library is located in the Third\_Party folder, which contains the source code of the FatFs filesystem, as shown in ④ of this Figure.

# IAP implementation environment

**Development board:** GD32F450I / GD32F350R / GD32F307C / GD32E103V / GD32F207I / GD32F107C / GD32F103B / GD32F103C / GD32F150R / GD32F303E–EVAL

**Development library:** GD32Fxxx\_Firmware\_Library

**Integrated development environment:** IAR v7.40 or MDK-ARM v4.74/v5.26

**DFU demo software:** GD32 MCU Dfu Tool\_v3.7.1.3145

**HID\_IAP demo software:** GD32 MCU HID IAP Programmer \_V2.0.1.3165

The implementation process is carried out on GD32F4 series, but the IAP function implemented by USB DFU is not limited to GD32F4 series.

# Introduction to IAP

IAP (In Application Program) mainly provides users with a more flexible way of firmware upgrade, which can be defined according to the application needs when and where the firmware upgrade happens. Before introducing the steps, it is necessary to have a certain understanding of IAP principles. Combined with the figure below (for reference only, the practical application of IAP is more flexible), in the application of IAP, user programs and IAP drivers are located in different storage areas. In the application process, the user firmware is loaded into a fixed position by using IAP driver to complete the upgrade.

Among them, the IAP driver first burns and solidifies, and will not change with the upgrade of the user program. After the upgrade of user firmware, under the guidance of IAP driver, jump to the corresponding Flash location of user firmware and execute user program.



# DFU for IAP implementation steps

## DFU tool installation

Install the DFU demo tool GD32 MCU Dfu Tool\_v3.7.1.3145, which mainly provides driver and Dfu Tool. Refer to “GigaDevice Dfu Tool User Manual” for details of Dfu Tool (installation steps, usage instructions).

## IAP driven firmware generation and loading

Using the USB DFU routine to generate IAP driver firmware (folder path: GD32\_USB\_IAP\Project\USBFS\_Device\_DFU)

Use burning tools to burn to the development board. After this routine is restarted, if the UserKey button is not pressed on the board or there is no valid user application, it will enter the USB DFU (IAP) mode and wait for upgrade; otherwise, it will enter the user application.

The start address corresponding to the user app is defined at 0x08004000. Users can also customize the user code addresses. The following points need to be noted (Taking GD32F4xx series as an example):

1. Forbid defining user code address before 0x08004000(0x0~0x08003FFF belongs to Sector0, has been opened as IAP driver area)

2. The sector to which the custom address belongs will be erased when upgraded, even if the defined address is not located at the first address of the corresponding sector.

3. Custom addresses need to be 512-byte aligned

4. Custom addresses need to be consistent with user firmware loading addresses and user interrupt vector table addresses.

In the code that implements jumping to the user program, the stack head address which the first 4 bytes pointed to in the user interrupt vector table is assigned to MSP. The second 4 bytes are reset interrupt vectors, pointing to the execution header address.

When developing IAP driver, it is necessary to avoid PC pointer jumping to user program area. At the same time, fully consider the size of Stack & Heap to avoid the phenomenon that the USB DFU can be identified but can not work normally.

## User APP generation

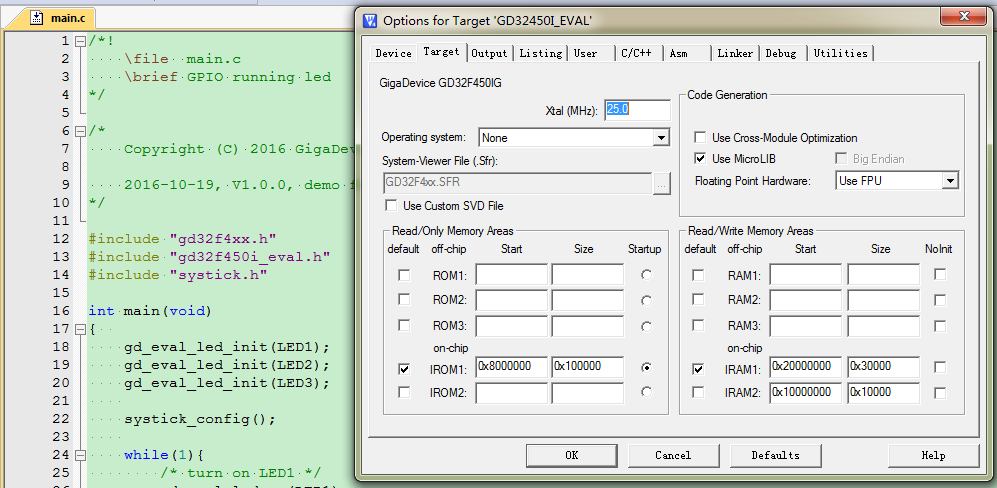
There are two main changes in user firmware required for IAP upgrade:

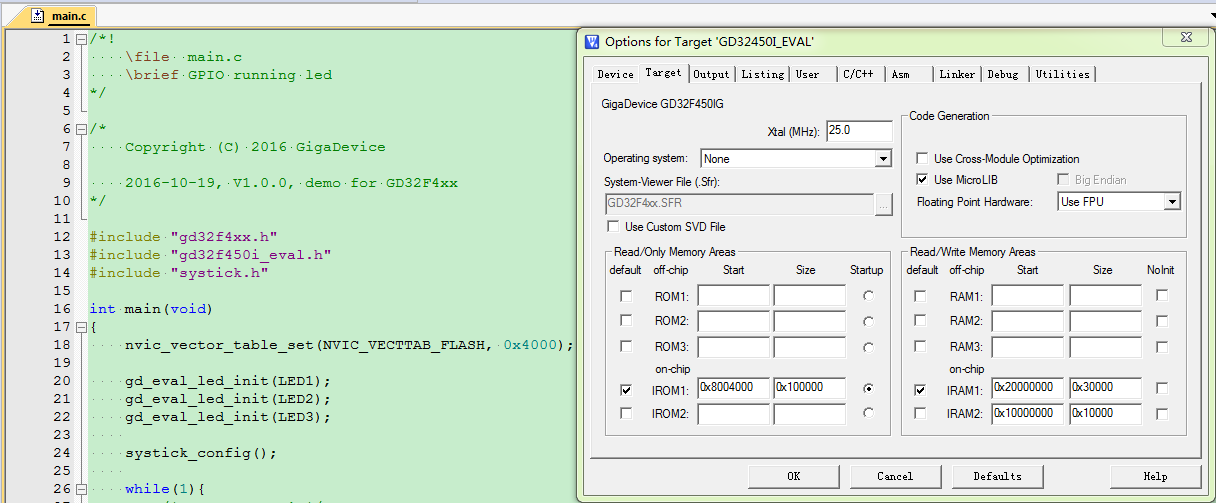
1. In link tools, load address of program

2. Start address of interrupt vector table

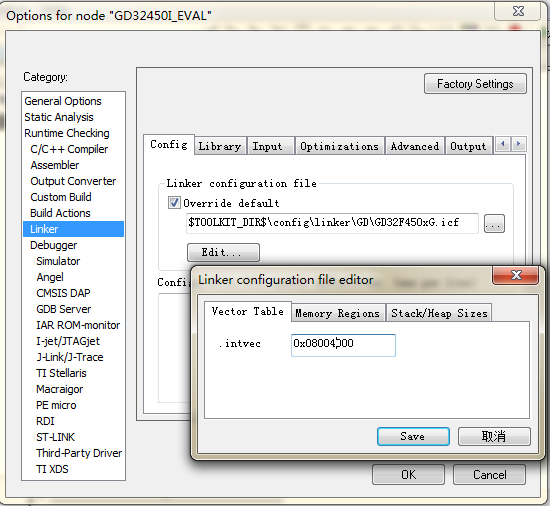
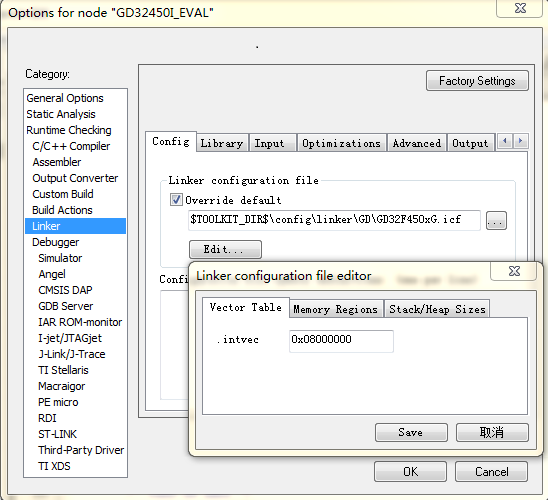
These two changes need to be consistent with the user area header address defined in IAP.

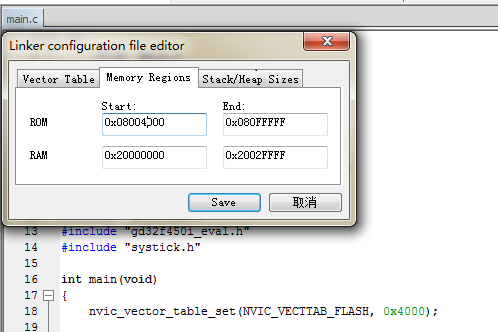
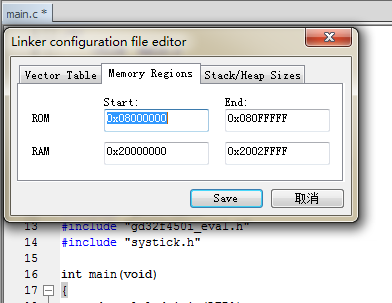
In this IAP driver, the user firmware loading address is 0x4000 relative to 0x08000000 offset address. In MDK, the changes are compared before and after as shown in the following figure.





In IAR, the change steps are as follows.

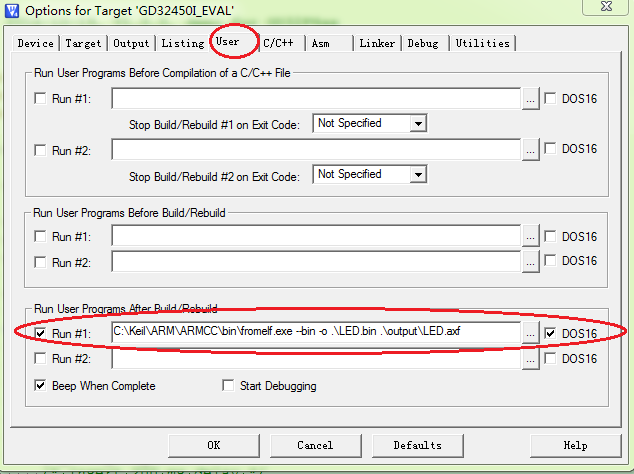




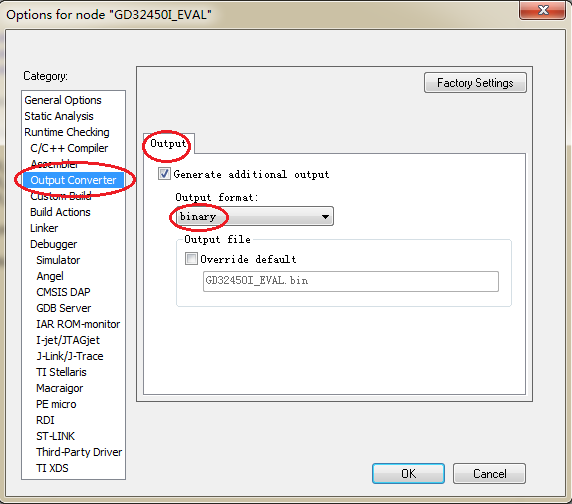
After modification, generate the hex file.

User also can generate the bin file, the steps are as follows:

In keil:



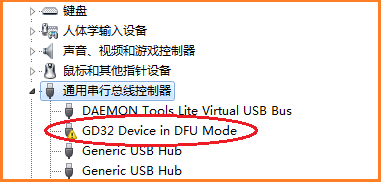
In IAR:



## User APP upgrade

To supply power for GD32F450I-EVAL board, the USB\_FS interface on board is connected to the computer through the USB cable. Because GD EVAL board has loaded the IAP application program, and there is no effective user application program, so in the IAP application program, it directly enters the USB DFU mode to provide the computer to identify the USB DFU device. Normal recognition is shown in the lower left.

The situation in the lower right is due to the incorrect loading of Driver. You can choose to update the driver software by right-clicking "DFU in FS Mode" browse the computer to find the driver software (the driver file is located in the GD32 MCU Dfu Tool\_v3.7.1.3145 installation directory GD32 MCU Dfu Tool\GD32 MCU Dfu Drivers\_v1.0.1.2316\x64 or x86).



Open DFU tool, user can upgrade the firmware. Please refer to “GigaDevice Dfu Tool User Manual”.

# Custom HID for IAP implementation steps

## HID IAP tool installation

User don’t need install the IAP demo tool GD32\_MCU\_HID\_IAP\_Programmer and can open it directly. The cutom HID IAP don’t need driver, because it uses the driver offered by windows OS.

## IAP driven firmware generation and loading

Using the USB IAP routine to generate IAP driver firmware (folder path: GD32\_USB\_IAP\Project\USBFS\_Device\_HID)

Use burning tools to burn to the development board. After this routine is restarted, if the UserKey button is not pressed on the board or there is no valid user application, it will enter the USB IAP mode and wait for upgrade; otherwise, it will enter the user application.

The start address corresponding to the user firmware is defined at 0x08004000. Users can also customize the user code addresses.

The custom HID IAP also need some notes, these notes are same as the DFU IAP, please refer to section 5.2.

## User APP generation

Please refer to section 5.3.

## User APP upgrade

Open custom HID IAP tool, user can upgrade the firmware. Please refer to “Gigadevice HID IAP Programmer User Manual”.

# MSC (USB HOST) for IAP implementation steps

## IAP driven firmware generation and loading

Using the USB MSC Host routine to generate IAP driver firmware (folder path: GD32\_USB\_IAP\Project\USBFS\_Host\_MSC)

Use burning tools to burn to the development board. After this routine is restarted, if the UserKey button is not pressed on the board or there is no valid user application, it will enter the USB IAP mode and wait for upgrade; otherwise, it will enter the user application.

The start address corresponding to the user firmware is defined at 0x08004000. Users can also customize the user code addresses.

The MSC IAP routine is implemented on USB host, so it is not necessary to download the program through the computer. The IAP driver identifies the ‘bin’ file in the accessed mass storage device and downloads the corresponding file to the corresponding address on the chip.

## User APP generation

Please refer to section 5.3.

## User APP upgrade

Because this routine applied on USB host and uses MSC class to implement IAP, it needs to be implemented with U disk.

1. First, the generated user program bin file is copied to the U disk.

2. Then connect the U disk to the development board. The Development Board recognizes the corresponding bin file and displays it on the LCD screen.

3. Users can select the bins that need to be downloaded by using the Wakeup key and UserKey, then download the selected bin file by using the Tamper button.

# MSC (USB Device) for IAP implementation steps

## IAP driven firmware generation and loading

Using the USB MSC Device routine to generate IAP driver firmware (folder path: GD32\_USB\_IAP\Project\USBFS\_Device\_MSC)

Use burning tools to burn to the development board. After this routine is restarted, if the UserKey button is not pressed on the board or there is no valid user application, it will enter the USB IAP mode. At this time, the PC will recognize a new removable disk. The user will need to copy the upgraded BIN file into the U-disk and change the file name to “update.bin”. Otherwise, it will enter the user application.

The start address corresponding to the user firmware is defined at 0x08020000. Users can also customize the user code addresses. It should be noted that the IAP driver uses the file system to read the file of update.bin from the MSC storage medium and write it to the user program area, so the file name of BIN which copied into the U-disk must be changed to “update.bin” to be recognized and upgraded by the IAP.

## User APP generation

Please refer to section 5.3.

## User APP upgrade

This routine uses MSC device class to implement IAP function. It needs to follow the following steps:

1. First, the program will enter IAP mode when startup.

2. Then, by connecting the development board to the PC, a new removable storage device (U-disk) is identified.

3. Next, copy the file to be upgraded, named update.bin, into the U-disk.

4. Finally, press the reset button or reboot the development board by pressing the UserKey, the program will determine whether there is an update.bin file. If there is exists, it will be written to the user program area. After the successful upgrade, LED1 or LED2 will flicker three times, and finally enter the user program area.

# Revision history

**Table 9‑1. Revision history**

|  |  |  |
| --- | --- | --- |
| **Revision No.** | **Description** | **Date** |
| 1.0 | Initial Release | Jan, 2019 |
|  |  |  |
|  |  |  |