

# Application Note

**Document No.: AN1080**

**APM32 Series Tool Chain User Manual**

**Version: V1.1**

# 1 Introduction

For an MCU, the quality of the tool chain will become one of the keys to its market performance. When tool chain is used to develop programs, the functional completeness and interface friendliness of the tool chain will directly affect the user experience. Moreover, the performance of the chip also depends on the performance of the compiler to a large extent.

In order to enable users to use MCU to complete the design and verification of application system solutions, MCU manufacturers generally provide support in two ways: one is that MCU manufacturers provide a full set of platform tools through their own research and development, and the other is that customers seek the support of third-party general tools (e.g. IAR).

According to our own product characteristics and application field requirements, Geehy Semiconductor independently develops and builds a complete set of software and hardware development tools, which can help users shorten the product development cycle, improve product stability, and provide strong support for customers in the process of MCU application development.

This application note provides basic instructions for use of Geehy software and hardware tools, including the guidance for the connection and configuration operation process of Geehy-Link, APM32PROG, APM32 ISP Multiport Programmer and DFU Programmer.

## Contents

<b>1</b>	<b>Introduction</b> .....	<b>1</b>
<b>2</b>	<b>Geehy tool chain tutorial</b> .....	<b>3</b>
<b>3</b>	<b>Geehy-Link tutorial</b> .....	<b>4</b>
3.1	Configure Geehy-Link debugger under Keil MDK .....	5
<b>4</b>	<b>APM32PROG tutorial</b> .....	<b>9</b>
4.1	APM32PROG software upper computer .....	9
4.2	APM32PROG offline programming.....	10
4.3	APM32PROG online reading.....	11
<b>5</b>	<b>APM32 ISP Multiport Programmer &amp; DFUProgrammer</b> .....	<b>14</b>
5.1	About BootLoader of APM32 MCU .....	14
5.2	APM32 ISP Multiport Programmer .....	14
5.3	DFUProgrammer .....	18
<b>6</b>	<b>Version history</b> .....	<b>22</b>

## 2 Geehy tool chain tutorial

> Preparatory work

> Hardware preparation

> Geehy-Link

> APM32 PROG

> USB-to-TTL module

> APM32F0/1/4 series MCU development board

> Software preparation

> Keil uVision5 (recommend version 5.25 and above)

> IAR EW for Arm 9.20.2

> APM32 PROG

> ISP Multiport Programmer

> DFUProgrammer

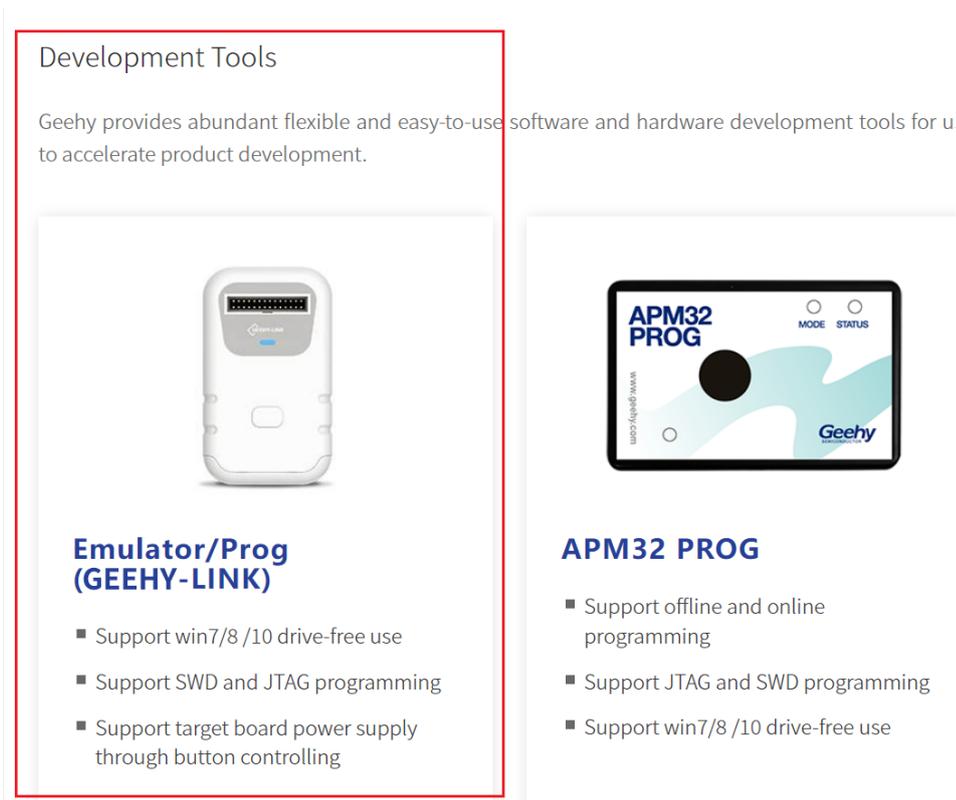
> APM32F0/1/4 series MCU SDK

> The above Geehy related software can be downloaded from Geehy data ([APM32 MCU (geehy.com)] (<https://geehy.com/support/apm32>))

### 3 Geehy-Link tutorial

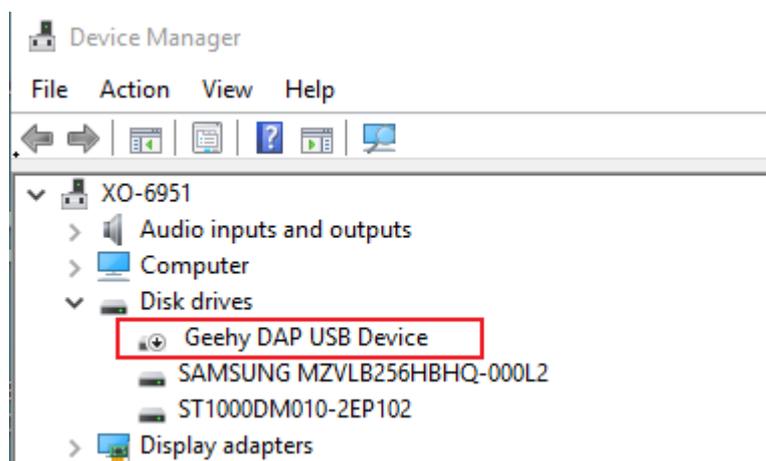
Geehy-LINK is a development tool integrating simulator and programmer, which can debug and simulate the APM32 full-range MCU products online in Keil, IAR and other integrated development environments. Support full-speed operation, single-step debugging, breakpoint setting and other debugging methods.

Figure 1 Geehy-Link Link on the Official Website



Use the USB cable to connect Geehy-LINK to the computer. After they are connected successfully, Geehy DAP USB Device will appear on the device manager, as shown in the following figure

Figure 2 Geehy DAP USB Device in Device Manager



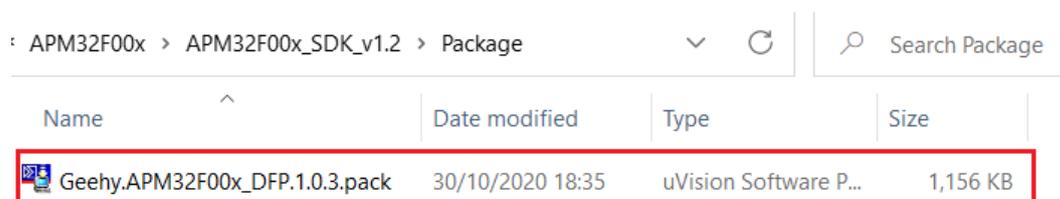
### 3.1 Configure Geehy-Link debugger under Keil MDK

#### 3.1.1 Pack supports installation

In the target environment, you need to prepare Keil uVisin5 (recommend version 5.25 and above) environment in advance, and install the Pack of APM32F0/1/4 series MCU that needs to be debugged. The installation method (taking APM32F003F6P6 chip as an example) is as follows:

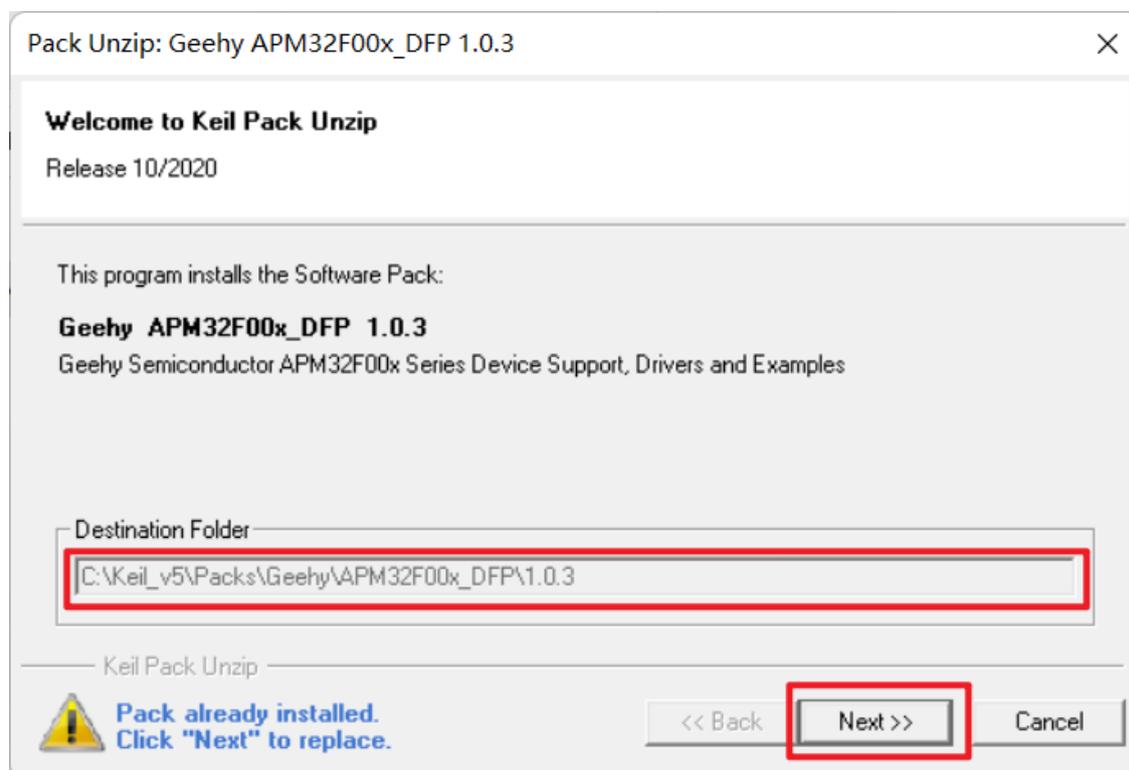
Open the downloaded pack under "APM32F00x\_SDK\_v1.2\Package".

Figure 3 Select the pack File in the SDK File



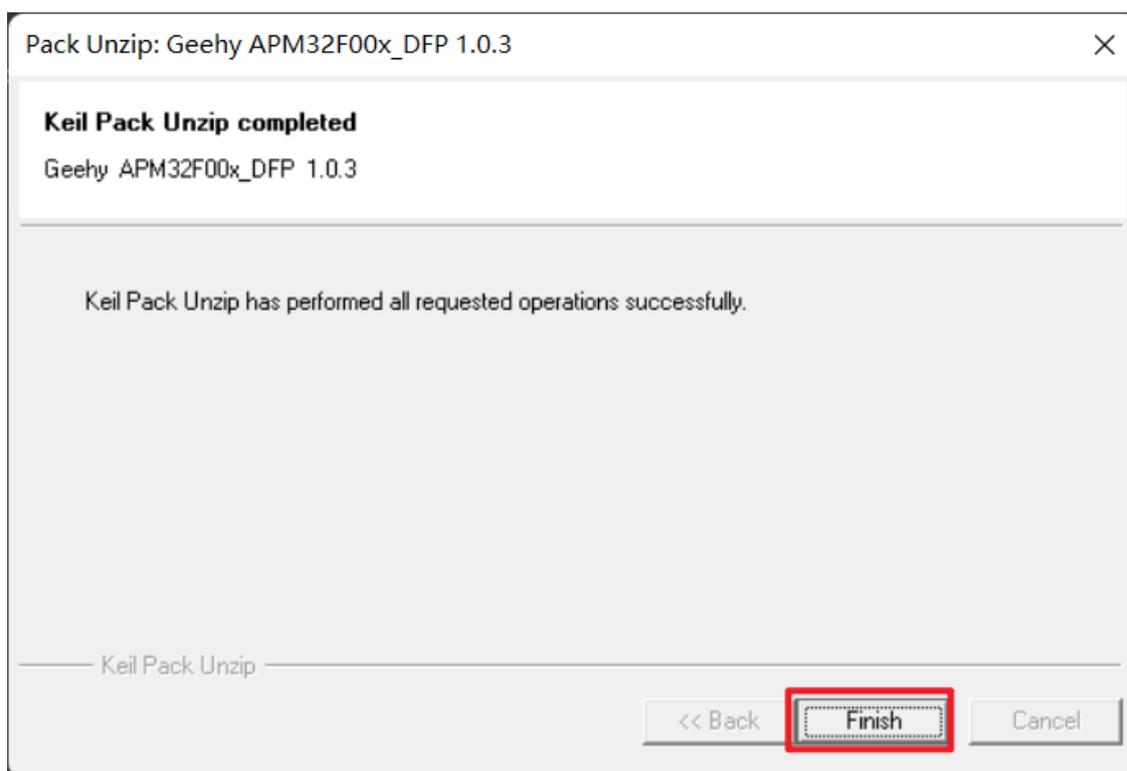
Install the Pack (double-click the left mouse button) to Pack installation directory of Keil (generally the default path is ok).

Figure 4 Example of Installation Directory



Wait until the installation is finished.

Figure 5 Example of Installation Finished

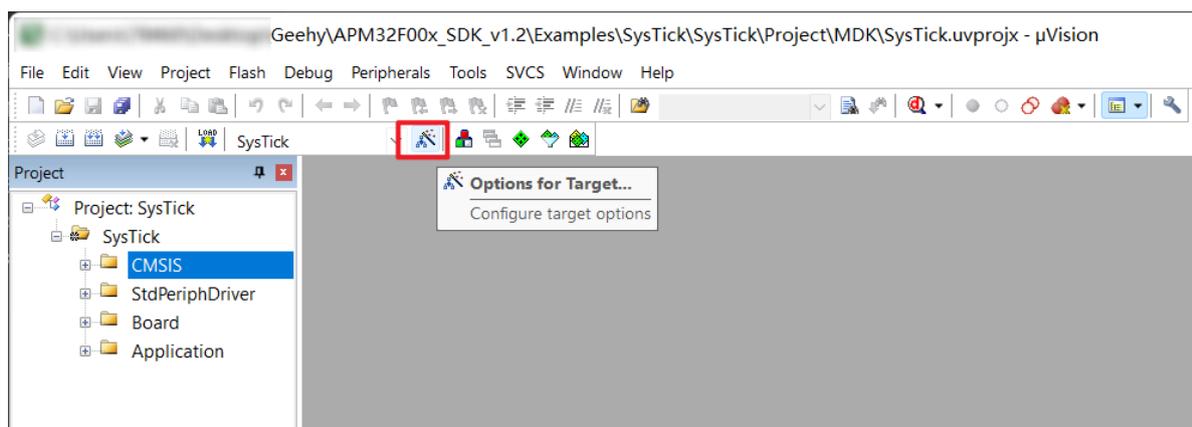


### 3.1.2 Configuring the Geehy-Link debugger

Open the project file

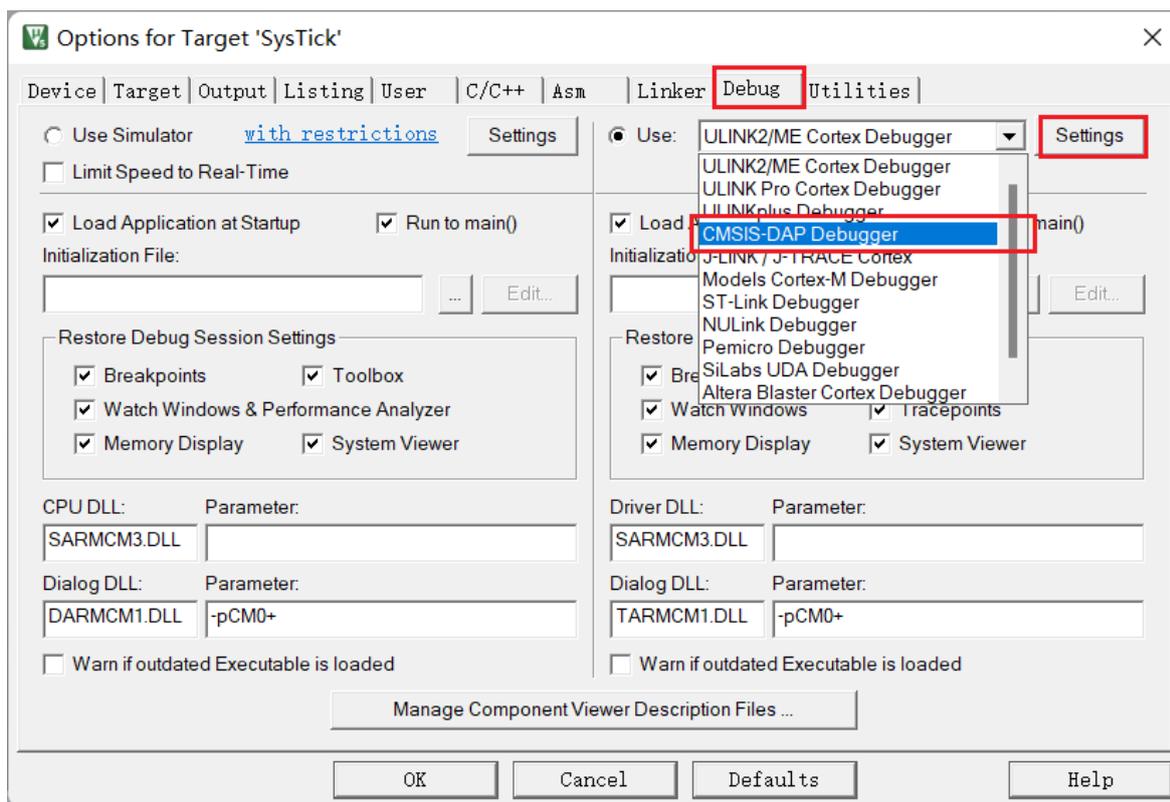
"APM32F00x\_SDK\_v1.2\Examples\SysTick\SysTick\Project\MDK\SysTick.uvprojx" and then select the "Options for Target" tab.

Figure 6 Select the "Options for target..." Tab



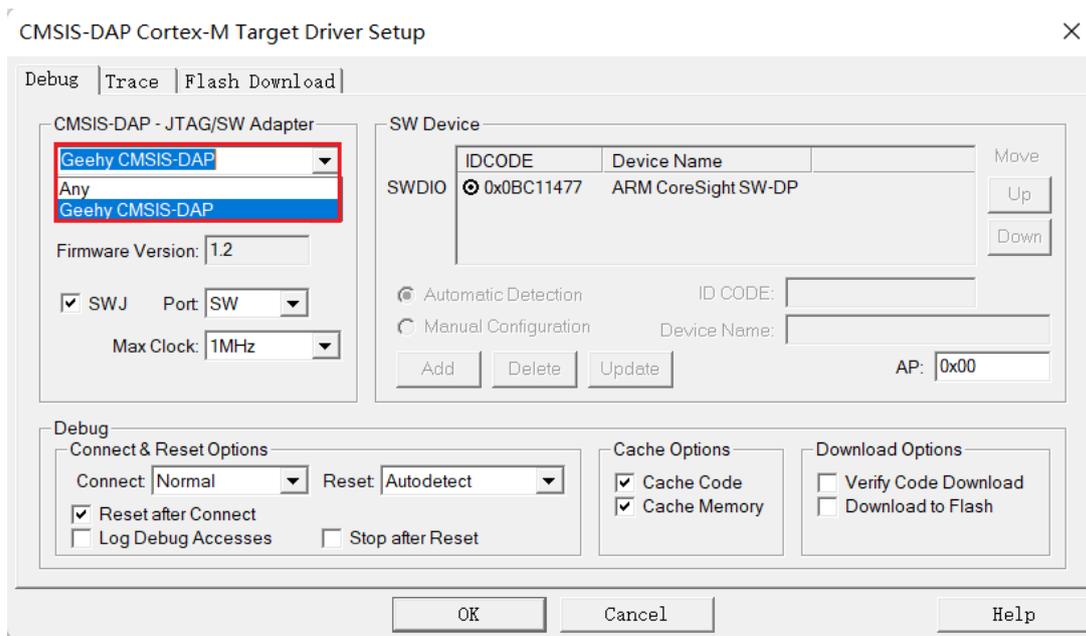
In the "Debug" option, download and select the simulator as "CMSIS-DAP Debugger" and then select "Setting".

Figure 7 Select "CMSIS-DAP Debugger" and then Select "Settings"



In the "CMSIS-DAP Cortex-M Driver Setup" interface, select "Geehy CMSIS-DAP" (the old version is "Apex CMSIS-DAP") and click "OK". The emulator configuration ends here.

Figure 8 Select "Geehy CMSIS-DAP" and Click "OK"

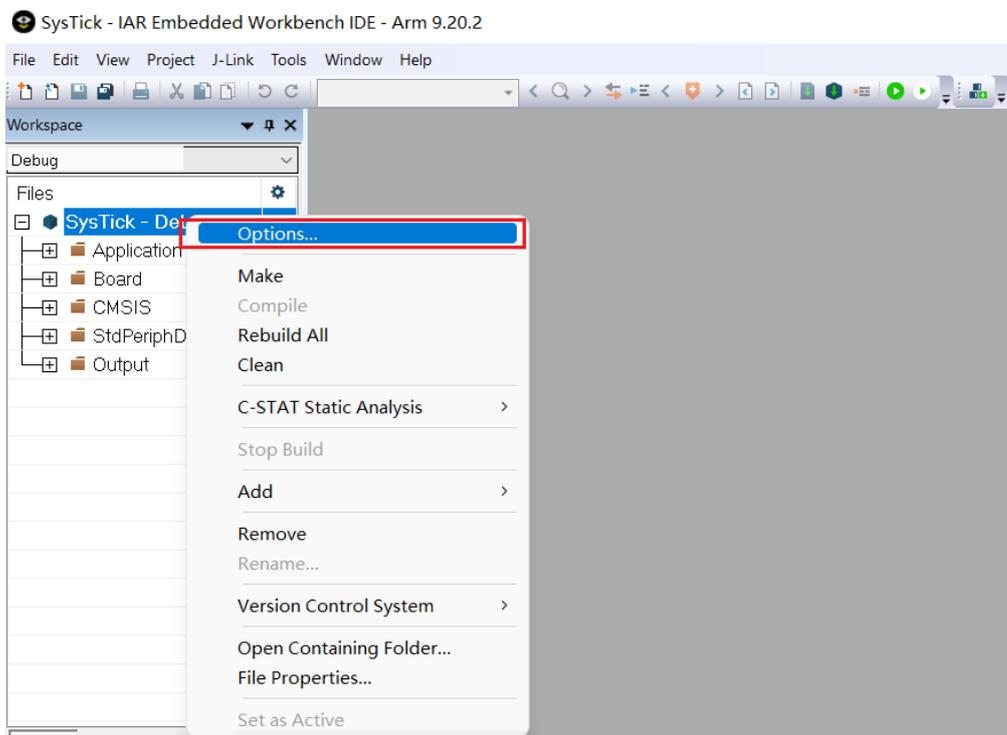


### 3.1.3 Configuring Geehy-Link debugger under IAR EW for Arm

Open the project file "APM32F00x\_SDK\_v1.2\Examples\SysTick\SysTick\Project\IAR\SysTick.eww"

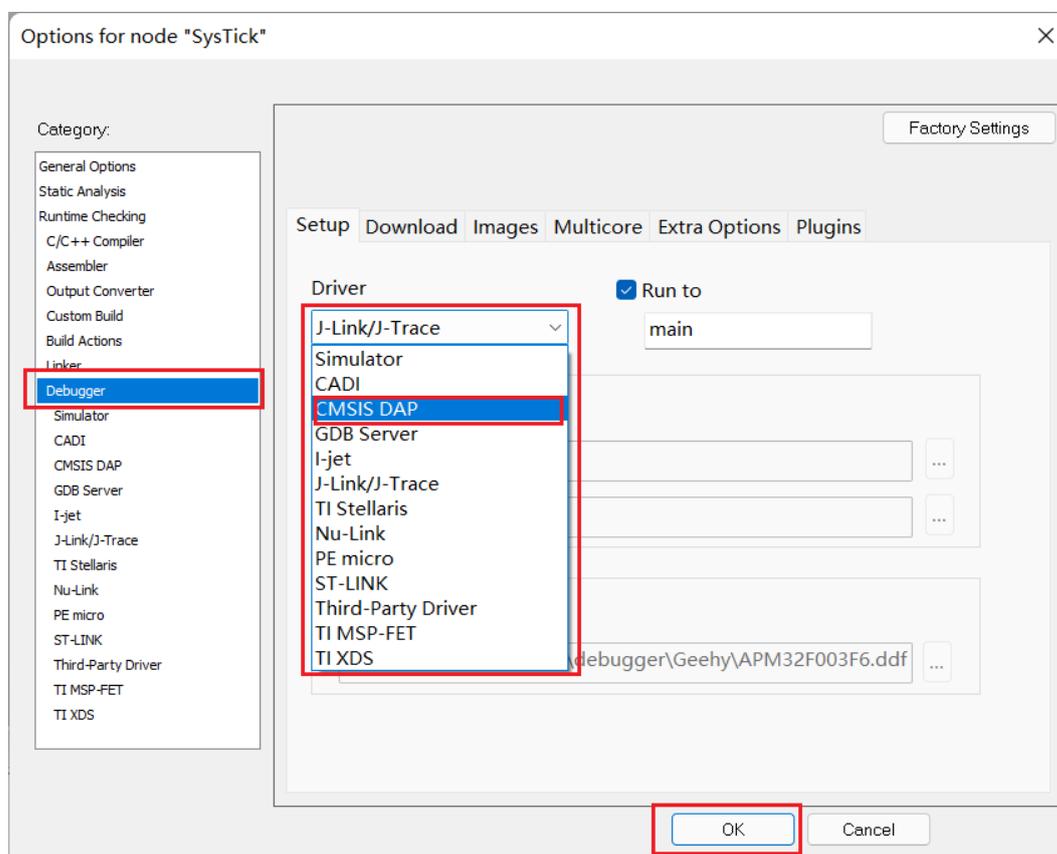
and then select the "Options" tab.

Figure 9 Open the Project File and Select "Options"



Select "CMSIS DAP" simulator in "Debugger" and click "OK".

Figure 10 Select "CMSIS-DAP" and Click "OK"

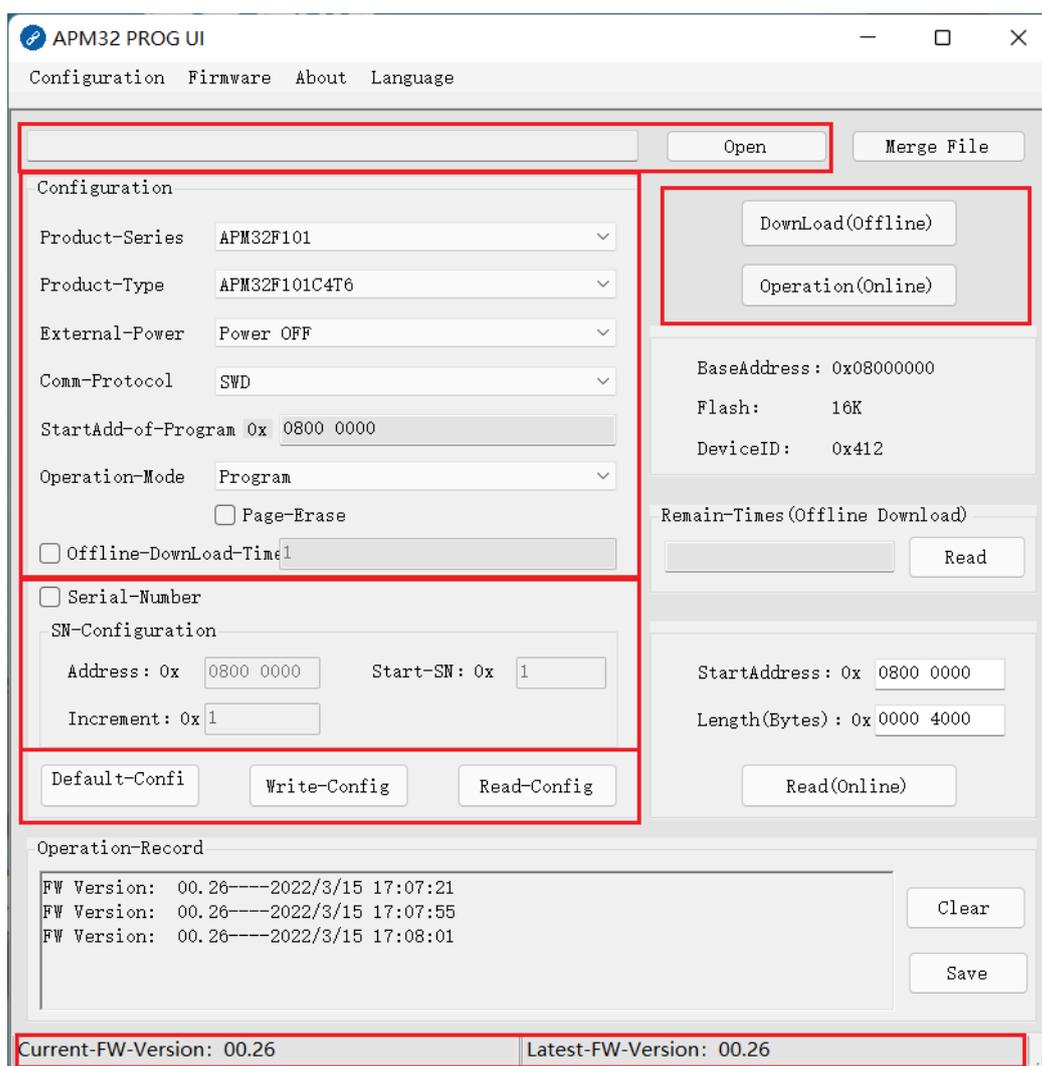


## 4 APM32PROG tutorial

### 4.1 APM32PROG software upper computer

Download the upper computer software Apex-APM32-Setup.msi, and then install it (the installation interface is omitted). Open the software and connect APM32PROG. After successful connection, the upper computer will read the firmware version of the current APM32PROG, as shown in the red box at the bottom of the figure below. If the firmware version of the current APM32PROG is lower than the latest version, you will be prompted to update the firmware version.

Figure 11 APM32 PROG Functional Area

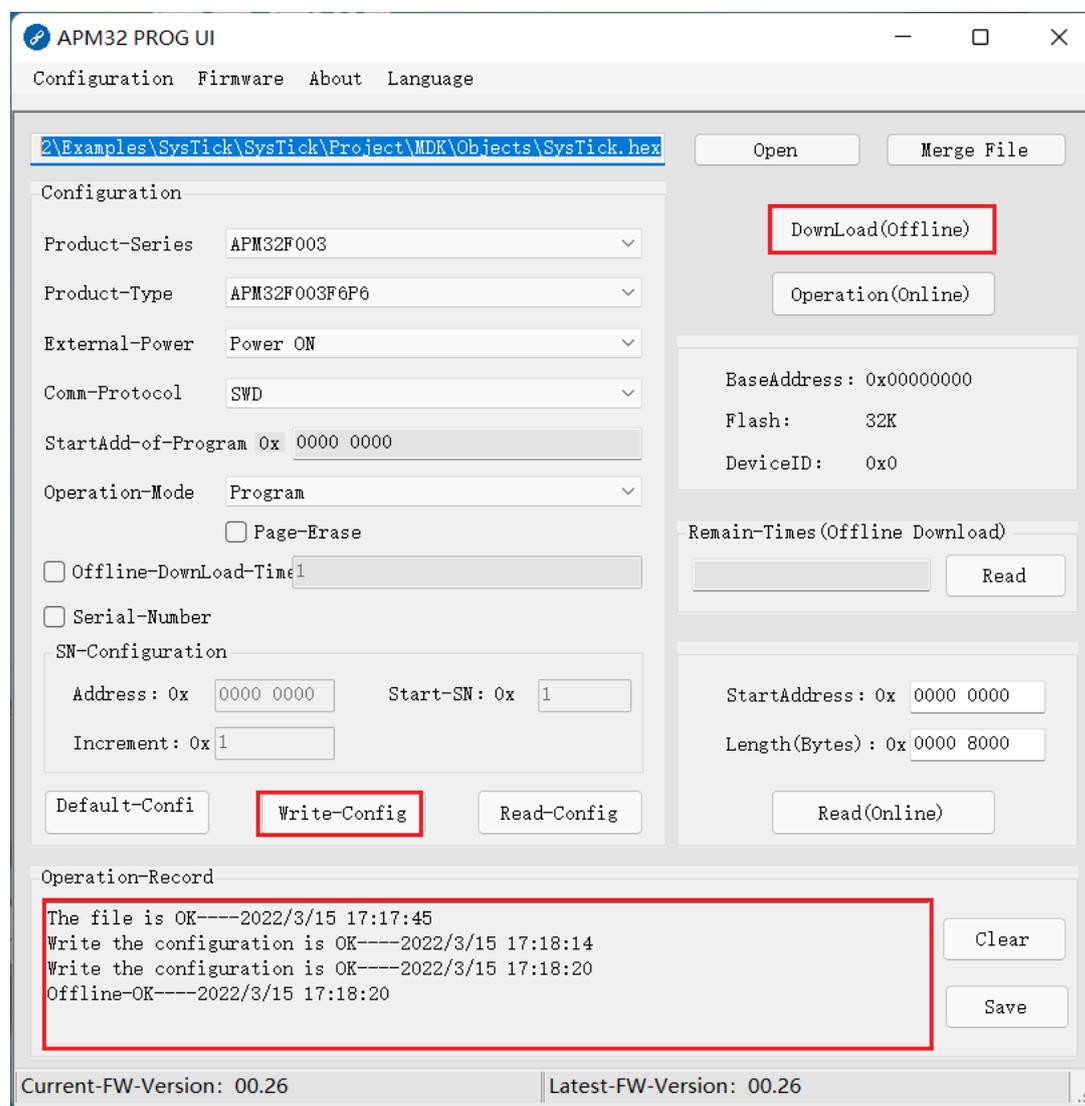


The functional area of the APM32PROG upper computer is divided into the file opening area, configuration area (including serial number configuration and configuration writing), and offline or online operation area. APM32PROG, together with the upper computer, can realize many functions such as programming, reading, encryption and decryption of the target MCU. The following takes APM32F003F6P6 as an example to introduce the methods of offline downloading and online reading MCU of APM32PROG. Readers of other functions can explore according to the software help and the instructions for use.

## 4.2 APM32PROG offline programming

Our chip is APM32F003F6P6, which only supports SWD debugging. Our APM32PROG configuration is shown in the figure below.

Figure 12 APM32F003F6P6 APM32 PROG Configuration

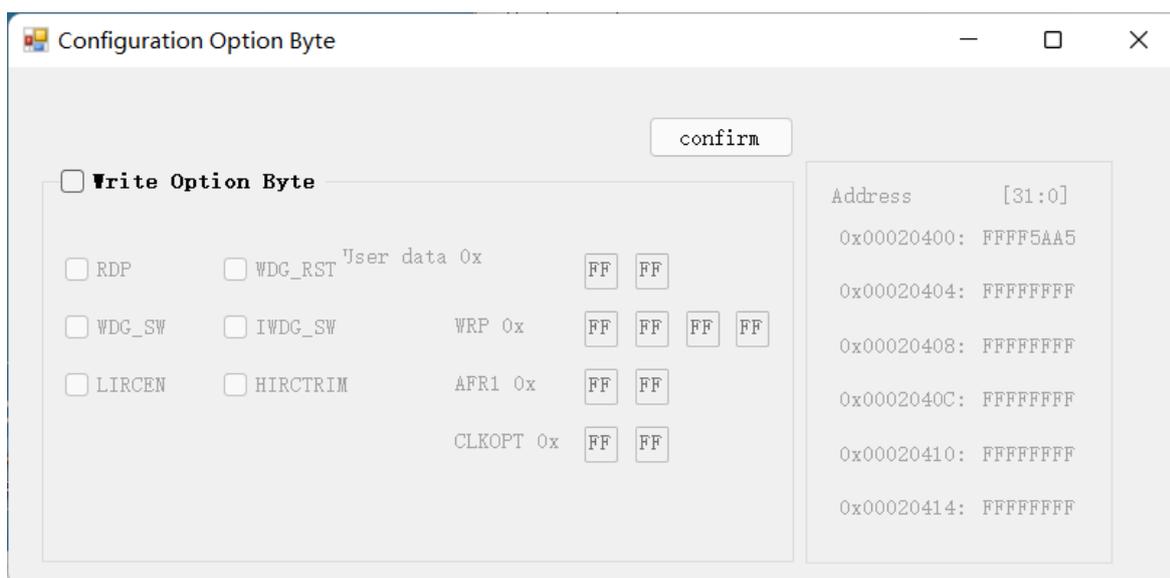


Select the hex (or bin) file to be downloaded,

1. Product-Series APM32F003
2. Product-Type APM32F003F6P6

At this time, the APM32PROG upper computer will pop up the window of configuring the option bytes. Readers who need to configure MCU option bytes can use this function. This function is not used for this offline programming, so it is omitted here.

Figure 13 Window of Configuring Option Byte



3. Select Power ON for External-Power
4. Select SWD for Comm-Protocol
5. Set BaseAddress to 0x00000000
6. Select Program for Operation-Mode

Serial-Number is not required and is not configured temporarily. After the above configuration is set, click "Write-Config" to write the configuration to APM32PROG. Then click "Download (Offline)" to download the program to APM32PROG. At this time, the configuration information and program have been downloaded to APM32PROG. Next, you only need to supply power to APM32PROG, connect the target chip, then click the programming start button, wait until the programming prompt light turns green and the offline programming work is completed.

### 4.3 APM32PROG online reading

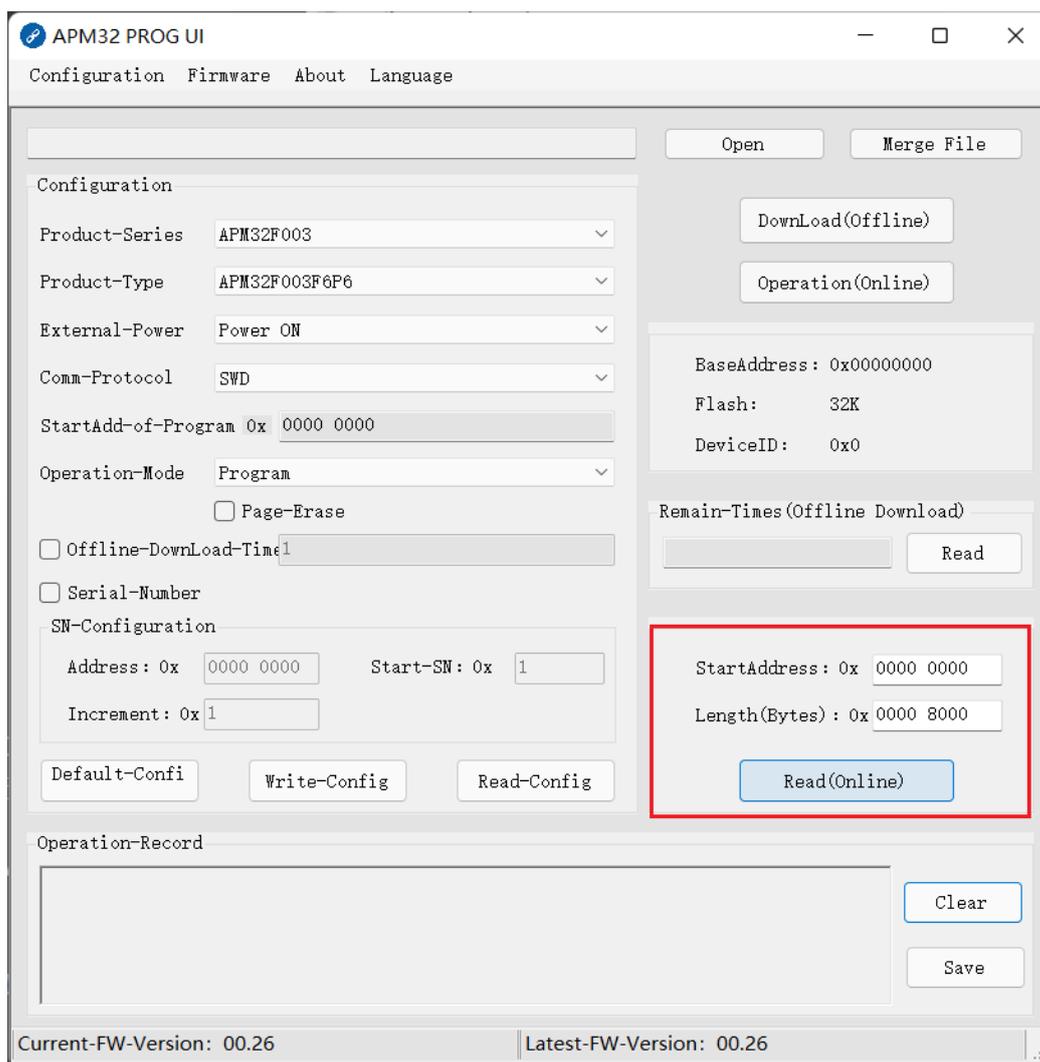
APM32PROG can not only complete the offline operation of the target MCU, but also perform online reading or other operations on the MCU. This section will take APM32F003F6PU as an example to read the program burned in the previous section.

The parameter configuration is summarized in the previous 2.2 in the read setting interface

1. Set StartAddress to 0x00000000
2. Set Length(Bytes) to 0x00008000

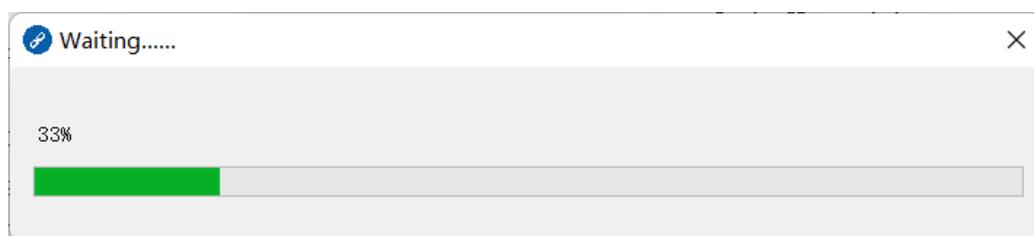
The last selected operation is "Read (Online)" (APM32PROG needs to connect the target MCU and PC) to read the content that the start address of the target MCU is 0x00000000 and the size is 0x8000 Bytes.

Figure 14 Read the Burnt Program



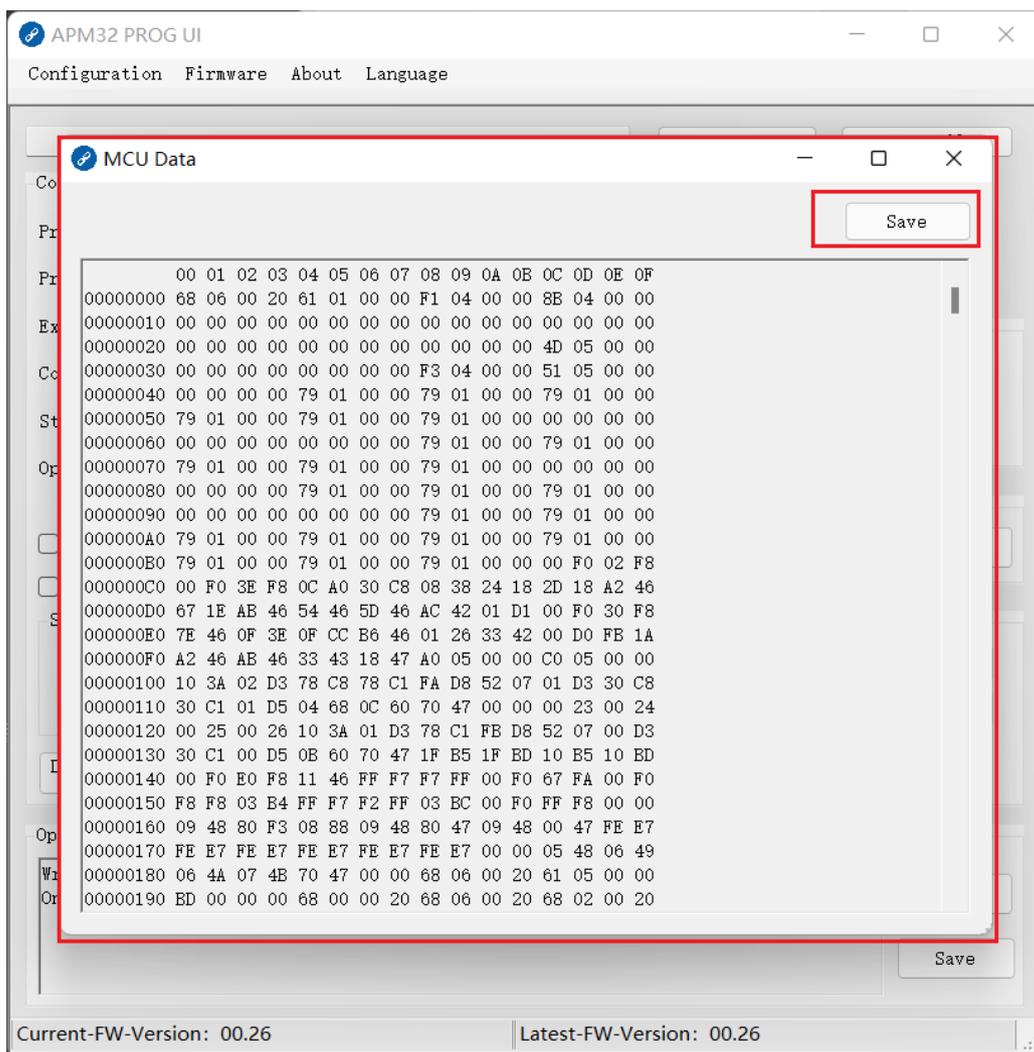
Wait to read the progress bar.

Figure 15 Read Progress Bar



After reading successfully, the data pane interface will pop up, and you can click the "Save" button to save the read data.

Figure 16 Data Pane Interface



## 5 APM32 ISP Multiport Programmer & DFUProgrammer

### 5.1 About BootLoader of APM32 MCU

A segment of BootLoader program and system storage area are built in APM32F0/1/4 and APM32E1 series MCU before they leave the factory, to facilitate users to implement program downloading, option bytes programming and other operations on MCU through ISP. The BootLoader of each chip has different communication methods to support upgrade.

Table 1 Communication Mode of Each Chip

General MCU series	Whether supporting ISP upgrade	Supporting interfaces	BootLoader entry mode
APM32F003x4x6	No	-	-
APM32F030x8/051x8	Yes	USART (PA9/PA10、PA14/PA15)	Boot0 connects to 1
APM32F030xC/091xC	Yes	USART (PA9/PA10、PA14/PA15) I2C	Boot0 connects to 1
APM32F072xB	Yes	USB (PA11/PA12) USART (PA9/PA10、PA14/PA15) I2C (PB6/PB7)	Boot0 connects to 1
APM32F103xBxCxE	Yes	USART (PA9/PA10)	Boot0 connects to 1, and Boot1 connects to 0
APM32F405xG 407xExG APM32F415xG 417xExG	Yes	USB (PA11/PA12) USART (PA9/PA10、PB11/PB12、PC11/PC12) CAN (PB5/PB13)	Boot0 connects to 1, and Boot1 connects to 0
APM32E103xE	Yes	USART (PA9/PA10)	Boot0 connects to 1, and Boot1 connects to 0

This brief summary will introduce the basic chip connection and programming methods for the PC-side tools of commonly used USART and USB. More information can be obtained from the official website or technical support engineers of Geehy .

### 5.2 APM32 ISP Multiport Programmer

This brief summary will take APM32F103VBT6 chip as an example to introduce the basic function of

APM32 ISP Multiport Programmer -- chip programming and erasing. The software can be obtained from the official website of Geehy.

### 5.2.1 Preparation before connecting MCU

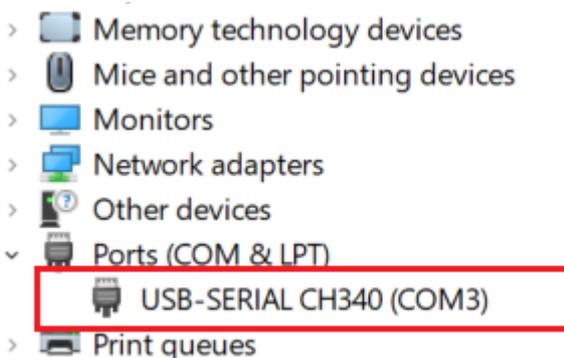
Since the communication mode between APM32 ISP Multiport Programmer and MCU is USART, here we need to use USB-to-TTL module to connect PC to MCU. The wiring method is as follows.

Table 2 Wiring Method

PC	USB to TTL		APM32F103VBT6 MINIBOARD
USB	USB	TX	PA10
		RX	PA9
		GND	GND
		VCC	VCC

After wiring, install the corresponding USB-to-TTL module driver. After the driver is installed normally, you will see the corresponding "USB Serial Port" port in the "Device Manager".

Figure17 "USB Serial Port" Port in "Device Manager"



### 5.2.2 APM32 ISP Multiport Programmer connects to MCU

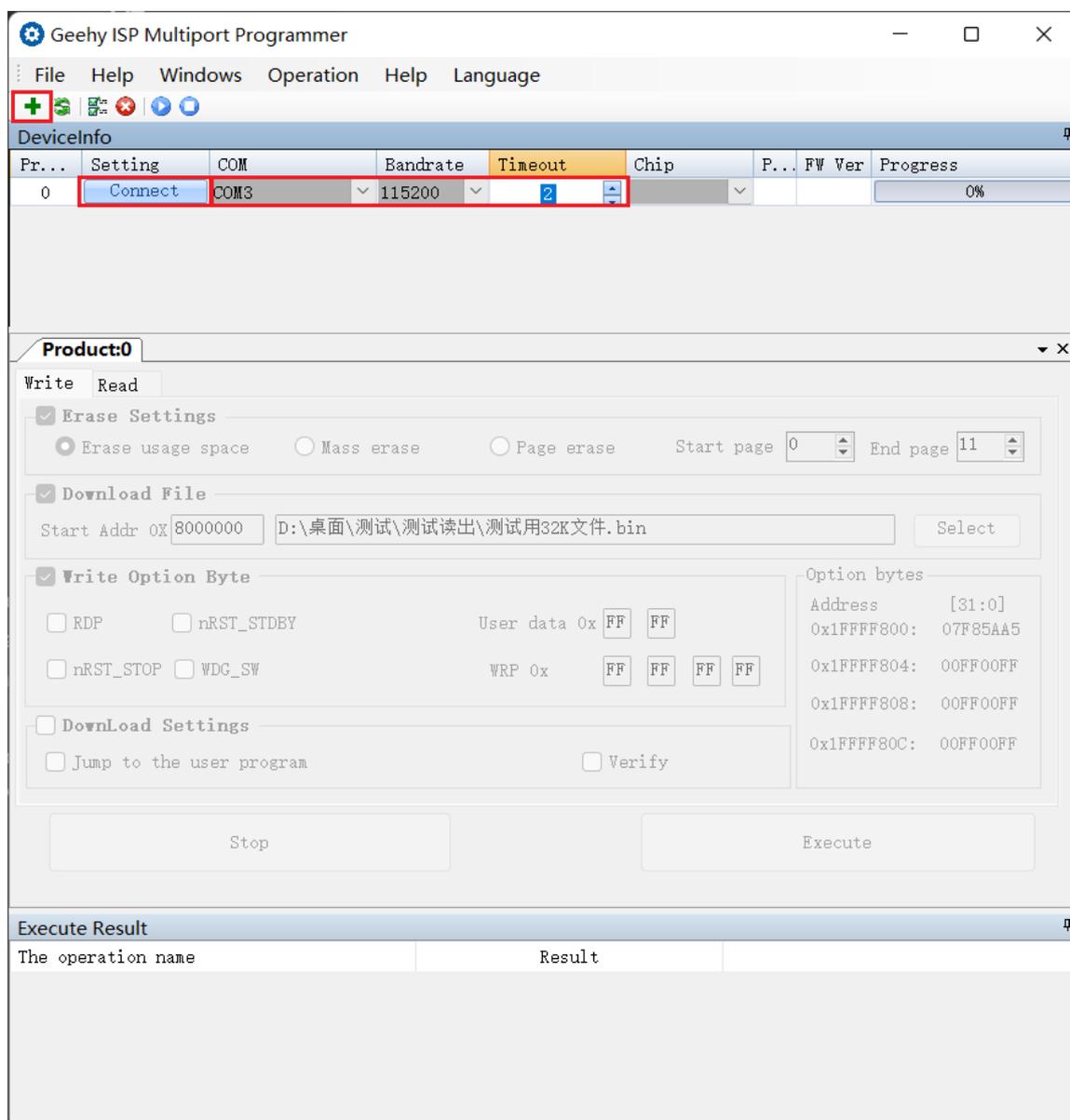
As shown in the table in Section 5.1, "Boot0 connects to 1 and Boot1 connects to 0" is required so that APM32F103VBT6 can enter BootLoader (MCU needs to be reset when MCU is powered on before the jumper cap is reconnected). After all wiring operations are completed, open the installed APM32 ISP Multiport Programmer (the installation process is omitted).

The general process of connecting APM32 ISP Multiport Programmer to MCU is as follows,

1. Click the plus sign Add New Window below the main interface to get the setting window
2. Select the corresponding system USB Serial Port under the COM in the setting window. Here is COM3

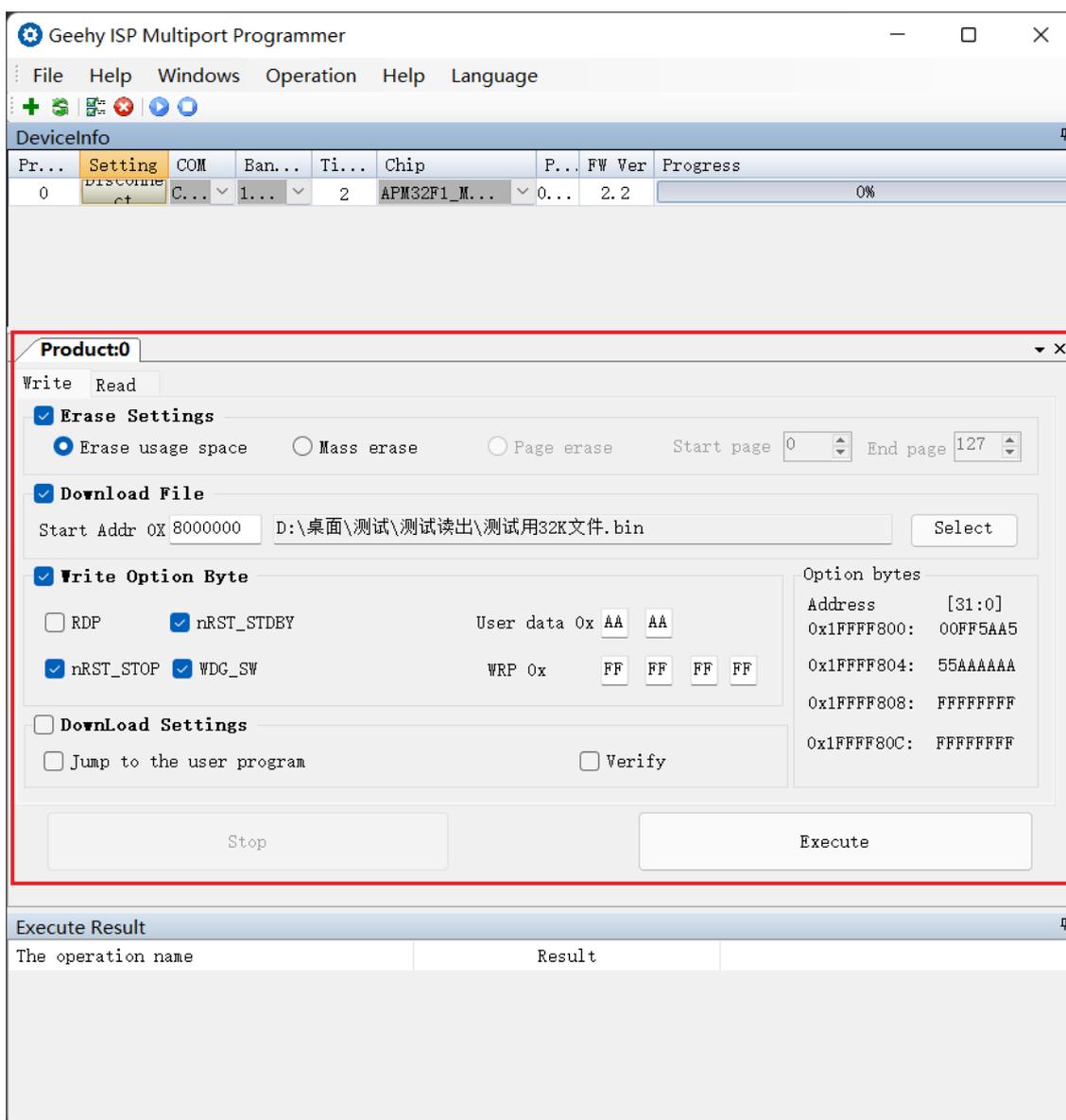
3. Set the required baud rate under the Bandrate in the setting window. Here it is set to 115200
4. Set the connection timeout under the Timeout in the setting window. It is set to 2 here
5. Finally, click the Connect button under Setting to complete the connection to the target MCU

Figure 18 Connect MCU



After successful connection, the information of the currently connected chip and some operable options will be displayed in the same interface.

Figure 19 Information and Operable Options of the Currently Connected Chip

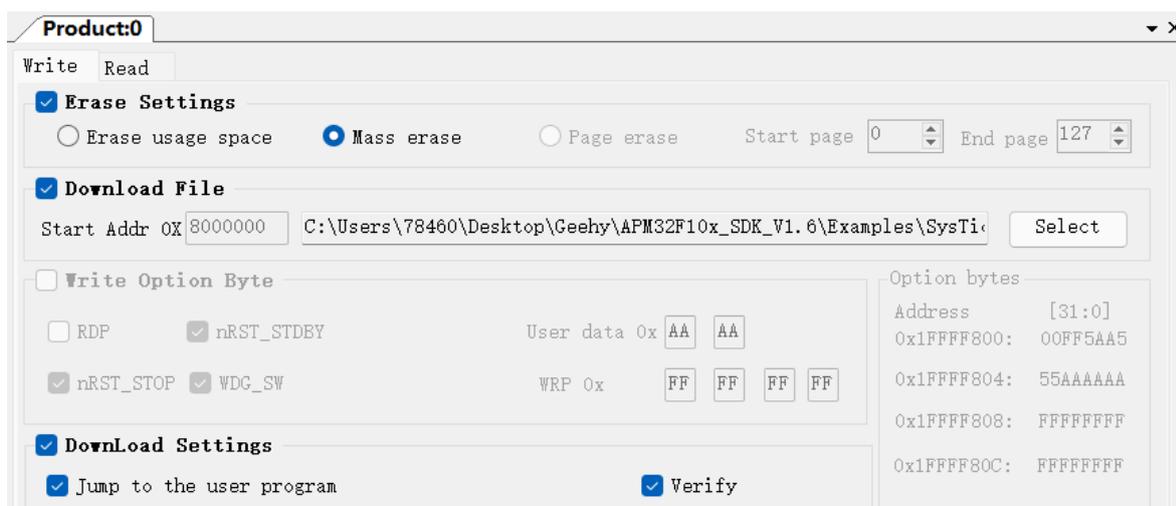


### 5.2.3 APM32 ISP Multiport Programmer download program

In Section 5.2.2, after successfully connecting APM32F103VBT6 chip, we can see that there are many operable items, such as: under Erase Setting, you can choose full erase or sector erase. Select the downloaded file In Download File; choose to operate the option bytes in Write Option Byte; in Download Setting, some operations after downloading can be selected.

Here we choose to download the program of APM32F103VBT6, adopt full erase, and do not operate the option byte, and after downloading, verify and execute the program. Relevant setting interface is shown below.

Figure 20 Setting Page



After the above options are checked, click "Execute" to execute our settings. Wait and when the progress bar is finished, we can view our operation log in the "Execute Result" window.

Figure 21 "Execute Result" Window

Execute Result			
The operation name			Result
2022/3/16 15:52:36	Product:0	Mass erase	Success
2022/3/16 15:52:37	Product:0	Download File	Success
2022/3/16 15:52:37	Product:0	Verify	Success
2022/3/16 15:52:37	Product:0	Jump to the user program	Success

### 5.3 DFUProgrammer

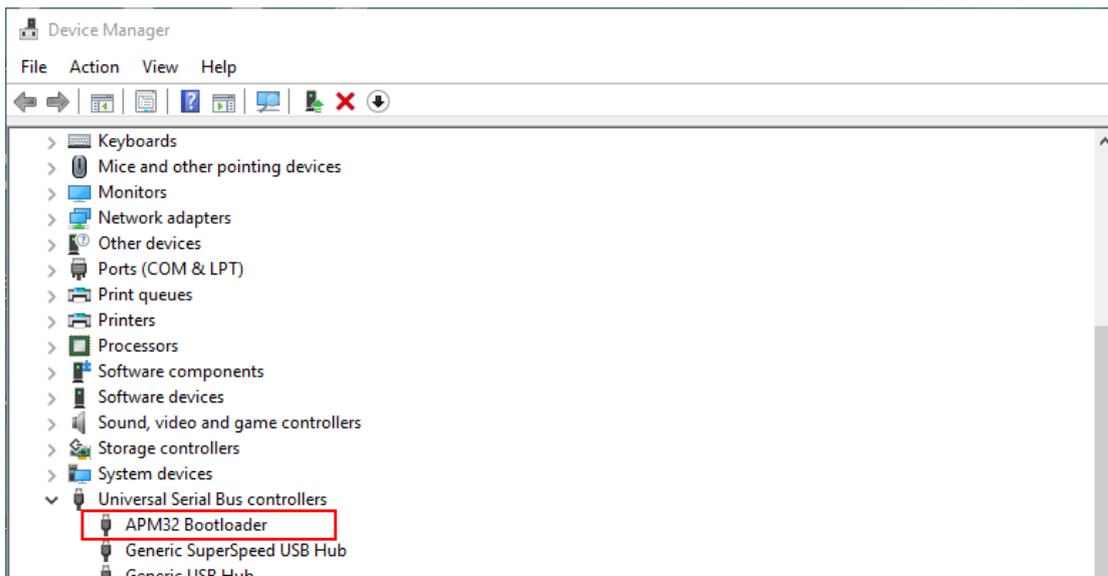
The function of this software is basically similar to that of APM32 ISP Multiport Programmer, except that it supports using the USB as the software to upgrade MCU of peripherals. This brief summary will use APM32F407IGMINIBOARD to demonstrate the chip program reading function of DFUProgrammer.

#### 5.3.1 Preparation before connecting MCU

The installation package can be obtained on the official website of Geehy. It includes the installation instructions of the software.

After the driver is correctly installed, connect APM32F407IGMINIBOARD "Boot0 connects to 1 and Boot1 connects to 0" (MCU needs to be reset when MCU is powered on before the jumper cap is reconnected). Use USB cable to connect PC and development board. At this time, we will see the "APM32 Bootloader" device under "Universal serial bus controller" on the device manager.

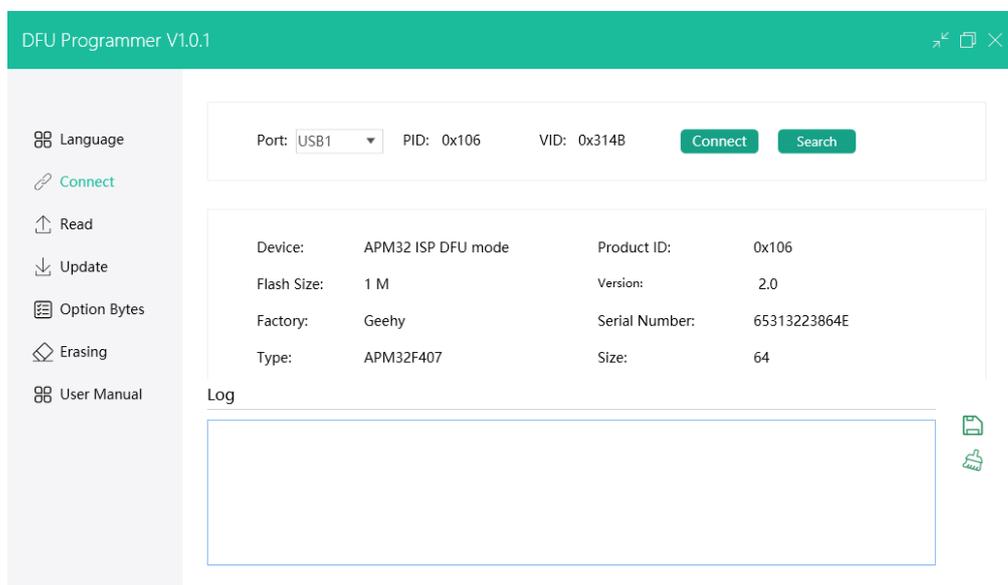
Figure 22 "APM32 Bootloader" in the "Device Manager" Interface



### 5.3.2 DFUProgrammer connects to MCU

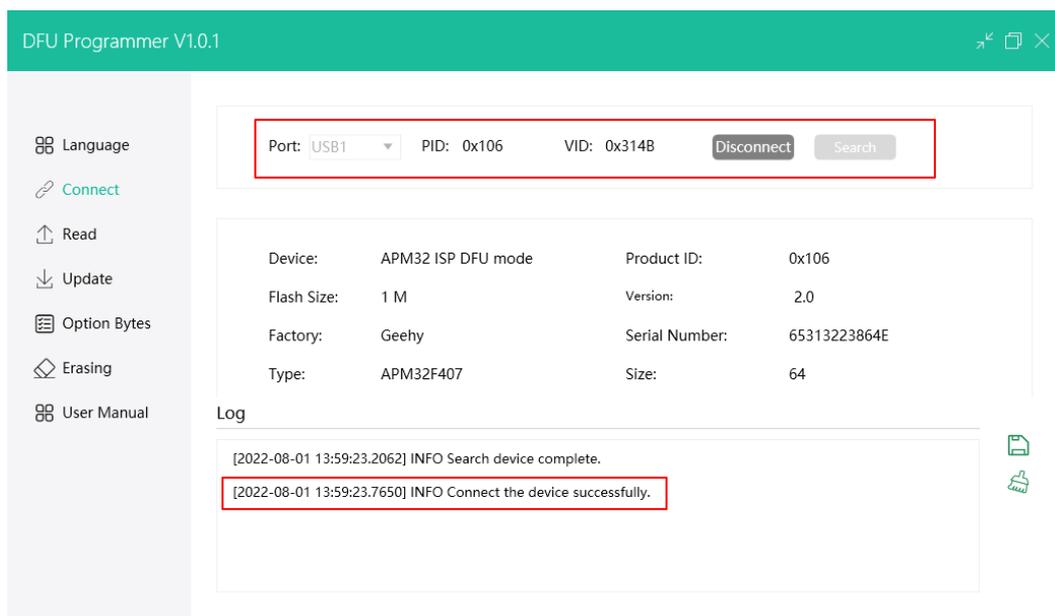
Open DFUProgrammer software. If the computer has been connected to an "APM32 Bootloader" device, the initial interface is shown below. If no device is found, click the "Search device" button to search for the device. Click the "Connect a device" button to connect to the target MCU.

Figure 23 Main Interface of DFUProgrammer Software



After the target MCU is successfully connected, we can see the corresponding prompt in the log interface.

Figure 24 Status of Device Successfully Connected



### 5.3.3 DFU Programmer read program

Click the "Read a chip" button on the left to enter the chip content reading interface. In this interface, you can set the content and size of read chip, click the "Read chip" button to read the corresponding data, and click the "Save data" button to save the data to PC.

Figure25 Read a Chip

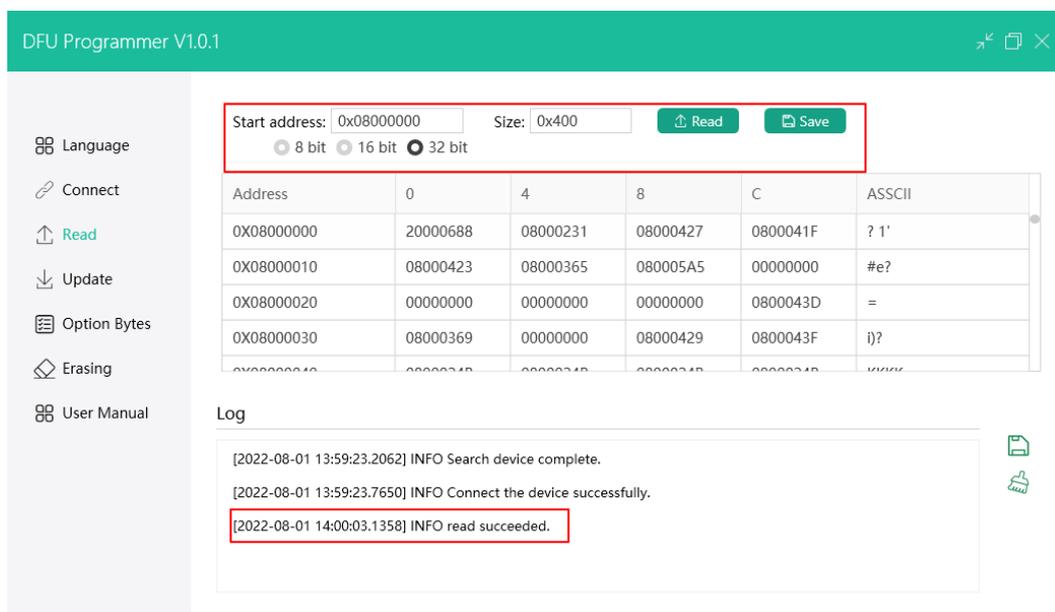
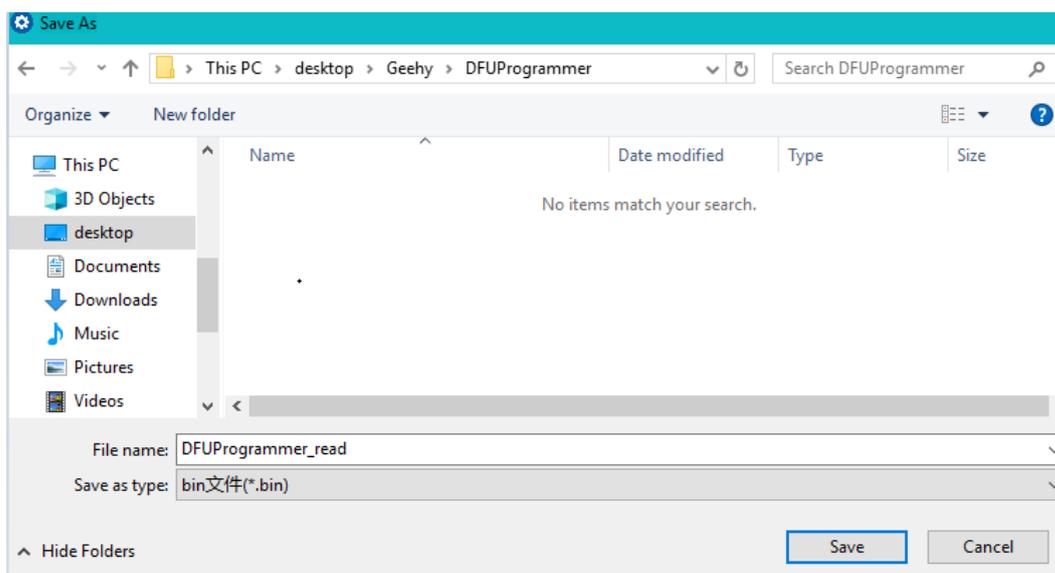
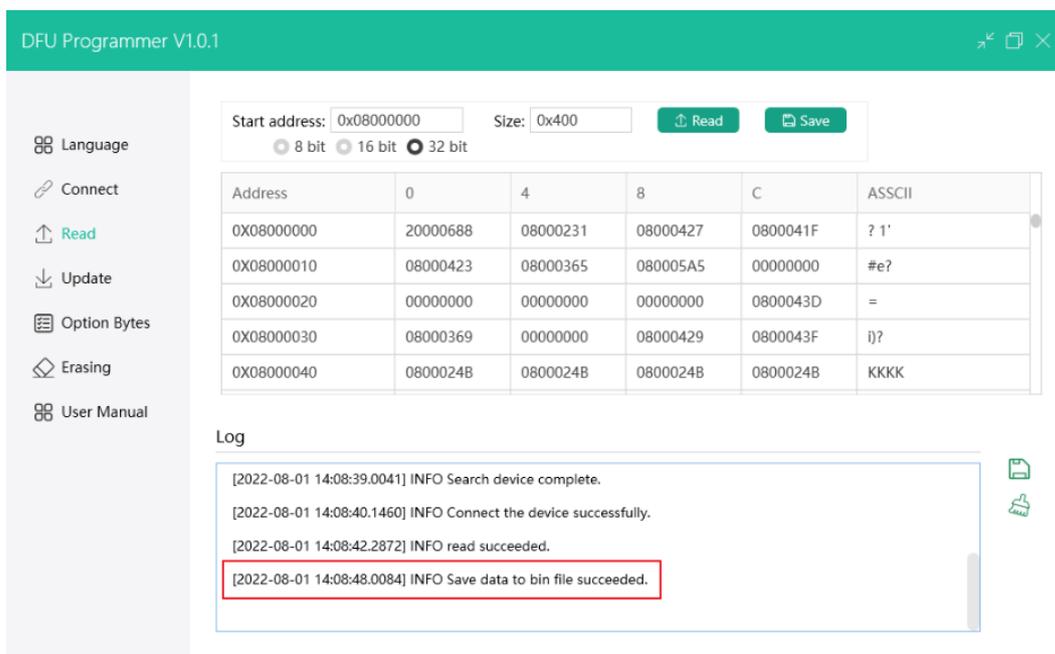


Figure 26 Save Data to PC



After the data is saved successfully, you can get the information of data successfully saved in the log window.

Figure 27 Data Saved Successfully



Then the program reading operation has been completed.

## 6 Version history

Table 3 Document Version History

Date	Version	Change History
June 6, 2022	1.0	New
Oct 26, 2022	1.1	Fixed the incorrect description of entering ISP mode in Section 5

# Statement

This document is formulated and published by Geehy Semiconductor Co., Ltd. (hereinafter referred to as “Geehy”). The contents in this document are protected by laws and regulations of trademark, copyright and software copyright. Geehy reserves the right to make corrections and modifications to this document at any time. Please read this document carefully before using Geehy products. Once you use the Geehy product, it means that you (hereinafter referred to as the “users”) have known and accepted all the contents of this document. Users shall use the Geehy product in accordance with relevant laws and regulations and the requirements of this document.

## 1. Ownership

This document can only be used in connection with the corresponding chip products or software products provided by Geehy. Without the prior permission of Geehy, no unit or individual may copy, transcribe, modify, edit or disseminate all or part of the contents of this document for any reason or in any form.

The “极海” or “Geehy” words or graphics with “®” or “™” in this document are trademarks of Geehy. Other product or service names displayed on Geehy products are the property of their respective owners.

## 2. No Intellectual Property License

Geehy owns all rights, ownership and intellectual property rights involved in this document.

Geehy shall not be deemed to grant the license or right of any intellectual property to users explicitly or implicitly due to the sale or distribution of Geehy products or this document.

If any third party’s products, services or intellectual property are involved in this document, it shall not be deemed that Geehy authorizes users to use the aforesaid third party’s products, services or intellectual property, unless otherwise agreed in sales order or sales contract.

## 3. Version Update

Users can obtain the latest document of the corresponding models when ordering Geehy products.

If the contents in this document are inconsistent with Geehy products, the agreement in the sales order or the sales contract shall prevail.

## 4. Information Reliability

The relevant data in this document are obtained from batch test by Geehy Laboratory or cooperative third-party testing organization. However, clerical errors in correction or errors caused by differences in testing environment may occur inevitably. Therefore, users should understand that Geehy does not bear any responsibility for such errors that may occur in this document. The relevant data in this document are only used to guide users as performance parameter reference and do not constitute Geehy’s guarantee for any product performance.

Users shall select appropriate Geehy products according to their own needs, and effectively verify and test the applicability of Geehy products to confirm that Geehy products meet their own needs, corresponding

standards, safety or other reliability requirements. If losses are caused to users due to the user's failure to fully verify and test Geehy products, Geehy will not bear any responsibility.

#### 5. Legality

USERS SHALL ABIDE BY ALL APPLICABLE LOCAL LAWS AND REGULATIONS WHEN USING THIS DOCUMENT AND THE MATCHING GEEHY PRODUCTS. USERS SHALL UNDERSTAND THAT THE PRODUCTS MAY BE RESTRICTED BY THE EXPORT, RE-EXPORT OR OTHER LAWS OF THE COUNTRIES OF THE PRODUCTS SUPPLIERS, GEEHY, GEEHY DISTRIBUTORS AND USERS. USERS (ON BEHALF OR ITSELF, SUBSIDIARIES AND AFFILIATED ENTERPRISES) SHALL AGREE AND PROMISE TO ABIDE BY ALL APPLICABLE LAWS AND REGULATIONS ON THE EXPORT AND RE-EXPORT OF GEEHY PRODUCTS AND/OR TECHNOLOGIES AND DIRECT PRODUCTS.

#### 6. Disclaimer of Warranty

THIS DOCUMENT IS PROVIDED BY GEEHY "AS IS" AND THERE IS NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

GEEHY WILL BEAR NO RESPONSIBILITY FOR ANY DISPUTES ARISING FROM THE SUBSEQUENT DESIGN OR USE BY USERS.

#### 7. Limitation of Liability

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL GEEHY OR ANY OTHER PARTY WHO PROVIDE THE DOCUMENT "AS IS", BE LIABLE FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, DIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE DOCUMENT (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY USERS OR THIRD PARTIES).

#### 8. Scope of Application

The information in this document replaces the information provided in all previous versions of the document.

© 2022 Geehy Semiconductor Co., Ltd. - All Rights Reserved