UM0001

ISP Programmer User manual

Introduction

This user manual gives an overview of Artery ISP Programmer. ISP Programmer acts a graphic Interface application designed to facilitate the use of ARTERY MCU. With the help of this programmer, users can perform ARTERY MCU devices through UART or USB ports.

Contents

List of tables

List of figures

1 Introduction

1.1 Environmental requirements

- **Software requirements**

Windows 7 and above are required.

Software version below 2.0.04, .Net framework 4.0 is required.

Software version 2.0.04 and above, .Net framework 4.6 is required.

- **Hardware requirements**

Serial communication port (COM).

USB communication port.

1.2 Glossary

- **ISP:**

This refer to in-system programming so that user can directly perform write or erase operations on the chip.

- **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 Installation

- **Hardware installation**

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

- **USB DFU driver installation**

If the USB DFU communication is used, the USB DFU driver must be installed. Please refer to the chapter USB DFU driver installation for detailed information.

- **Software installation**

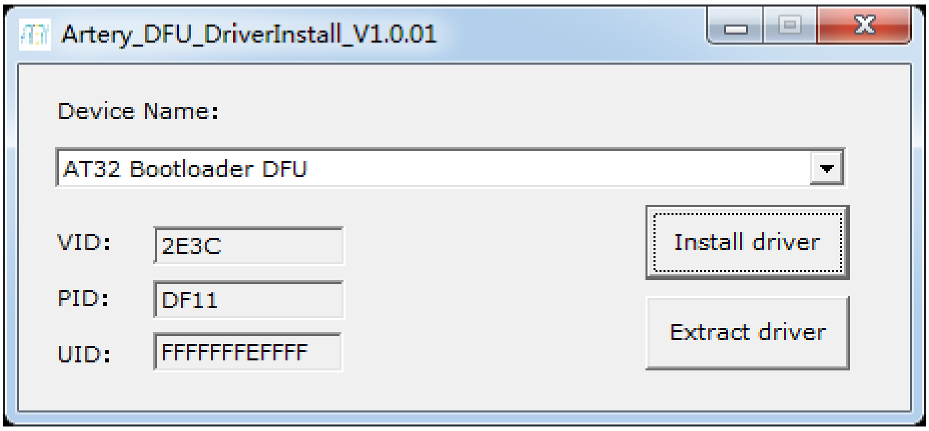
This software is not required, just directly run the executable program "ArteryISPProgrammer.exe".

3 USB DFU driver installation

Artery provides the USB DFU driver automatic installation program "Artery_DFU_DriverInstall.exe". Double-click it to enter the installation interface. (As shown in Figure 1)

The driver installation program will automatically scan all the "AT32 Bootloader DFU" devices connected to the computer. When the devices are connected, the "VID", "PID", and "UID" of each device can be displayed respectively.

Figure 1. DFU driver install



Device Name: AT32 Bootloader DFU

VID: 2E3C

PID: DF11

UID: FFFFFFFEFFFF

3.1 Install driver automatically

Click on "***Install driver***" button to start the automatic installation of the driver.

If the installation is successful, a successful installation message will be displayed.

If failed, an error message will be displayed.

If the driver is already installed, "***Install driver***" will become "***Reinstall driver***". Click on this button will reinstall the driver.

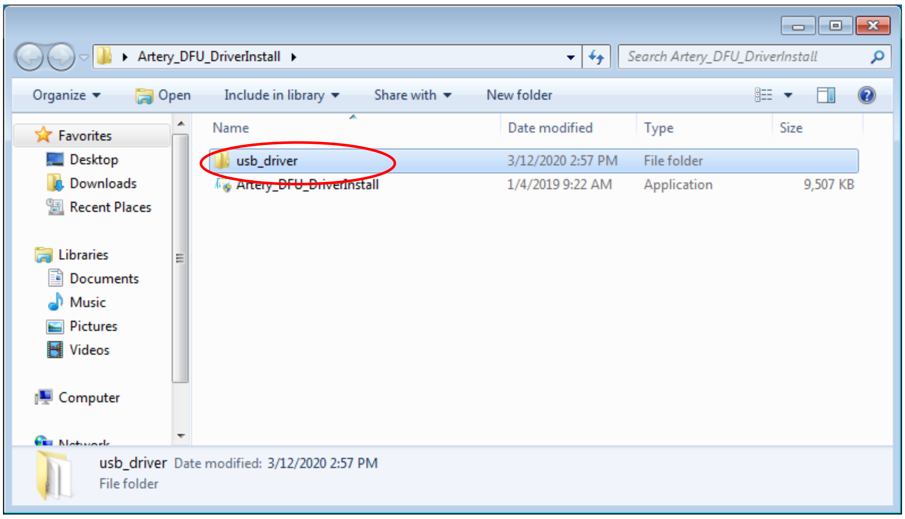
3.2 Install driver manually

When the automatic installation failed or the user needs to install the driver manually, refer to this chapter for manual Installation.

Click on "***Extract driver***" button, a driver installation package ("***usb_driver***" folder) will be generated in the current Directory (As shown in Figure 2).

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

Figure 2. Manual install-driver location



Name	Date modified	Type	Size
usb_driver	3/12/2020 2:57 PM	File folder	
Artery_DFU_DriverInstall	1/4/2019 9:22 AM	Application	9,507 KB

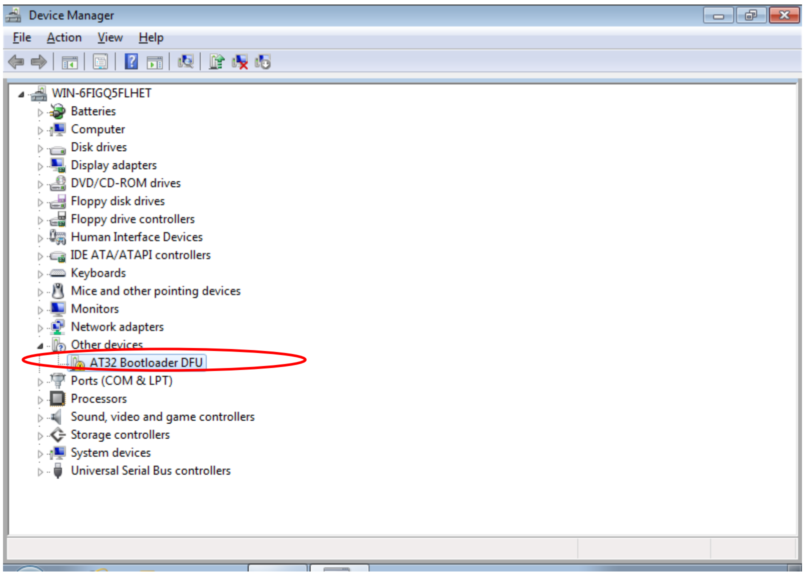
usb_driver Date modified: 3/12/2020 2:57 PM
File folder

The procedures of manual installation are as follows (take windows7 as an example):

- Open the "***Device Manager***" (As shown in Figure 3)

First make sure that the "***AT32 Bootloader DFU***" device is properly connected to the computer.

Figure 3. Manual install-device manager

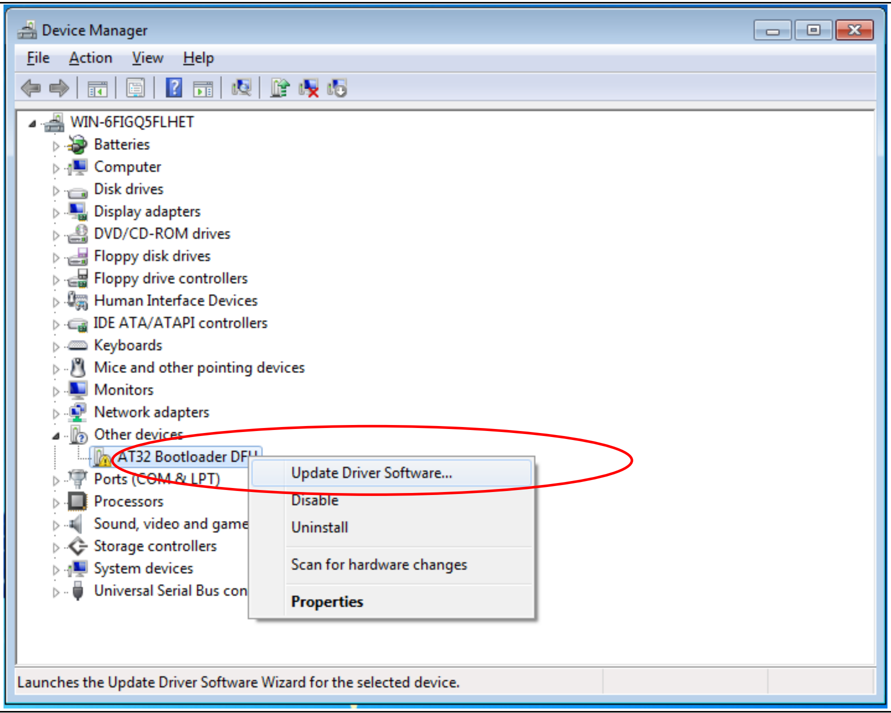


In this case, the "***Device Manager***" will scan the device "***AT32 Bootloader DFU***" without driver installed.

If the device "***AT32 Bootloader DFU***" is not found, please rescan it, that is, click on the "***Device Manager***"-"***Action***" menu and select "***Scan for hardware changes***".

- Right-click on the device "***AT32 Bootloader DFU***" and select "***Update Driver Software***" (As shown in Figure 4).

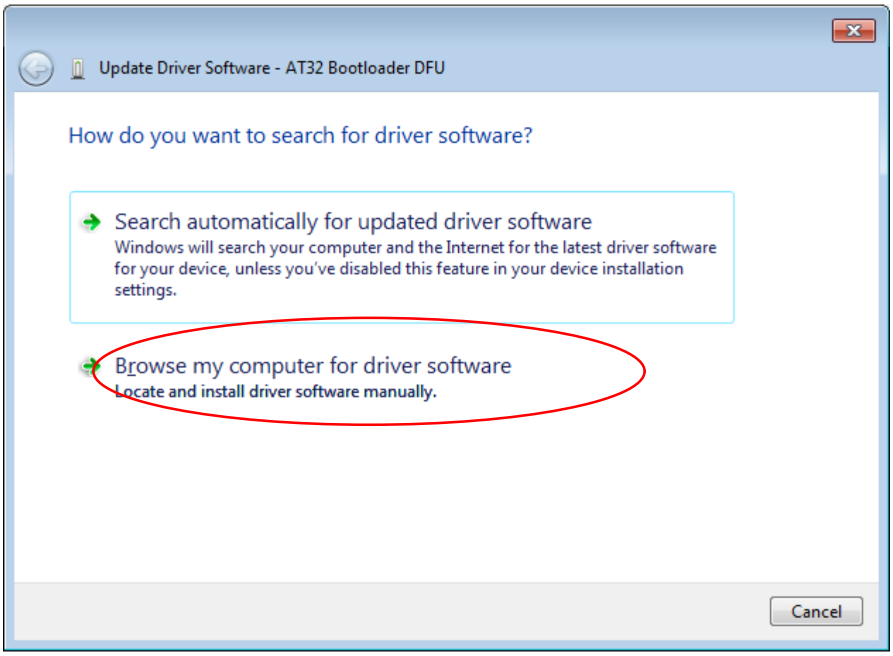
Figure 4. Manual install-update driver software



Launches the Update Driver Software Wizard for the selected device.

- Select "***Browse my computer for driver software***". (As shown in Figure 5)

Figure 5. Manual install-browse my computer for driver



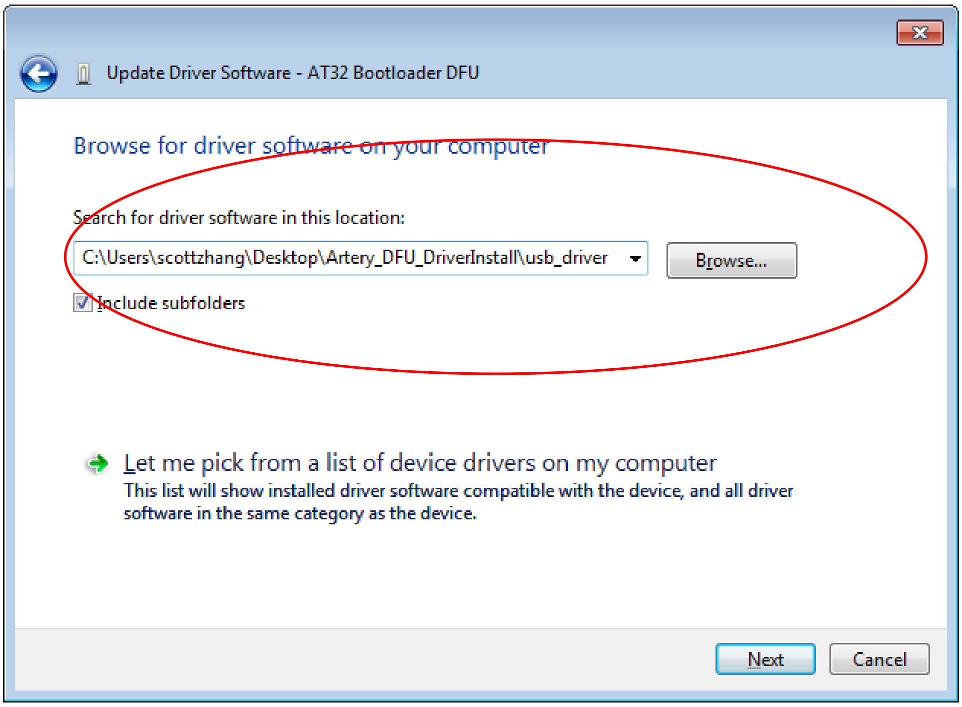
How do you want to search for driver software?

Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

Browse my computer for driver software
Locate and install driver software manually.

- Please select the location of the driver correctly, that is, click on "***Extract driver***" to generate a driver installation package ("***usb_driver***" folder). Then click on "***Next***" (As shown in Figure 6).

Figure 6. Manual install-select driver software



Update Driver Software - AT32 Bootloader DFU

Browse for driver software on your computer

Search for driver software in this location:

C:\Users\scottzhang\Desktop\Artery_DFU_DriverInstall\usb_driver

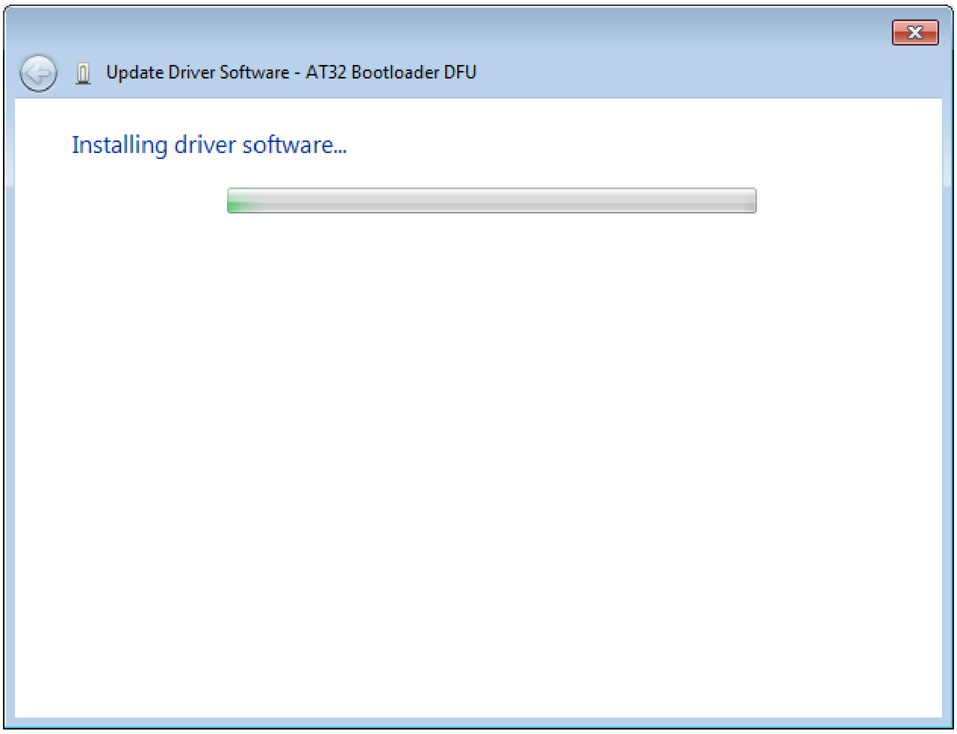
☑ Include subfolders

Let me pick from a list of device drivers on my computer

This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

- Installing driver softwar. (As shown in Figure 7).

Figure 7. Manual install-driver software installing



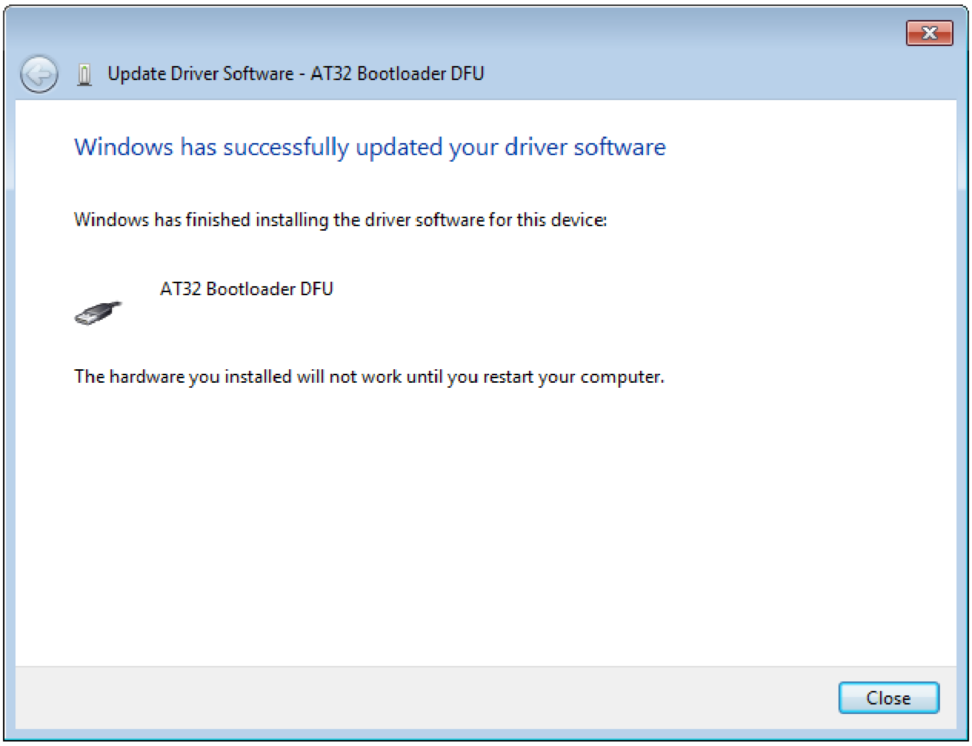
Update Driver Software - AT32 Bootloader DFU

Installing driver software...

Please wait for the driver installation to be complete. After it is completed, click on "***Close***" (As shown in Figure 8).

The manual installation of the driver is now completed.

Figure 8. Manual install successful



Update Driver Software - AT32 Bootloader DFU

Windows has successfully updated your driver software

Windows has finished installing the driver software for this device:

AT32 Bootloader DFU

The hardware you installed will not work until you restart your computer.

4 Product part number and interface

4.1 AT32F403 part number and interface

Table 1. AT32F403 part number and interface

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F403ZCT6	256KB	LQFP144	Y	Y	Y
AT32F403VCT6	256KB	LQFP100	Y	Y	Y
AT32F403RCT6	256KB	LQFP64	Y	Y	Y
AT32F403CCT6	256KB	LQFP48	Y	Y	Y
AT32F403ZGT6	1024KB	LQFP144	Y	Y	Y
AT32F403VGT6	1024KB	LQFP100	Y	Y	Y
AT32F403RGT6	1024KB	LQFP64	Y	Y	Y
AT32F403CGT6	1024KB	LQFP48	Y	Y	Y
AT32F403ZET6	512KB	LQFP144	Y	Y	Y
AT32F403VET6	512KB	LQFP100	Y	Y	Y
AT32F403RET6	512KB	LQFP64	Y	Y	Y
AT32F403CET6	512KB	LQFP48	Y	Y	Y
AT32F403CGU6	1024KB	QFN48	Y	Y	Y
AT32F403CEU6	512KB	QFN48	Y	Y	Y
AT32F403CCU6	256KB	QFN48	Y	Y	Y
AT32F403CBT6	128KB	LQFP48	Y	Y	Y
AR8F403CGT6	1024KB	LQFP48	Y	Y	Y

Table 2. AT32F403 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	AT32F403ZGT6/ AT32F403VGT6	PD5	PD6
	Others	PA2	PA3
DFU	All	PA11	PA12

4.2 AT32F413 part number and interface

Table 3. AT32F413 part number and interface

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F413RCT7	256KB	LQFP64	Y	Y	Y
AT32F413CCT7	256KB	LQFP48	Y	Y	Y
AT32F413KCU7-4	256KB	QFN32	Y	Y	Y
AT32F413CCU7	256KB	QFN48	Y	Y	Y
AT32F413RBT7	128KB	LQFP64	Y	Y	Y
AT32F413CBT7	128KB	LQFP48	Y	Y	Y
AT32F413KBU7-4	128KB	QFN32	Y	Y	Y
AT32F413CBU7	128KB	QFN48	Y	Y	Y

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F413C8T7	64KB	LQFP48	Y	Y	Y
AT32FEBKC8T7	64KB	LQFP48	Y	Y	Y

Table 4. AT32F413 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	All	PA2	PA3
DFU	All	PA11	PA12

4.3 AT32F415 part number and interface

Table 5. AT32F415 part number and interface

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F415RCT7	256KB	LQFP64	Y	Y	Y
AT32F415RCW	256KB	LQFP64	Y	Y	Y
AT32F415RCT7-7	256KB	LQFP64	Y	Y	Y
AT32F415CCT7	256KB	LQFP48	Y	Y	Y
AT32F415KCU7-4	256KB	QFN32	Y	Y	Y
AT32F415RBT7	128KB	LQFP64	Y	Y	Y
AT32F415RBW	128KB	LQFP64	Y	Y	Y
AT32F415RBT7-7	128KB	LQFP64	Y	Y	Y
AT32F415CBT7	128KB	LQFP48	Y	Y	Y
AT32F415KBU7-4	128KB	QFN32	Y	Y	Y
AT32F415R8T7	64KB	LQFP64	Y	Y	Y
AT32F415R8T7-7	64KB	LQFP64	Y	Y	Y
AT32F415C8T7	64KB	LQFP48	Y	Y	Y
AT32F415K8U7-4	64KB	QFN32	Y	Y	Y
AT32F415CCU7	256KB	QFN48	Y	Y	Y
AT32F415CBU7	128KB	QFN48	Y	Y	Y

Table 6. AT32F415 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	All	PA2	PA3
DFU	All	PA11	PA12

4.4 AT32F403A/F407 part number and interface

Table 7. AT32F403A/F407 part number and interface

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F403AVCT7	256KB	LQFP100	Y	Y	Y
AT32F403ARCT7	256KB	LQFP64	Y	Y	Y

AT32F403ACCT7	256KB	LQFP48	Y	Y	Y
AT32F403ACCU7	256KB	QFN48	Y	Y	Y
AT32F403AVET7	512KB	LQFP100	Y	Y	Y
AT32F403ARET7	512KB	LQFP64	Y	Y	Y
AT32F403ACET7	512KB	LQFP48	Y	Y	Y
AT32F403ACEU7	512KB	QFN48	Y	Y	Y
AT32F403AVGT7	1024KB	LQFP100	Y	Y	Y
AT32F403ARGT7	1024KB	LQFP64	Y	Y	Y
AT32F403ACGT7	1024KB	LQFP48	Y	Y	Y
AT32F403ACGU7	1024KB	QFN48	Y	Y	Y
AT32F407VCT7	256KB	LQFP100	Y	Y	Y
AT32F407RCT7	256KB	LQFP64	Y	Y	Y
AT32F407VET7	512KB	LQFP100	Y	Y	Y
AT32F407RET7	512KB	LQFP64	Y	Y	Y
AT32F407VGT7	1024KB	LQFP100	Y	Y	Y
AT32F407RGT7	1024KB	LQFP64	Y	Y	Y
AT32F407AVCT7	256KB	LQFP100	Y	Y	Y
AR8F403CGT6-A	1024KB	LQFP48	Y	Y	Y
AT32F403AVGW	1024KB	LQFP100	Y	Y	Y

Table 8. AT32F403A/F407 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	AT32F403AVGT7/AT32F407VGT7	PD5	PD6
	Others	PA2	PA3
DFU	All	PA11	PA12

4.5 AT32F421 part number and interface

Table 9. AT32F421 part number and interface

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F421C8T7	64KB	LQFP48	Y	Y	N
AT32F421C8W	64KB	LQFP48	Y	Y	N
AT32F421K8T7	64KB	LQFP32	Y	Y	N
AT32F421K8U7	64KB	QFN32	Y	Y	N
AT32F421K8U7-4	64KB	QFN32	Y	Y	N
AT32F421F8U7	64KB	QFN20	Y	Y	N
AT32F421F8P7	64KB	TSSOP20	Y	Y	N
AT32F421PF8P7	64KB	TSSOP20	Y	Y	N
AT32F421G8U7	64KB	QFN28	Y	Y	N
AT32F421C6T7	32KB	LQFP48	Y	Y	N
AT32F421K6T7	32KB	LQFP32	Y	Y	N
AT32F421K6U7	32KB	QFN32	Y	Y	N

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F421K6U7-4	32KB	QFN32	Y	Y	N
AT32F421F6U7	32KB	QFN20	Y	Y	N
AT32F421F6P7	32KB	TSSOP20	Y	Y	N
AT32F421G6U7	32KB	QFN28	Y	Y	N
AT32F421C4T7	16KB	LQFP48	Y	Y	N
AT32F421K4T7	16KB	LQFP32	Y	Y	N
AT32F421K4U7	16KB	QFN32	Y	Y	N
AT32F421K4U7-4	16KB	QFN32	Y	Y	N
AT32F421F4U7	16KB	QFN20	Y	Y	N
AT32F421F4P7	16KB	TSSOP20	Y	Y	N
AT32F421PF4P7	16KB	TSSOP20	Y	Y	N
AT32F421G4U7	16KB	QFN28	Y	Y	N
AT32F4212C8T7	64KB	LQFP48	Y	Y	N

Table 10. AT32F421 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	All	PA2	PA3

4.6 AT32F435/F437 part number and interface

Table 11. AT32F435/F437 part number and interface

Part number	Flash size	Pin	Interface				
			UART1	UART2	DFU	DFU1	DFU2
AT32F435ZMT7	4032KB	LQFP144	Y	Y	Y	Y	Y
AT32F435ZGT7	1024KB	LQFP144	Y	Y	Y	Y	Y
AT32F435ZCT7	256KB	LQFP144	Y	Y	Y	Y	Y
AT32F435VMT7	4032KB	LQFP100	Y	Y	Y	Y	Y
AT32F435VGT7	1024KB	LQFP100	Y	Y	Y	Y	Y
AT32F435VCT7	256KB	LQFP100	Y	Y	Y	Y	Y
AT32F435RMT7	4032KB	LQFP64	Y	Y	Y	Y	Y
AT32F435RGT7	1024KB	LQFP64	Y	Y	Y	Y	Y
AT32F435RCT7	256KB	LQFP64	Y	Y	Y	Y	Y
AT32F435CMT7	4032KB	LQFP48	Y	Y	Y	Y	Y
AT32F435CGT7	1024KB	LQFP48	Y	Y	Y	Y	Y
AT32F435CCT7	256KB	LQFP48	Y	Y	Y	Y	Y
AT32F435CMU7	4032KB	QFN48	Y	Y	Y	Y	Y
AT32F435CGU7	1024KB	QFN48	Y	Y	Y	Y	Y
AT32F435CCU7	256KB	QFN48	Y	Y	Y	Y	Y
AT32F437ZMT7	4032KB	LQFP144	Y	Y	Y	Y	Y
AT32F437ZGT7	1024KB	LQFP144	Y	Y	Y	Y	Y
AT32F437ZCT7	256KB	LQFP144	Y	Y	Y	Y	Y

Part number	Flash size	Pin	Interface				
			UART1	UART2	DFU	DFU1	DFU2
AT32F437VMT7	4032KB	LQFP100	Y	Y	Y	Y	Y
AT32F437VGT7	1024KB	LQFP100	Y	Y	Y	Y	Y
AT32F437VCT7	256KB	LQFP100	Y	Y	Y	Y	Y
AT32F437RMT7	4032KB	LQFP64	Y	Y	Y	Y	Y
AT32F437RGT7	1024KB	LQFP64	Y	Y	Y	Y	Y
AT32F437RCT7	256KB	LQFP64	Y	Y	Y	Y	Y
AT32F435CMT7-E	4032KB	LQFP48	Y	Y	Y	Y	Y
AT32F435CGT7-W	960KB	LQFP48	Y	Y	Y	Y	Y
AT32F435CCT7-W	192KB	LQFP48	Y	Y	Y	Y	Y
AT32F435CMU7-E	4032KB	QFN48	Y	Y	Y	Y	Y
AT32F435CGU7-W	960KB	QFN48	Y	Y	Y	Y	Y

Table 12. AT32F435/F437 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	AT32F435/F437ZxT7、AT32F435/F437VxT7	PD5	PD6
	Others	PA2	PA3
UART3	AT32F435/F437ZxT7、AT32F435/F437VxT7、 AT32F435/F437RxT7	PC10	PC11
	Others	PB10	PB11
DFU1	All	PA11	PA12
DFU2	All	PB14	PB15

4.7 AT32WB415 part number and interface

Table 13. AT32WB415 part number and interface

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32WB415CCU7-7	256KB	QFN48	N	Y	Y

Table 14. AT32WB415 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	None	None
UART2	All	PA2	PA3
DFU	All	PA11	PA12

4.8 AT32F425 part number and interface

Table 15. AT32F425 part number and interface

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F425R8T7	64KB	64LQFP	Y	Y	N
AT32F425R6T7	32KB	64LQFP	Y	Y	N
AT32F425R8T7-7	64KB	64LQFP	Y	Y	N

Part number	Flash size	Pin	Interface		
			UART1	UART2	DFU
AT32F425R6T7-7	32KB	64LQFP	Y	Y	N
AT32F425C8T7	64KB	48LQFP	Y	Y	N
AT32F425C6T7	32KB	48LQFP	Y	Y	N
AT32F425C8U7	64KB	48QFN	Y	Y	N
AT32F425C6U7	32KB	48QFN	Y	Y	N
AT32F425K8T7	64KB	32LQFP	Y	Y	N
AT32F425K6T7	32KB	32LQFP	Y	Y	N
AT32F425K8U7-4	64KB	32QFN	Y	Y	N
AT32F425K6U7-4	32KB	32QFN	Y	Y	N
AT32F425F8P7	64KB	20TSSOP	Y	Y	N
AT32F425F6P7	32KB	20TSSOP	Y	Y	N
AT32F425G8U7	64KB	28QFN	Y	Y	N
AT32F425G6U7	32KB	28QFN	Y	Y	N

Table 16. AT32F425 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	All	PA2	PA3

4.9 AT32L021 part number and interface

Table 17. AT32L021 part number and interface

Part number	Flash Size	Pin	Interface		
			UART1	UART2	DFU
AT32L021C8T7	64KB	48LQFP	Y	Y	N
AT32L021C6T7	32KB	48LQFP	Y	Y	N
AT32L021C4T7	16KB	48LQFP	Y	Y	N
AT32L021K8T7	64KB	32LQFP	Y	Y	N
AT32L021K6T7	32KB	32LQFP	Y	Y	N
AT32L021K4T7	16KB	32LQFP	Y	Y	N
AT32L021K8U7	64KB	32QFN	Y	Y	N
AT32L021K6U7	32KB	32QFN	Y	Y	N
AT32L021K4U7	16KB	32QFN	Y	Y	N
AT32L021K8U7-4	64KB	32QFN	Y	Y	N
AT32L021K6U7-4	32KB	32QFN	Y	Y	N
AT32L021K4U7-4	16KB	32QFN	Y	Y	N
AT32L021F8U7	64KB	20QFN	Y	Y	N
AT32L021F6U7	32KB	20QFN	Y	Y	N
AT32L021F4U7	16KB	20QFN	Y	Y	N
AT32L021F8P7	64KB	20TSSOP	Y	Y	N
AT32L021F6P7	32KB	20TSSOP	Y	Y	N

Part number	Flash Size	Pin	Interface		
			UART1	UART2	DFU
AT32L021F4P7	16KB	20TSSOP	Y	Y	N
AT32L021G8U7	64KB	28QFN	Y	Y	N
AT32L021G6U7	32KB	28QFN	Y	Y	N
AT32L021G4U7	16KB	28QFN	Y	Y	N

Table 18. AT32L021 GPIO Pin Map

IP	MCU	TX Pin/DM	RX Pin/DP
UART1	All	PA9	PA10
UART2	All	PA2	PA3

5 User Interface

5.1 Connection settings

On this page, you can select the corresponding connection mode, that is, the interface type: UART or DFU.

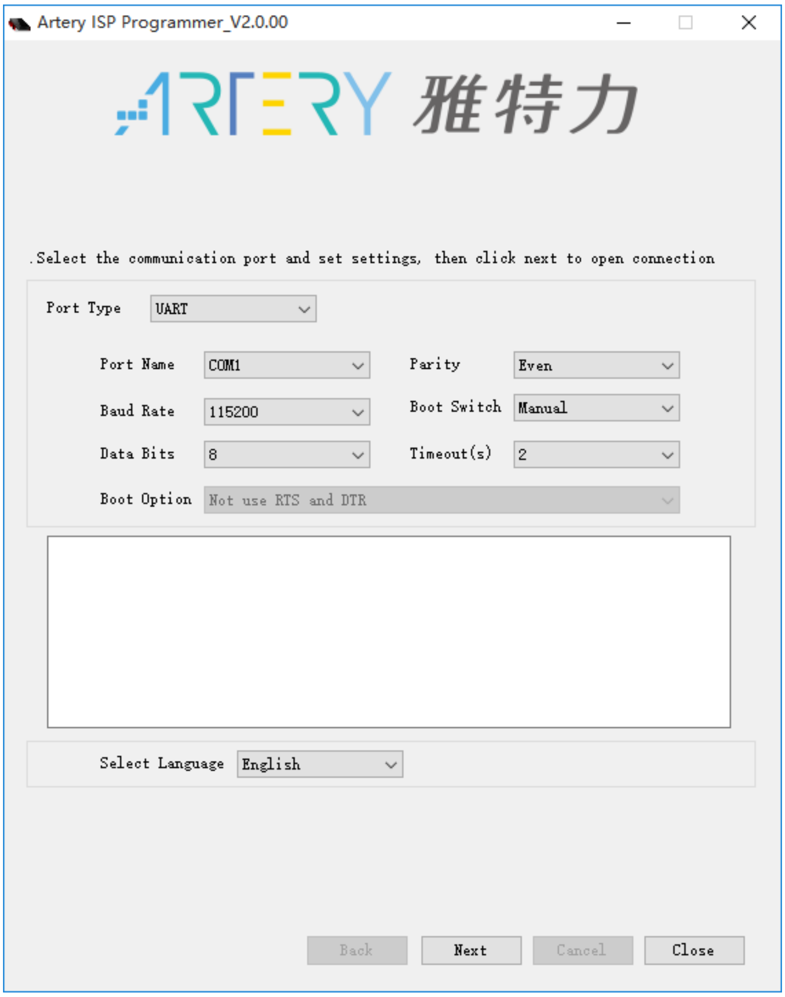
5.1.1 UART connection

After using the UART connection, you can select the serial interface to be operated and make related settings (As shown in Figure 9). Please ensure that the device to be operated is properly connected to the selected serial interface.

When "***Boot Switch***" is set to "***Manual***", you need to manually reset the device to restart the "***BootLoader***" program in device. If the device supports automatic connection circuitry, reset can be controlled by controlling the DTR and RTS signals. You can select the control mode of the current device in "***Boot Option***".

After setting, click on "***Next***", if the connection is successful, skip to the next page. If failed, an error message will be displayed.

Figure 9. UART connection window



Artery ISP Programmer_V2.0.00

.Select the communication port and set settings, then click next to open connection

Port Type: UART

Port Name: COM1

Baud Rate: 115200

Data Bits: 8

Parity: Even

Boot Switch: Manual

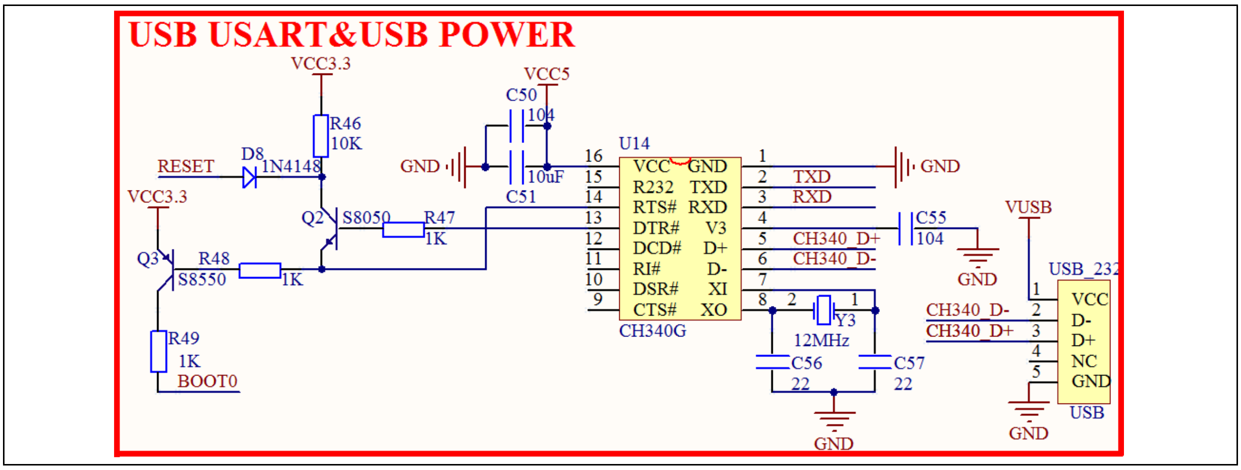
Timeout(s): 2

Boot Option: Not use RTS and DTR

Select Language: English

The USB serial interface automatic connection circuit can be designed with reference to the following figure. (As shown in Figure 10):

Figure 10. USB interface auto connection diagram



The combination of Q2 and Q3 in figure 10 constitutes the automatic connection circuit of the development board, which only should be set in the ISP software: DTR low level reset, RTS high level to load bootloader. In this case, it can be connected automatically without setting B0 manually and pressing reset button. Among them, RESET is the reset signal of the 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 ISP controls DTR to output low level, then DTR_N output high, and RTS is set high, then RTS_N output is low, so Q3 is turned on and BOOT0 is pulled up, that is, BOOT0 is set to 1, and Q2 will also be turned on at the same time, the reset pin of chip is pulled low to realize reset. Then, after a delay of 100ms, the ISP controls DTR to be high level, then DTR_N output low level, and RTS maintains high, then RTS_N continues to be at low level, in this case, the reset pin of chip becomes high since Q2 is no longer on, and the chip ends reset, but BOOT0 remains at 1, and enters the BootLoader Mode, and then ISP starts to connect and download the code.

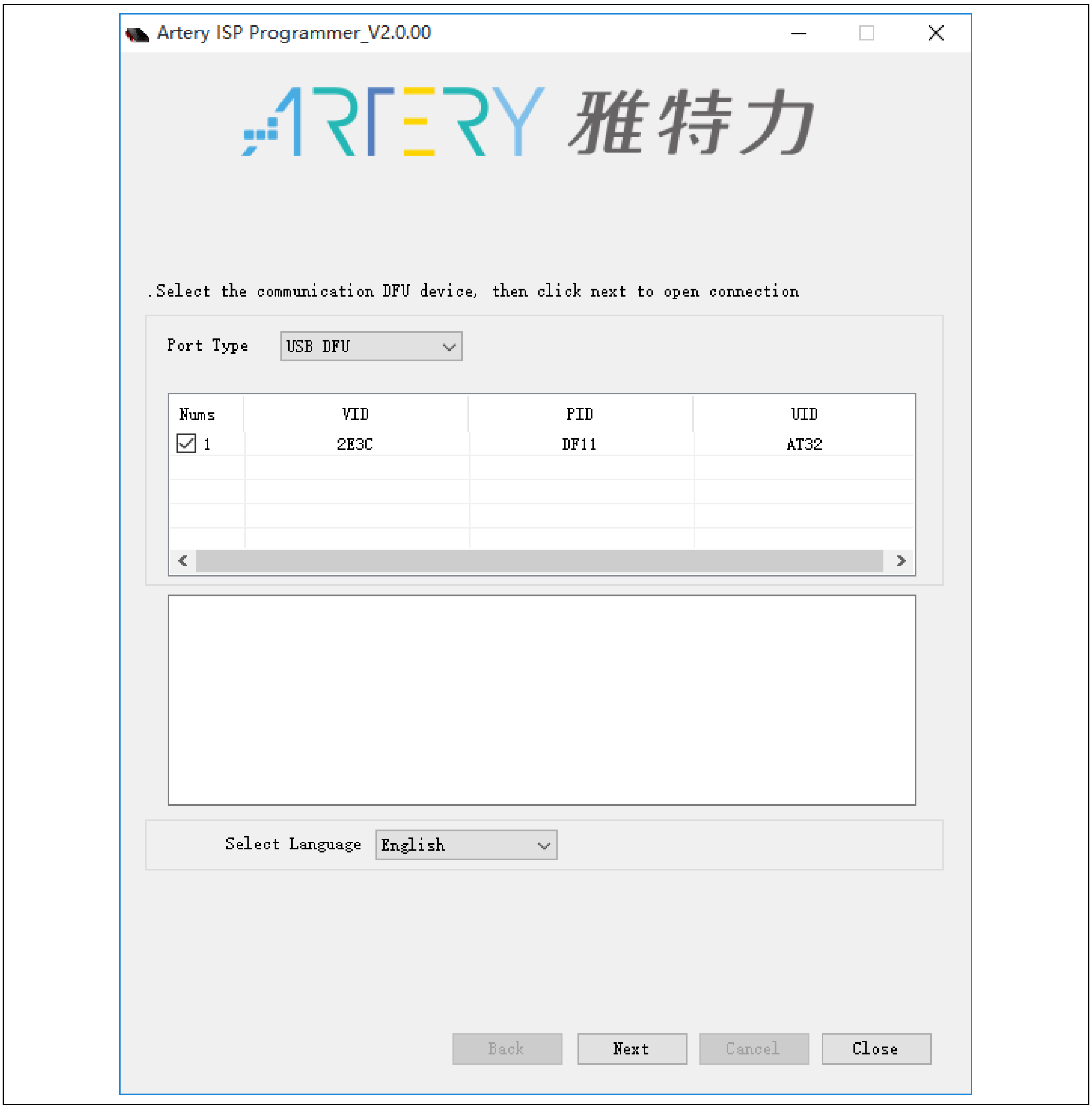
5.1.2 DFU connection

After selecting the DFU connection, you can select the DFU device to be connected (as shown in Figure 11). Please ensure that the device to be operated is connected to the corresponding USB port of PC.

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

After selecting the DFU device to be connected, click on "***Next***", and if the connection is successful, skip to the next page. If failed, an error message will be displayed.

Figure 11. DFU connection window



.Select the communication DFU device, then click next to open connection

Port Type: USB DFU

Nums	VID	PID	UID
☑ 1	2E3C	DF11	AT32

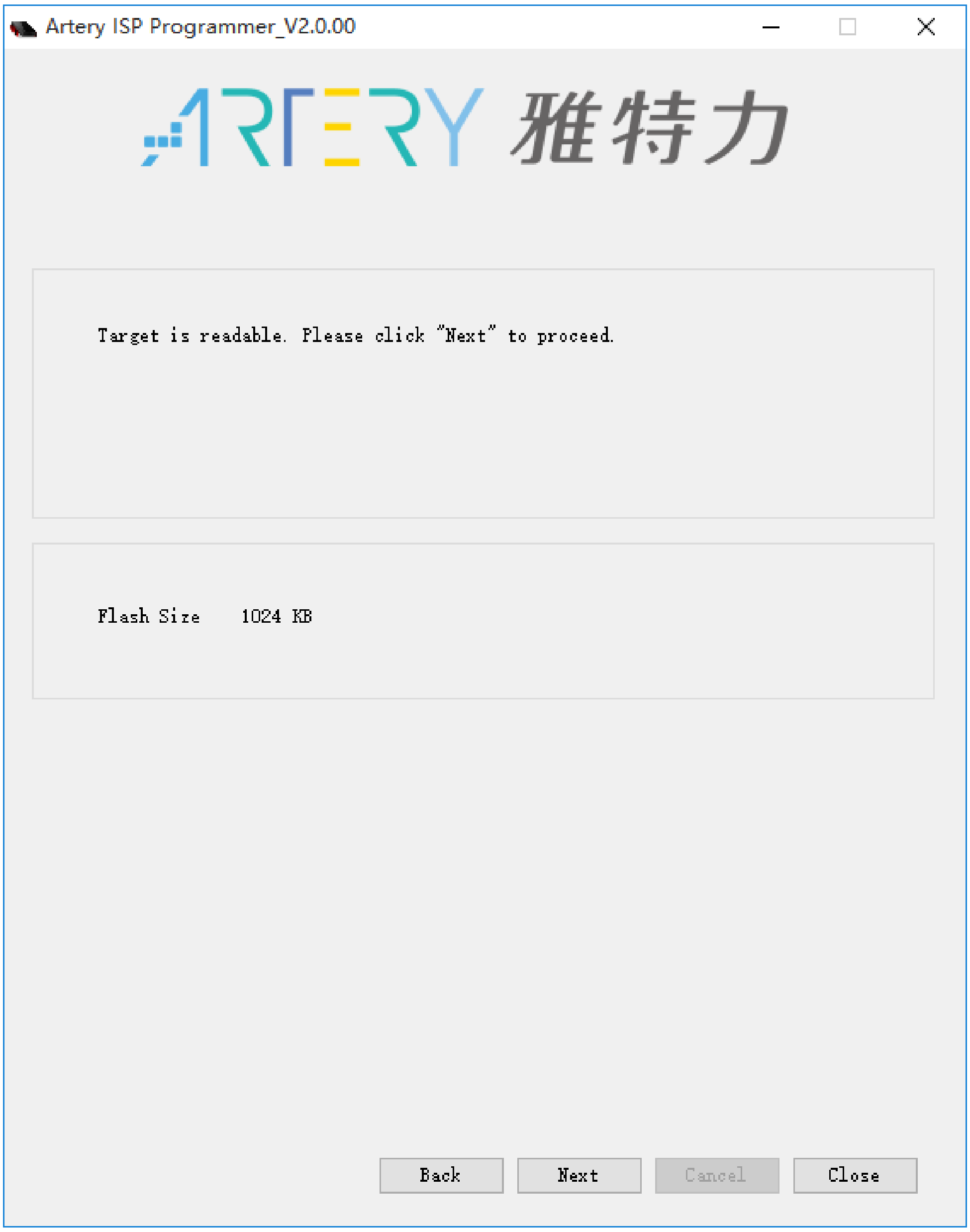
Select Language: English

5.2 Flash status page

The connection is now set up, and the status of Flash is displayed on this page (As shown in Figure 12).

If "***Access protection***" is enabled, the device will restrict the use of some functions, that is, it is only allowed to use Firmware CRC function /Flash CRC/ Disable access protection function.

Figure 12. Flash status window



Target is readable. Please click "Next" to proceed.

Flash Size 1024 KB

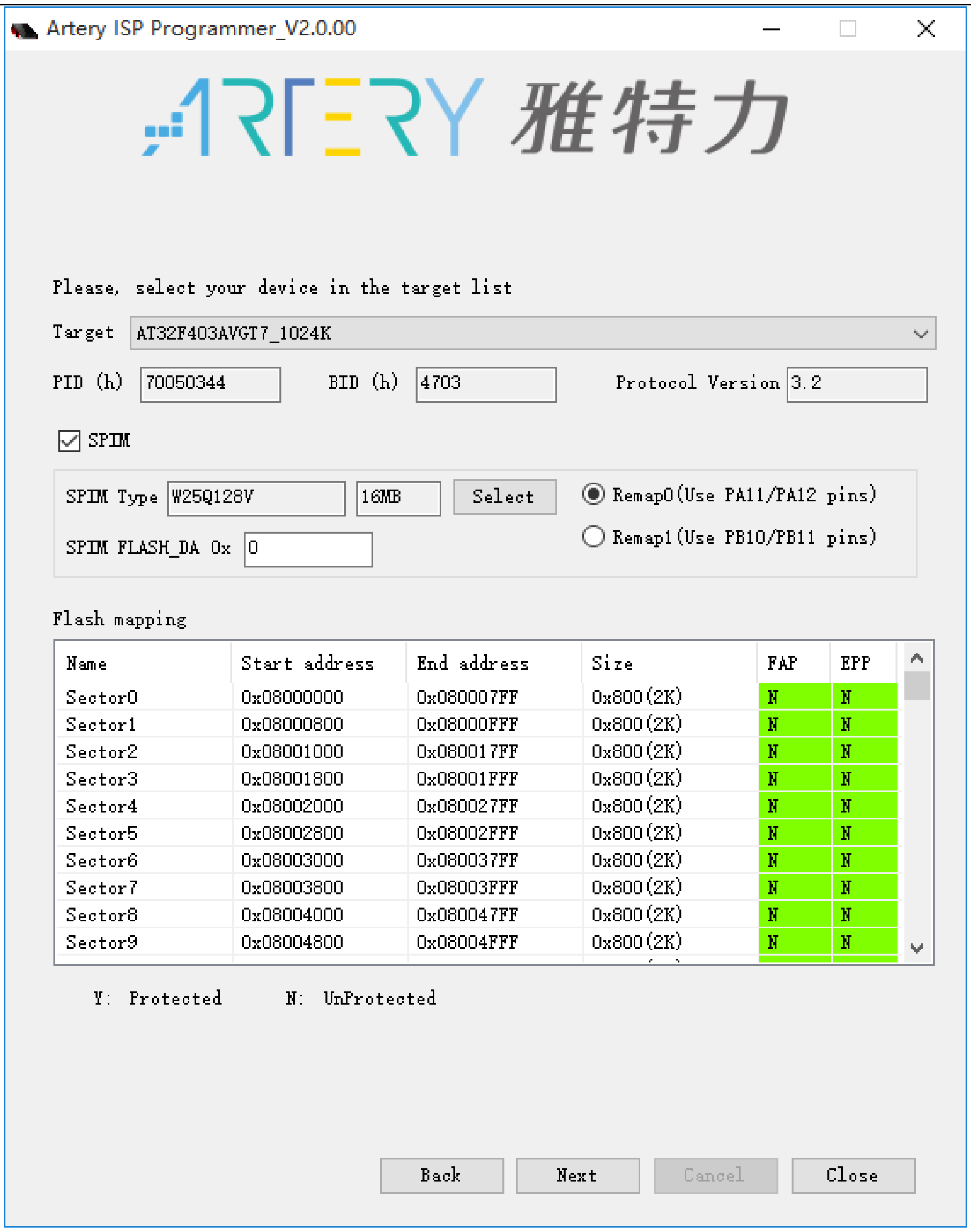
5.3 Device Information page

This page displays device-related information such as target device, PID, BID, protocol version, Flash mapping and Flash protection status (As shown in Figure 13).

If SPIM is connected, please check "***SPIM***" and select "***SPIM Type***". The SPIM size depends on the "SPIM Type".If SPIM encryption is required, set the SPIM FLASH_DA.

In this case, all sectors of main flash and SPIM are automatically displayed in the Flash map.

Figure 13. Device information



Please, select your device in the target list

Target AT32F403AVGT7_1024K

PID (h) 70050344 BID (h) 4703 Protocol Version 3.2

☑ SPIM

SPIM Type W25Q128V 16MB

SPIM FLASH_DA 0x 0

● Remap0(Use PA11/PA12 pins)

○ Remap1(Use PB10/PB11 pins)

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

In UART communication mode:

1. AT32F403 series MCUs support SPIM.
2. AT32F413 series MCUs support SPIM.
3. AT32F415 series MCUs do not support SPIM.

4. AT32F403A series MCUs support SPIM.
5. AT32F407 series MCUs support SPIM.
6. AT32F421 series MCUs do not support SPIM.
7. AT32F435 series MCUs do not support SPIM.
8. AT32F437 series MCUs do not support SPIM.
9. AT32F425 series MCUs do not support SPIM.
10. AT32L021 series MCUs do not support SPIM.

In DFU communication mode:

1. AT32F403 series MCUs do not support SPIM.
2. AT32F413KCU7-4 and AT32F413KBU7-4 in the AT32F413 series do not support SPIM; other models of AT32F413 series MCUs support SPIM.
3. AT32F415 series MCUs do not support SPIM.
4. AT32F403A series MCUs support SPIM.
5. AT32F407 series MCUs support SPIM.
6. AT32F421 series MCUs do not support DFU and SPIM.
7. AT32F435 series MCUs do not support SPIM.
8. AT32F437 series MCUs do not support SPIM.
9. AT32F425 series MCUs do not support DFU and SPIM.
10. AT32L021 series MCUs do not support DFU and SPIM.

■ Checked “***SPIM***”

Allows operation on SPIM.

■ Unchecked “***SPIM***”

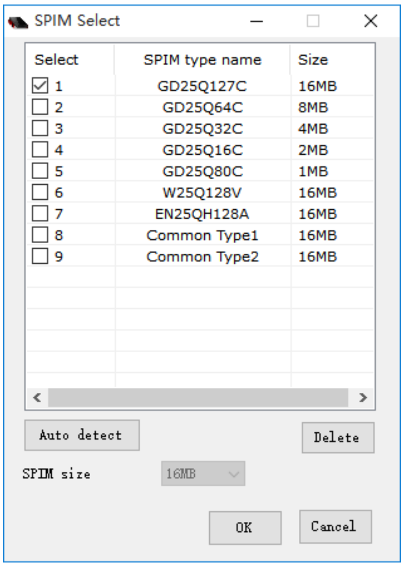
Operation on SPIM is not allowed.

■ SPIM Type

You can select SPIM type with “***Select***" button.

Click on "***Select***" button, a dialog box will pop up. (As shown in Figure 14)

Figure 14. SPIM selection



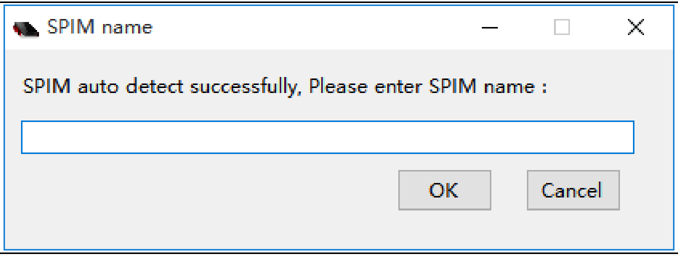
SPIM Select

Select	SPIM type name	Size
☑ 1	GD25Q127C	16MB
□ 2	GD25Q64C	8MB
□ 3	GD25Q32C	4MB
□ 4	GD25Q16C	2MB
□ 5	GD25Q80C	1MB
□ 6	W25Q128V	16MB
□ 7	EN25QH128A	16MB
□ 8	Common Type1	16MB
□ 9	Common Type2	16MB

SPIM size: 16MB

Auto detect：it will automatically detect whether the SPIM meets the requirements of this software operation. (Auto Detect will overwrite some data of SPIM, please use it with caution)
If the detection is successful, a dialog box will pop up. (As shown in Figure 15)

Figure 15. SPIM name

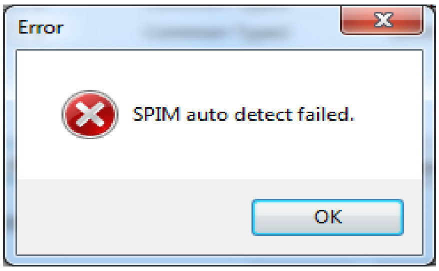


SPIM auto detect successfully, Please enter SPIM name :

Click on "***OK***" to add the detected SPIM to the SPIM list.
Click on "***Cancel***" to cancel auto detect.

If Auto detect failed, a failure dialog box will pop up. (As shown in Figure 16)

Figure 16. SPIM name



SPIM auto detect failed.

SPIM size: this is used to select SPIM size, except for the default type.
Delete: delete the selected SPIM from the list, except for the default type.
OK: SPIM selected.
Cancel: cancel.

- SPIM Size

SPIM size is depending on the selected SPIM type.

- SPIM FLASH_DA

Set the encryption range when downloading files to the SPIM. The encryption range calculates starting from address 0x08400000.

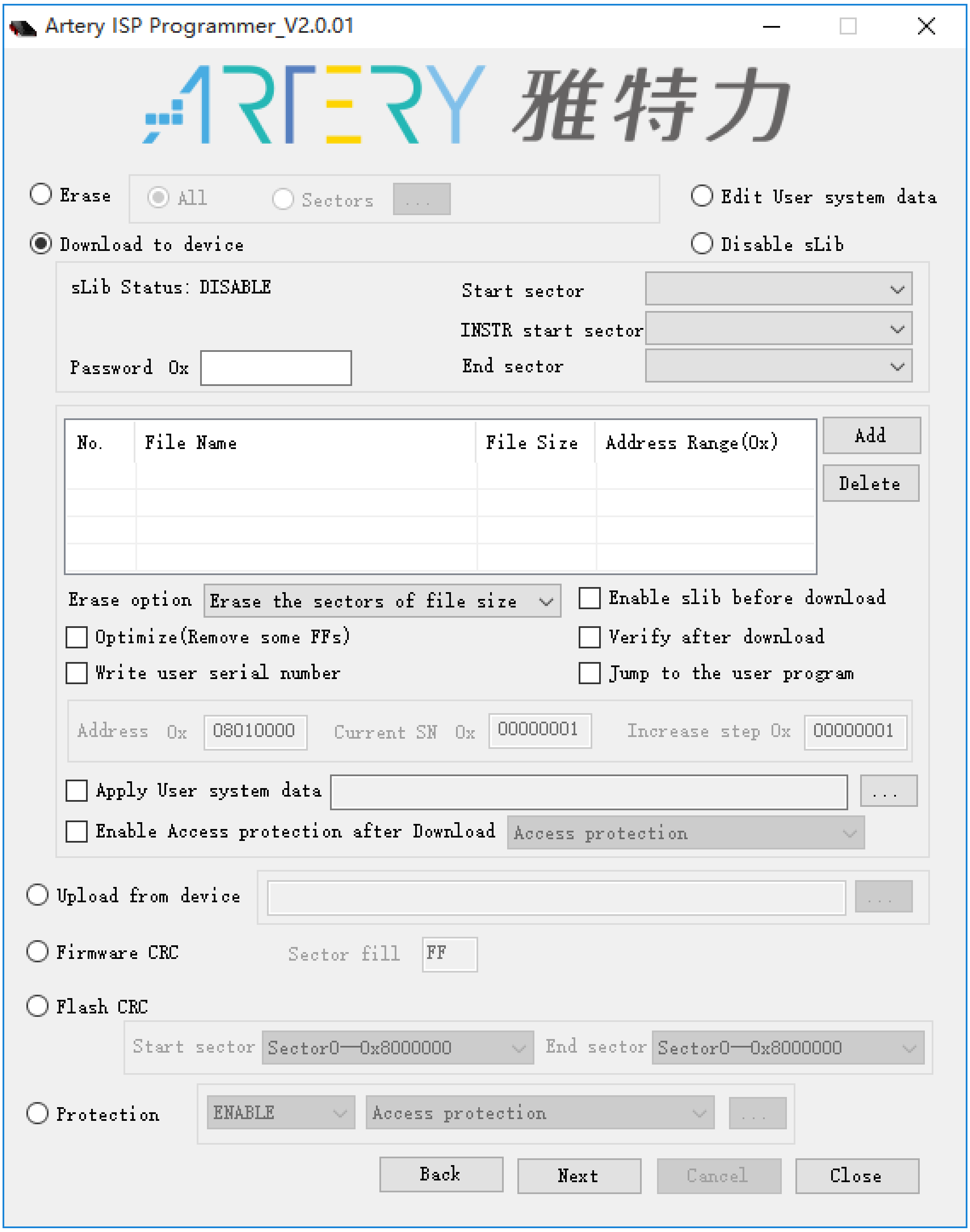
- Remap0 (use PA11/PA12 pins)
- Remap1 (use PB10/PB11 pins)

Select the desired pins. This option is only available for AT32F413/F403/F407 series UART interfaces.

5.4 Operation configuration page

Choose what you need to do on this page. (As shown in Figure 17)

Figure 17. Operation configuration



Artery ISP Programmer_V2.0.01

○ Erase ● All ○ Sectors ...

○ Edit User system data

● Download to device

○ Disable sLib

sLib Status: DISABLE

Start sector

INSTR start sector

End sector

Password 0x

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

Erase option: Erase the sectors of file size

□ Enable slib before download

□ Optimize(Remove some FFs)

□ Verify after download

□ Write user serial number

□ Jump to the user program

Address 0x 08010000 Current SN 0x 00000001 Increase step 0x 00000001

□ Apply User system data ...

□ Enable Access protection after Download: Access protection

○ Upload from device ...

○ Firmware CRC Sector fill FF

○ Flash CRC

Start sector Sector0—0x8000000 End sector Sector0—0x8000000

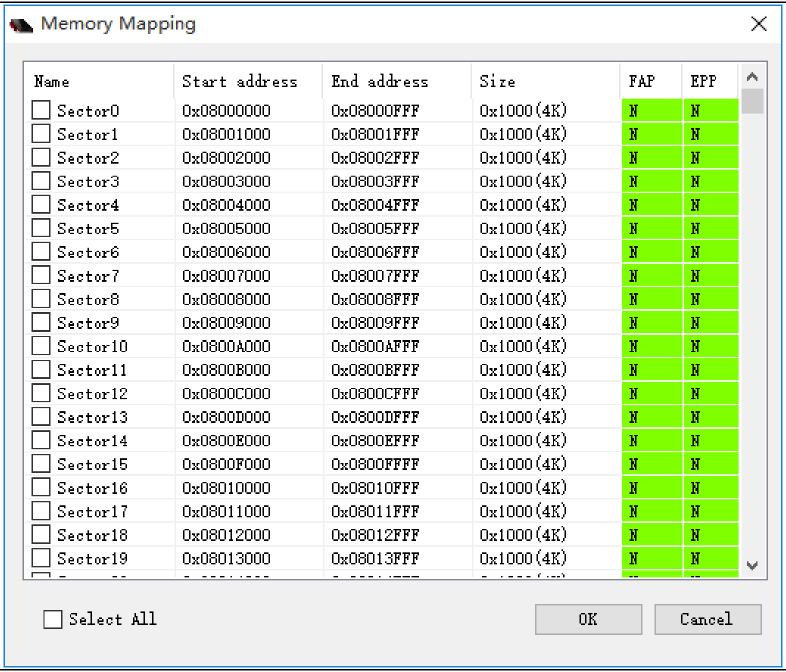
○ Protection ENABLE Access protection ...

5.4.1 Erase

— Click on "***All***" to erase the whole memory (Including SPIM).

— Click on "***Sectors***" to customize the sectors to be erased. At this time, click on "**…**" to select the sector to be erased in the pop-up dialog box. (As shown in Figure 18)

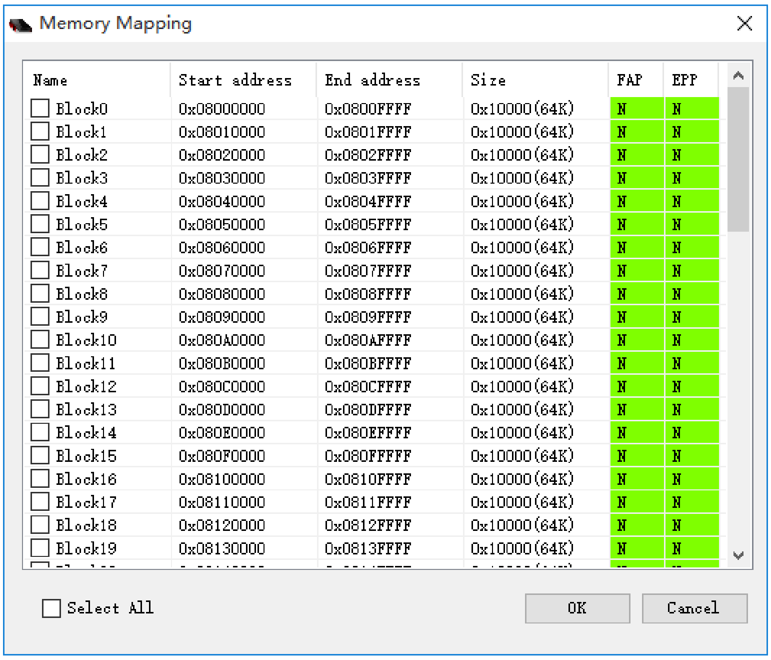
Figure 18. Sector erase selection



Name	Start address	End address	Size	FAP	EPP
□ Sector0	0x08000000	0x08000FFF	0x1000(4K)	N	N
□ Sector1	0x08001000	0x08001FFF	0x1000(4K)	N	N
□ Sector2	0x08002000	0x08002FFF	0x1000(4K)	N	N
□ Sector3	0x08003000	0x08003FFF	0x1000(4K)	N	N
□ Sector4	0x08004000	0x08004FFF	0x1000(4K)	N	N
□ Sector5	0x08005000	0x08005FFF	0x1000(4K)	N	N
□ Sector6	0x08006000	0x08006FFF	0x1000(4K)	N	N
□ Sector7	0x08007000	0x08007FFF	0x1000(4K)	N	N
□ Sector8	0x08008000	0x08008FFF	0x1000(4K)	N	N
□ Sector9	0x08009000	0x08009FFF	0x1000(4K)	N	N
□ Sector10	0x0800A000	0x0800AFFF	0x1000(4K)	N	N
□ Sector11	0x0800B000	0x0800BFFF	0x1000(4K)	N	N
□ Sector12	0x0800C000	0x0800CFFF	0x1000(4K)	N	N
□ Sector13	0x0800D000	0x0800DFFF	0x1000(4K)	N	N
□ Sector14	0x0800E000	0x0800EFFF	0x1000(4K)	N	N
□ Sector15	0x0800F000	0x0800FFFF	0x1000(4K)	N	N
□ Sector16	0x08010000	0x08010FFF	0x1000(4K)	N	N
□ Sector17	0x08011000	0x08011FFF	0x1000(4K)	N	N
□ Sector18	0x08012000	0x08012FFF	0x1000(4K)	N	N
□ Sector19	0x08013000	0x08013FFF	0x1000(4K)	N	N

— Click on "***Blocks***" to customize the blocks to be erased. At this time, click on "**…**" to select the block to be erased in the pop-up dialog box. (As shown in Figure 19)

Figure 19. Block erase selection



Name	Start address	End address	Size	FAP	EPP
□ Block0	0x08000000	0x0800FFFF	0x10000(64K)	N	N
□ Block1	0x08010000	0x0801FFFF	0x10000(64K)	N	N
□ Block2	0x08020000	0x0802FFFF	0x10000(64K)	N	N
□ Block3	0x08030000	0x0803FFFF	0x10000(64K)	N	N
□ Block4	0x08040000	0x0804FFFF	0x10000(64K)	N	N
□ Block5	0x08050000	0x0805FFFF	0x10000(64K)	N	N
□ Block6	0x08060000	0x0806FFFF	0x10000(64K)	N	N
□ Block7	0x08070000	0x0807FFFF	0x10000(64K)	N	N
□ Block8	0x08080000	0x0808FFFF	0x10000(64K)	N	N
□ Block9	0x08090000	0x0809FFFF	0x10000(64K)	N	N
□ Block10	0x080A0000	0x080AFFFF	0x10000(64K)	N	N
□ Block11	0x080B0000	0x080BFFFF	0x10000(64K)	N	N
□ Block12	0x080C0000	0x080CFFFF	0x10000(64K)	N	N
□ Block13	0x080D0000	0x080DFFFF	0x10000(64K)	N	N
□ Block14	0x080E0000	0x080EFFFF	0x10000(64K)	N	N
□ Block15	0x080F0000	0x080FFFFF	0x10000(64K)	N	N
□ Block16	0x08100000	0x0810FFFF	0x10000(64K)	N	N
□ Block17	0x08110000	0x0811FFFF	0x10000(64K)	N	N
□ Block18	0x08120000	0x0812FFFF	0x10000(64K)	N	N
□ Block19	0x08130000	0x0813FFFF	0x10000(64K)	N	N

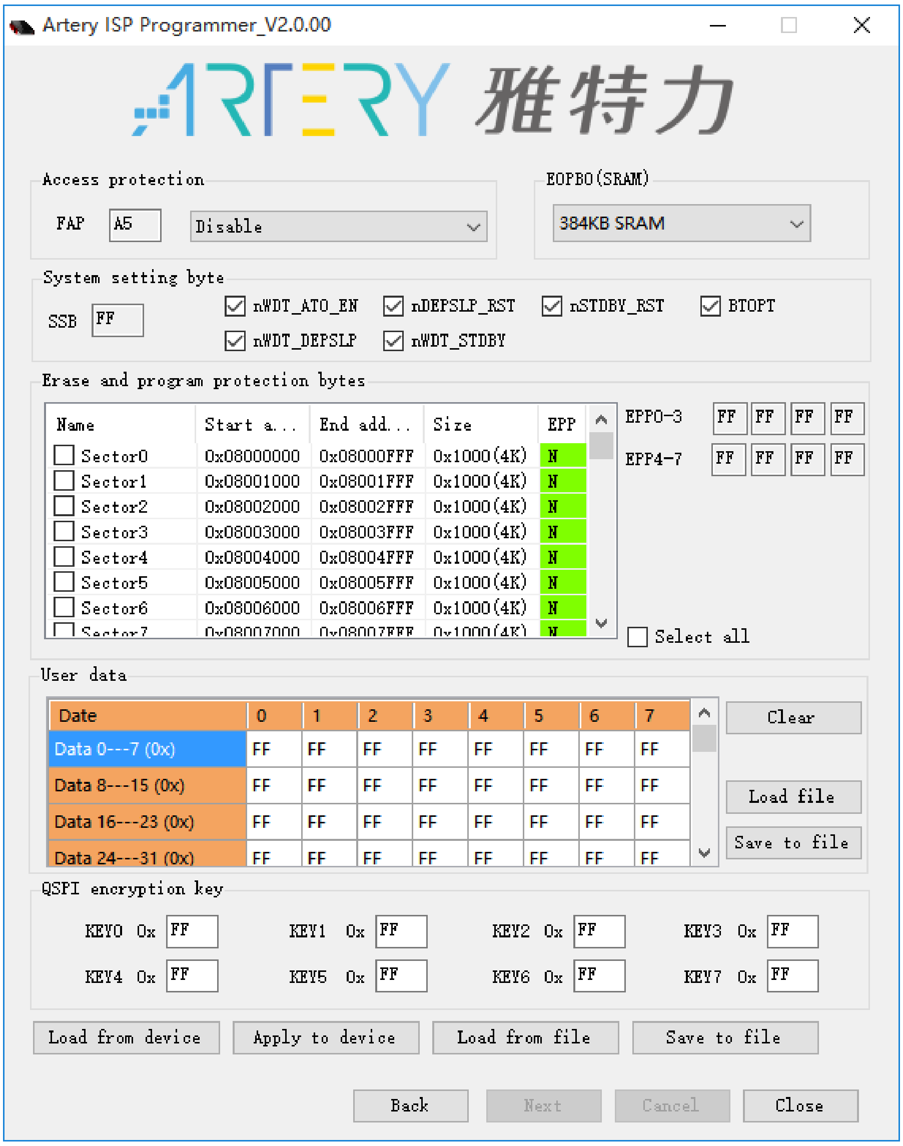
5.4.2 Edit User system data

Select "***Edit User system data***" and click on "***Next***".

On this page, users can configure the "User system data" through graphical interface (As shown in Figure 20).

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.

Figure 20. User system data



Access protection

FAP A5 Disable

EOPB0(SRAM)

384KB SRAM

System setting byte

SSB FF

☑ nWDT_ATO_EN ☑ nDEPSLP_RST ☑ nSTDBY_RST ☑ BTOPT

☑ nWDT_DEPSLP ☑ nWDT_STDBY

Erase and program protection bytes

Name	Start a...	End add...	Size	EPP
□ Sector0	0x08000000	0x08000FFF	0x1000(4K)	N
□ Sector1	0x08001000	0x08001FFF	0x1000(4K)	N
□ Sector2	0x08002000	0x08002FFF	0x1000(4K)	N
□ Sector3	0x08003000	0x08003FFF	0x1000(4K)	N
□ Sector4	0x08004000	0x08004FFF	0x1000(4K)	N
□ Sector5	0x08005000	0x08005FFF	0x1000(4K)	N
□ Sector6	0x08006000	0x08006FFF	0x1000(4K)	N
□ Sector7	0x08007000	0x08007FFF	0x1000(4K)	N

EPP0-3 FF FF FF FF

EPP4-7 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
Data 8---15 (0x)	FF	FF	FF	FF	FF	FF	FF	FF
Data 16---23 (0x)	FF	FF	FF	FF	FF	FF	FF	FF
Data 24---31 (0x)	FF	FF	FF	FF	FF	FF	FF	FF

QSPI 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

- Access protection

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

AT32F403/F413/F403A/F407/F435/F437:

Enabled: FAP----0xFF.

Disabled: FAP---0xA5.

AT32F415/F421/F425/L021:

Access protection: FAP----0xFF.

High level access protection: FAP----0xCC (Access protection and user system data erase

protection). (AT32F425/L021 high level access protection is irreversible. Once enabled, it will never be unlocked, with its debugging interface permanently disabled. Please use with caution.)
Disabled: FAP----0xA5.

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

- System setting byte

nWDT_ATO_EN:
Unchecked—Hardware watchdog.
Checked—Software watchdog.

nDEPSLP_RST:
Unchecked—Reset occurs when entering Deep Sleep mode.
Checked—No reset occurs when entering Deep Sleep mode.

nSTDBY_RST:
Unchecked—Reset occurs when entering Standby mode.
Checked—No reset occurs when entering Standby mode.

BTOPT (AT32F403/F413/F403A/F407/F435/F437)
Unchecked—when the device is set to boot from flash memory bank 1 or bank 2, if bank 2 has no startup program, boots from bank 1, otherwise, bank 2.
Checked—when the device is set to boot from flash memory (default value), it starts from bank 1.

nBOOT1 (AT32F421/F425/L021)
Boot mode is determined together with BOOT0, and when BOOT0 = 1,
Unchecked----SRAM is selected as boot space.
Checked---Boot memory is selected as boot space.

nWDT_DEPSLP:
Unchecked----WDT stop count when entering Deep Sleep mode.
Checked---WDT does not stop count when entering Deep Sleep mode.

nWDT_STDBY:
Unchecked---- WDT stop count when entering Standby mode.
Checked--- WDT does not stop count when entering Standby mode.

SRAM_Parity: （AT32L021）
Unchecked ----- Enable odd check of RAM.
Checked ------- Disable odd check of RAM.

- EOPB0(SRAM)

AT32F403/F403A/F407: (AT32F403CBT6 not support)
224 KB SRAM—SRAM 224 KB.

96 KB SRAM—SRAM 96 KB.

AT32F413: (AT32F413C8T7/AT32FEBKC8T7 not support)
64 KB SRAM—SRAM 64 KB.
32 KB SRAM—SRAM 32 KB.
16 KB SRAM—SRAM 16 KB.

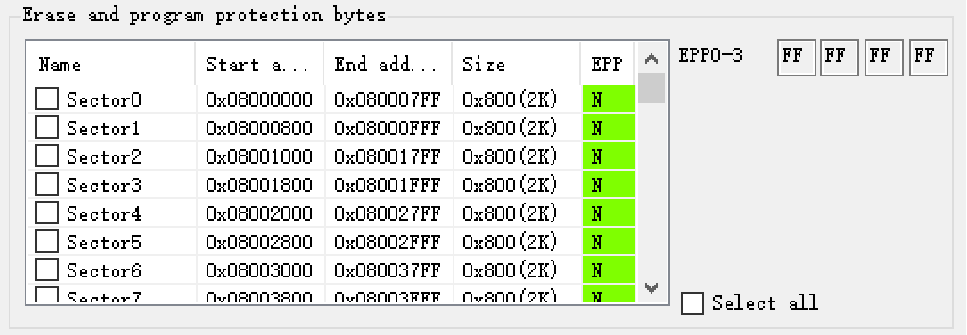
AT32F415/F421/F425/L021: (not support)

AT32F435/F437:
Flash size 256K and below:
512 KB SRAM—SRAM 512 KB.
448 KB SRAM—SRAM 448 KB.
384 KB SRAM—SRAM 384 KB.
Flash size 1024K and above:
512 KB SRAM—SRAM 512 KB.
448 KB SRAM—SRAM 448 KB.
384 KB SRAM—SRAM 384 KB.
320 KB SRAM—SRAM 320 KB.
256 KB SRAM—SRAM 256 KB.
192 KB SRAM—SRAM 192 KB.
128 KB SRAM—SRAM 128 KB.

- Erase and program protection bytes

You can choose which sectors need to be erase and program protected. (As shown in Figure 21)

Figure 21. Erase and program protection bytes



Erase and program protection bytes

Name	Start a...	End add...	Size	EPP
□ Sector0	0x08000000	0x080007FF	0x800(2K)	N
□ Sector1	0x08000800	0x08000FFF	0x800(2K)	N
□ Sector2	0x08001000	0x080017FF	0x800(2K)	N
□ Sector3	0x08001800	0x08001FFF	0x800(2K)	N
□ Sector4	0x08002000	0x080027FF	0x800(2K)	N
□ Sector5	0x08002800	0x08002FFF	0x800(2K)	N
□ Sector6	0x08003000	0x080037FF	0x800(2K)	N
□ Sector7	0x08003800	0x08003FFF	0x800(2K)	N

EPP0-3 FF FF FF FF

□ Select all

EPP0:

AT32F403/F413/F403A/F407: controls the erase and program protection of sectors in the range of Flash 1K-32K.

AT32F415: controls the erase and program protection of Sector0-Sector15.

AT32F421: controls the erase and program protection of Sector0-Sector31.

AT32F435/F437: controls the erase and program protection of sectors in the range of Flash 1K-32K. Each bit protects 4K bytes sectors.

AT32F425: controls the erase and program protection of Sector0-Sector31.

AT32L021: controls the erase and program protection of Sector0-Sector31.

EPP1:

AT32F403/F413/F403A/F407: controls the erase and program protection of sectors in the range of Flash 33K-64K.
AT32F415: controls the erase and program protection of Sector16-Sector31.
AT32F421: controls the erase and program protection of Sector32-Sector63.
AT32F435/F437: controls the erase and program protection of sectors in the range of Flash 33K-64K. Each bit protects 4K bytes sectors.
AT32F425: controls the erase and program protection of Sector32-Sector63.
AT32L021: controls the erase and program protection of Sector32-Sector63.

EPP2:
AT32F403/F413/F403A/F407: controls the erase and program protection of sectors in the range of Flash 65K-96K.
AT32F415: controls the erase and program protection of Sector32-Sector47.
AT32F435/F437: controls the erase and program protection of sectors in the range of Flash 65K-96K. Each bit protects 4K bytes sectors.

EPP3:
AT32F403/F413/F403A/F407:
Bit 0-6 controls the erase and program protection of sectors in the range of 97K-124K;
Bit 7 controls the erase and program protection of all Sectors after Flash 124K, including SPIM.
AT32F415:
Bits 0-6 control the erase and program protection of Sector48-Sector61;
Bit 7 controls the erase and program protection of all subsequent sectors, including boot memory (boot memory in AP mode).
AT32F421: Bit 7 controls the boot memory area (boot memory in AP mode)
AT32F435/F437: controls the erase and program protection of sectors in the range of Flash 97K-128K. Each bit protects 4K bytes sectors.
AT32F425: Bit 7 controls the boot memory area (boot memory in AP mode)
AT32L021: Bit 7 controls the boot memory area (boot memory in AP mode)

EPP4:
AT32F435/F437: controls the erase and program protection of sectors in the range of Flash 129K-1152K. Each bit protects 128K bytes sectors.

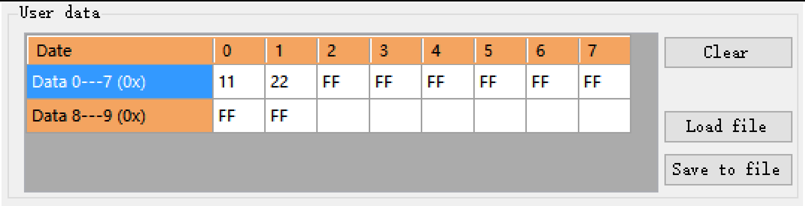
EPP5:
AT32F435/F437: controls the erase and program protection of sectors in the range of Flash 1153K-2176K. Each bit protects 128K bytes sectors.

EPP6:
AT32F435/F437: controls the erase and program protection of sectors in the range of Flash 2177K-3200K. Each bit protects 128K bytes sectors.

EPP7:
AT32F435/F437: Bit 0-6 controls the erase and program protection of sectors in the range of Flash 3201K-4032K. Each bit protects 128K bytes sectors.

- User data

Figure 22. User data



Date	0	1	2	3	4	5	6	7
Data 0---7 (0x)	11	22	FF	FF	FF	FF	FF	FF
Data 8---9 (0x)	FF	FF						

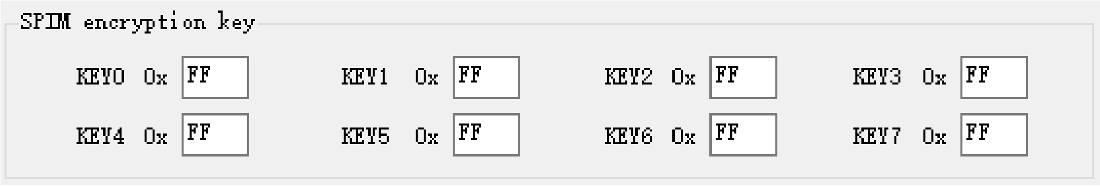
AT32F403/F413/F403A/F407: user data 8 bytes.
AT32F415: user data 10 bytes.
AT32F421: user data 250 bytes.
AT32F435/F437: Flash size is less than 4032K, user data 220 bytes. Flash size 4032K, user data 2012 bytes.
AT32L021: user data 250 bytes.
AT32F425: user data 250 bytes.

Clear: Reset all user system data to 0xFF, which is not saved to the device
Load file: Load the user system data file into the table for display
Save to file: Save the user system data in the table to the file.

- SPIM encryption key (AT32F403/F413/F403A/F407)

You can set the encryption key when downloading the SPIM. (As shown in Figure 23)

Figure 23. SPIM encryption key

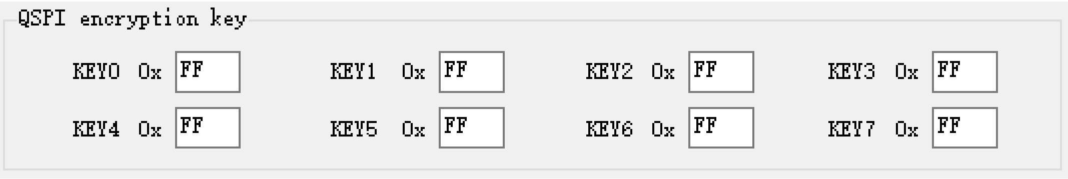


KEY0 0x FF KEY1 0x FF KEY2 0x FF KEY3 0x FF
KEY4 0x FF KEY5 0x FF KEY6 0x FF KEY7 0x FF

- QSPI encryption key (AT32F435/F437)

You can set the encryption key when downloading the QSPI. (As shown in Figure 24)

Figure 24. QSPI 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

- Load from device

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

- Apply to device

Save the settings of the user system data to the device.

- Load from file

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

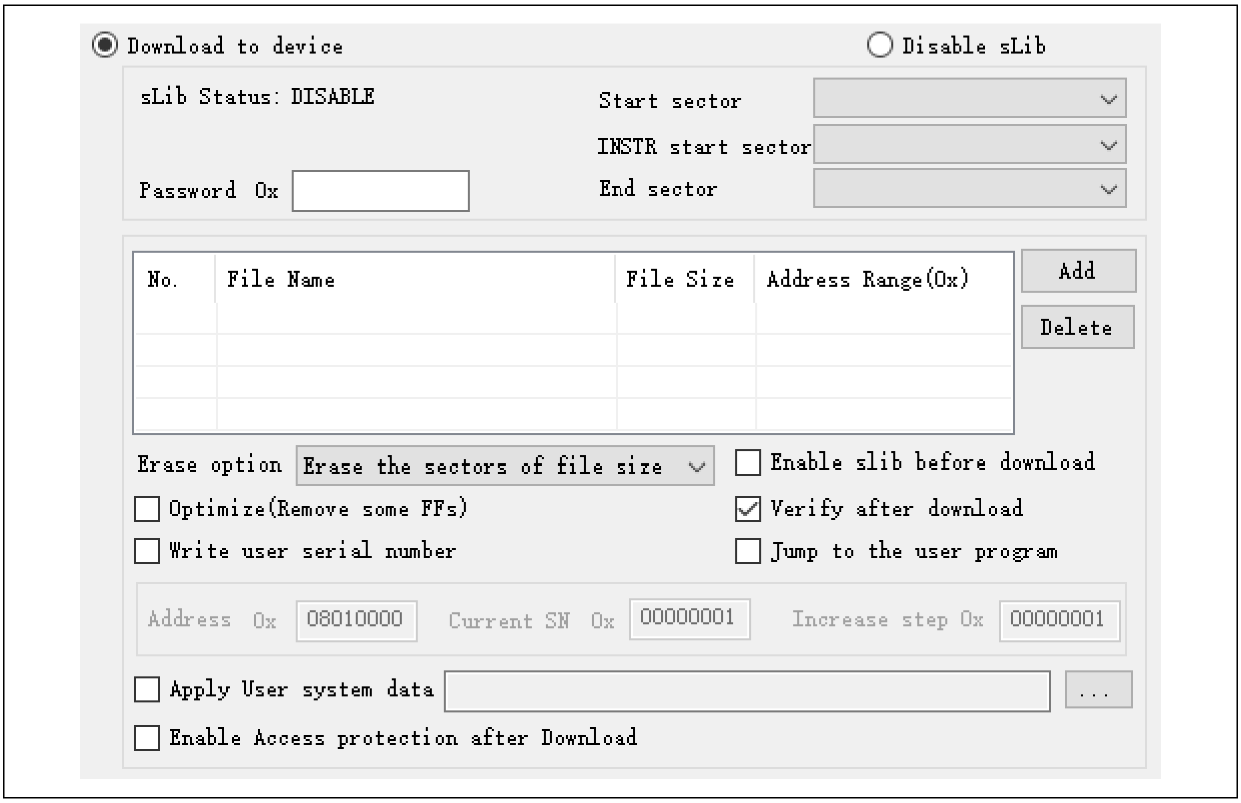
- Save to file

Save the user system data settings to a file.

5.4.3 Download to device

(As show in Figure 25)

Figure 25. Download to device



● Download to device ○ Disable sLib

sLib Status: DISABLE

Start sector

INSTR start sector

Password 0x

End sector

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

Erase option: Erase the sectors of file size

☐ Enable slib before download

☐ Optimize(Remove some FFs)

☑ Verify after download

☐ Write user serial number

☐ Jump to the user program

Address 0x 08010000 Current SN 0x 00000001 Increase step 0x 00000001

☐ Apply User system data

☐ Enable Access protection after Download

- **sLib settings**

（AT32F403 not support sLib）

— sLib status

The sLib status of the current connected chip, disabled or enabled.

— Remaining usage times (AT32F413/F403A/F407)

It means the remaining number of times of sLib. It can be used up to 256 times, and will be reduced after each use. When the remaining number of 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

AT32F413/F415/F403A/F407:

The start sector of sLib area. The instruction area is from the "Start sector" to the "DATA start sector"(not including The DATA start sector). When sLib is enabled, the data in this area cannot be erased, written or read.

AT32F421/F435/F437/F425/L021:

The start sector of sLib area.The area from “Start sector” to “INSTR start sector” (not including

“INSTR start sector”) is a mixed instruction and data (read only area). Once sLib is enabled, the data in this area cannot be erased, written, but can be read.

— DATA start Sector/INSTR start Sector

AT32F413/F415/F403A/F407：

The start sector of the sLib data area. This data area is from "DATA start sector" to "End sector"(including "End sector"). After sLib is enabled, the data in this area cannot be erased and written, but can be read. When set to "none", it is set to no data area.

AT32F421/F435/F437/F425/L021:

The start sector of sLib instruction area. The instruction area is from "INSTR start sector" to "End sector" (including "End sector"). After sLib is enabled, the data in this area cannot be erased, written or read. When it is set to "none", it is no instruction area.

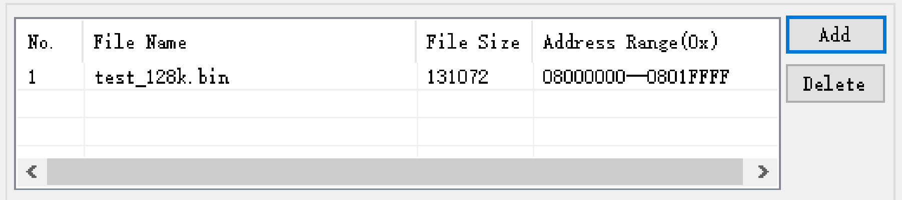
— End Sector

The end position of the sLib area.

- **Other download settings**

Three file types are supported: bin (binary), hex (hexadecimal), and s19 / srec (Motorola S file). (As shown in Figure 26)

Figure 26. Download file selection



No.	File Name	File Size	Address Range(0x)
1	test_128k.bin	131072	08000000—0801FFFF

If you are adding a bin file, you need to choose a download address.

If you are adding a hex or S19 / SREC file, the download address is obtained from the loaded file.

— Check "***Erase the sectors of file size***" to erase sectors where the downloaded file is located before download.

— Check "***No Erase***", no erase operation will be performed before download.

— Check "***Global Erase***" to erase the whole memory (including SPIM) before download.

— Check "***Jump to the user program***" to run the program directly after the download is complete.

— Check "***Enable sLib before download***" to enable sLib before download. You need to enter the password, start sector, DATA/INSTR start sector, and end sector to enable sLib.

— Check "***Verify after download"*** to run the verify program after downloading to verify whether the downloaded data is correct.

— 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 of the current programming.

Increase step: this is the amount added to the next serial number after each serial number is programmed

— Check "***Apply User system data***", load the user system data file after download, and set the value to the device.

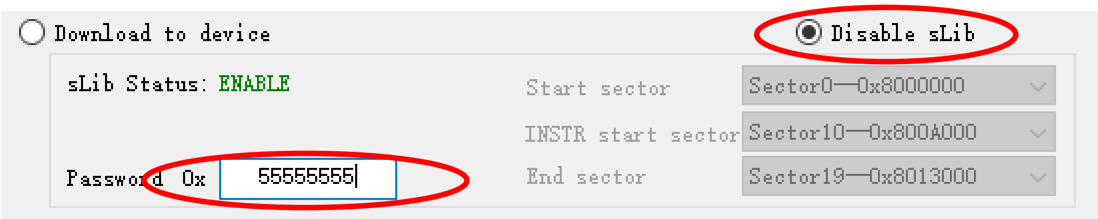
— Check "***Enable Access Protection after Download***" to enable access protection after download.

For AT32F415/F421/F425/L021, you can enable access protection and high level access protection (Access protection and user system data erase protection). (AT32F425/L021 high level access protection is irreversible. Once enabled, it will never be unlocked, with its debugging interface permanently disabled. Please use with caution.)

5.4.4 Disable sLib

To disable sLib, enter the disable password. (That is, enter the password when sLib was last enabled) (As shown in Figure 27):

Figure 27. Disable sLib



○ Download to device

● Disable sLib

sLib Status: ENABLE

Password 0x 55555555

Start sector: Sector0—0x8000000

INSTR start sector: Sector10—0x800A000

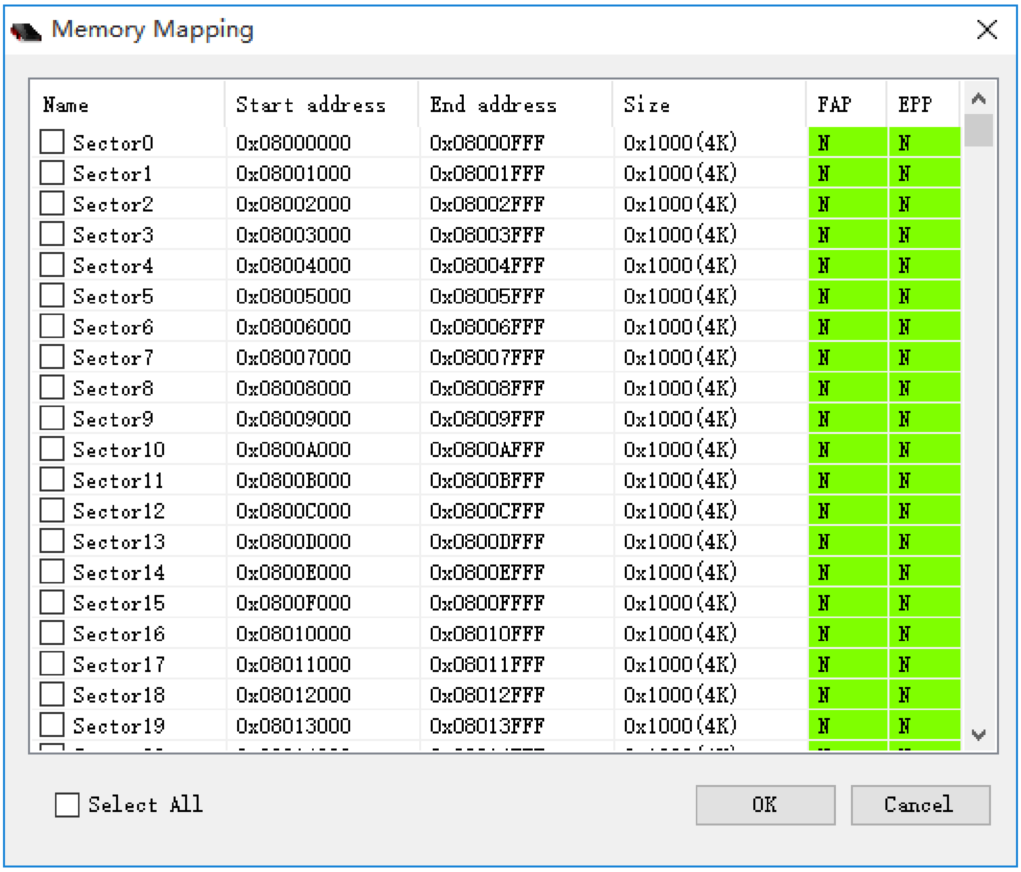
End sector: Sector19—0x8013000

When disabling sLib successfully, the whole chip will be erased.

5.4.5 Upload from device

Three file types are supported: bin (binary), hex (hexadecimal), and s19 / srec (Motorola S file). Select the upload sectors. (As shown in Figure 28)

Figure 28. Upload from device



Memory Mapping

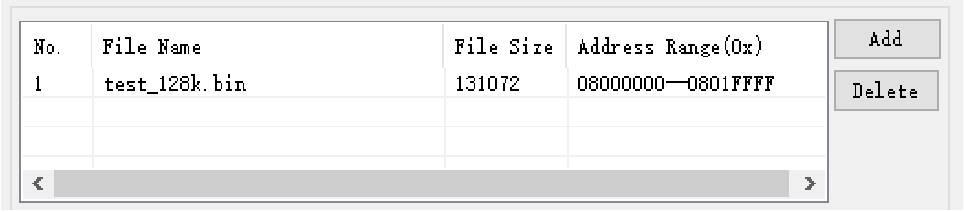
Name	Start address	End address	Size	FAP	EPP
□ Sector0	0x08000000	0x08000FFF	0x1000(4K)	N	N
□ Sector1	0x08001000	0x08001FFF	0x1000(4K)	N	N
□ Sector2	0x08002000	0x08002FFF	0x1000(4K)	N	N
□ Sector3	0x08003000	0x08003FFF	0x1000(4K)	N	N
□ Sector4	0x08004000	0x08004FFF	0x1000(4K)	N	N
□ Sector5	0x08005000	0x08005FFF	0x1000(4K)	N	N
□ Sector6	0x08006000	0x08006FFF	0x1000(4K)	N	N
□ Sector7	0x08007000	0x08007FFF	0x1000(4K)	N	N
□ Sector8	0x08008000	0x08008FFF	0x1000(4K)	N	N
□ Sector9	0x08009000	0x08009FFF	0x1000(4K)	N	N
□ Sector10	0x0800A000	0x0800AFFF	0x1000(4K)	N	N
□ Sector11	0x0800B000	0x0800BFFF	0x1000(4K)	N	N
□ Sector12	0x0800C000	0x0800CFFF	0x1000(4K)	N	N
□ Sector13	0x0800D000	0x0800DFFF	0x1000(4K)	N	N
□ Sector14	0x0800E000	0x0800EFFF	0x1000(4K)	N	N
□ Sector15	0x0800F000	0x0800FFFF	0x1000(4K)	N	N
□ Sector16	0x08010000	0x08010FFF	0x1000(4K)	N	N
□ Sector17	0x08011000	0x08011FFF	0x1000(4K)	N	N
□ Sector18	0x08012000	0x08012FFF	0x1000(4K)	N	N
□ Sector19	0x08013000	0x08013FFF	0x1000(4K)	N	N

5.4.6 Firmware CRC

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

First you need to select the file to be compared. (As shown in Figure 29)

Figure 29. Firmware CRC



No.	File Name	File Size	Address Range(0x)
1	test_128k.bin	131072	08000000—0801FFFF

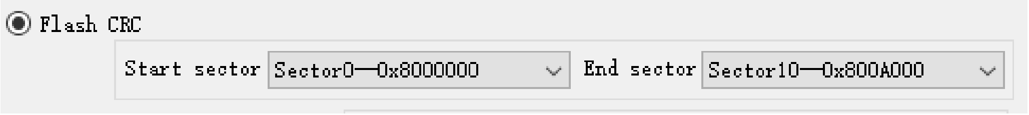
"Sector fill": the Firmware CRC is performed in units of sectors. What is filled in here is the download data that is not filled in the sector part. Generally, it is "FF".

5.4.7 Flash CRC

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

(As shown in Figure 30):

Figure 30. Flash CRC



● Flash CRC

Start sector: Sector0—0x8000000

End sector: Sector10—0x800A000

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

5.4.8 Protection

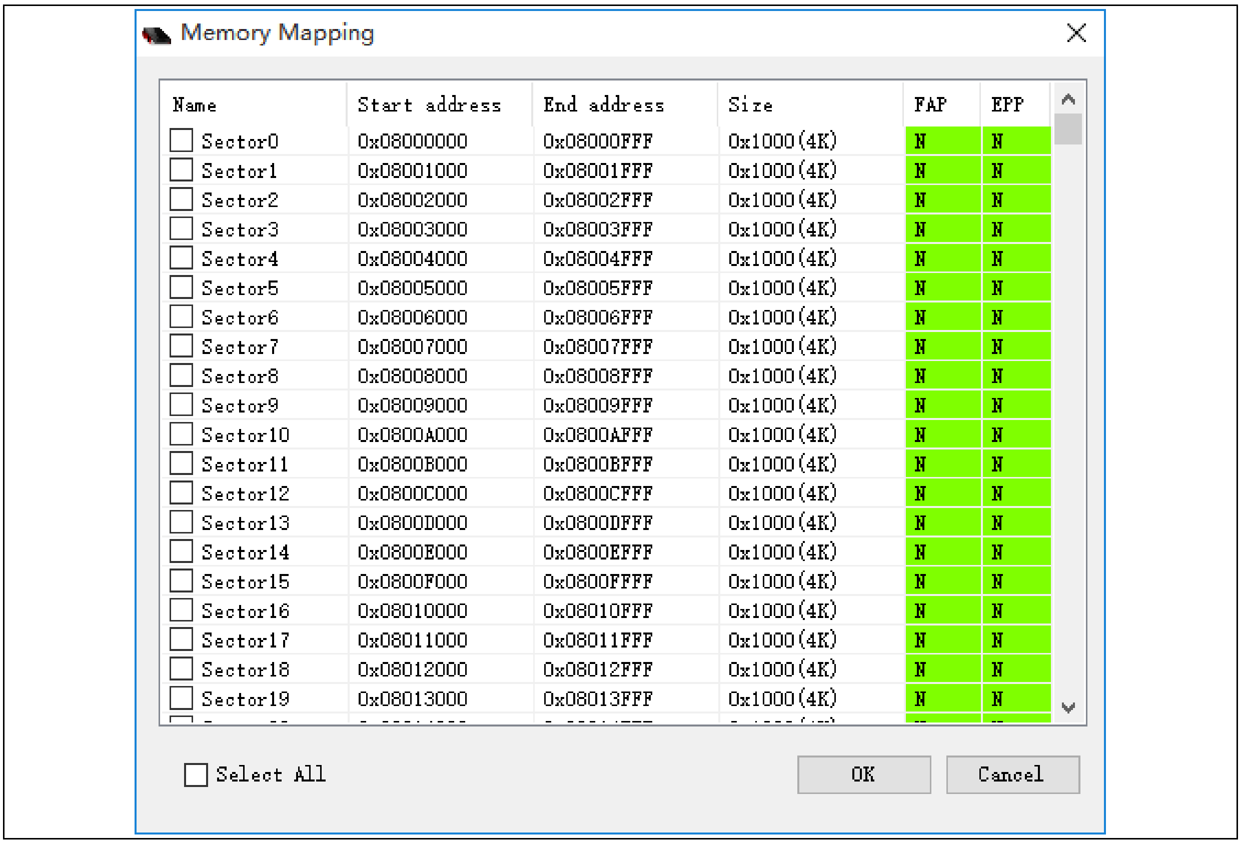
— Select "***Enable***" - "***Access Protection***" to enable the flash access protection. The whole flash will be access protected.

AT32F415/F421/F425/L021: enable access protection and high level access protection (Access protection and user system data erase protection). (AT32F425/L021 high level access protection is irreversible. Once enabled, it will never be unlocked, with its debugging interface permanently disabled. Please use with caution.)

— Select "***Disable***" - "***Access Protection***" to disable the access protection of the whole flash.

— Select "***Enable***" - ***"Erase and program protection"***, and click **"..."**, you can select the sectors to enable erase and program protection in the dialog box that pops up. (As shown in Figure 31)

Figure 31. Enable erase and program protection



Name	Start address	End address	Size	FAP	EPP
□ Sector0	0x08000000	0x08000FFF	0x1000(4K)	N	N
□ Sector1	0x08001000	0x08001FFF	0x1000(4K)	N	N
□ Sector2	0x08002000	0x08002FFF	0x1000(4K)	N	N
□ Sector3	0x08003000	0x08003FFF	0x1000(4K)	N	N
□ Sector4	0x08004000	0x08004FFF	0x1000(4K)	N	N
□ Sector5	0x08005000	0x08005FFF	0x1000(4K)	N	N
□ Sector6	0x08006000	0x08006FFF	0x1000(4K)	N	N
□ Sector7	0x08007000	0x08007FFF	0x1000(4K)	N	N
□ Sector8	0x08008000	0x08008FFF	0x1000(4K)	N	N
□ Sector9	0x08009000	0x08009FFF	0x1000(4K)	N	N
□ Sector10	0x0800A000	0x0800AFFF	0x1000(4K)	N	N
□ Sector11	0x0800B000	0x0800BFFF	0x1000(4K)	N	N
□ Sector12	0x0800C000	0x0800CFFF	0x1000(4K)	N	N
□ Sector13	0x0800D000	0x0800DFFF	0x1000(4K)	N	N
□ Sector14	0x0800E000	0x0800EFFF	0x1000(4K)	N	N
□ Sector15	0x0800F000	0x0800FFFF	0x1000(4K)	N	N
□ Sector16	0x08010000	0x08010FFF	0x1000(4K)	N	N
□ Sector17	0x08011000	0x08011FFF	0x1000(4K)	N	N
□ Sector18	0x08012000	0x08012FFF	0x1000(4K)	N	N
□ Sector19	0x08013000	0x08013FFF	0x1000(4K)	N	N

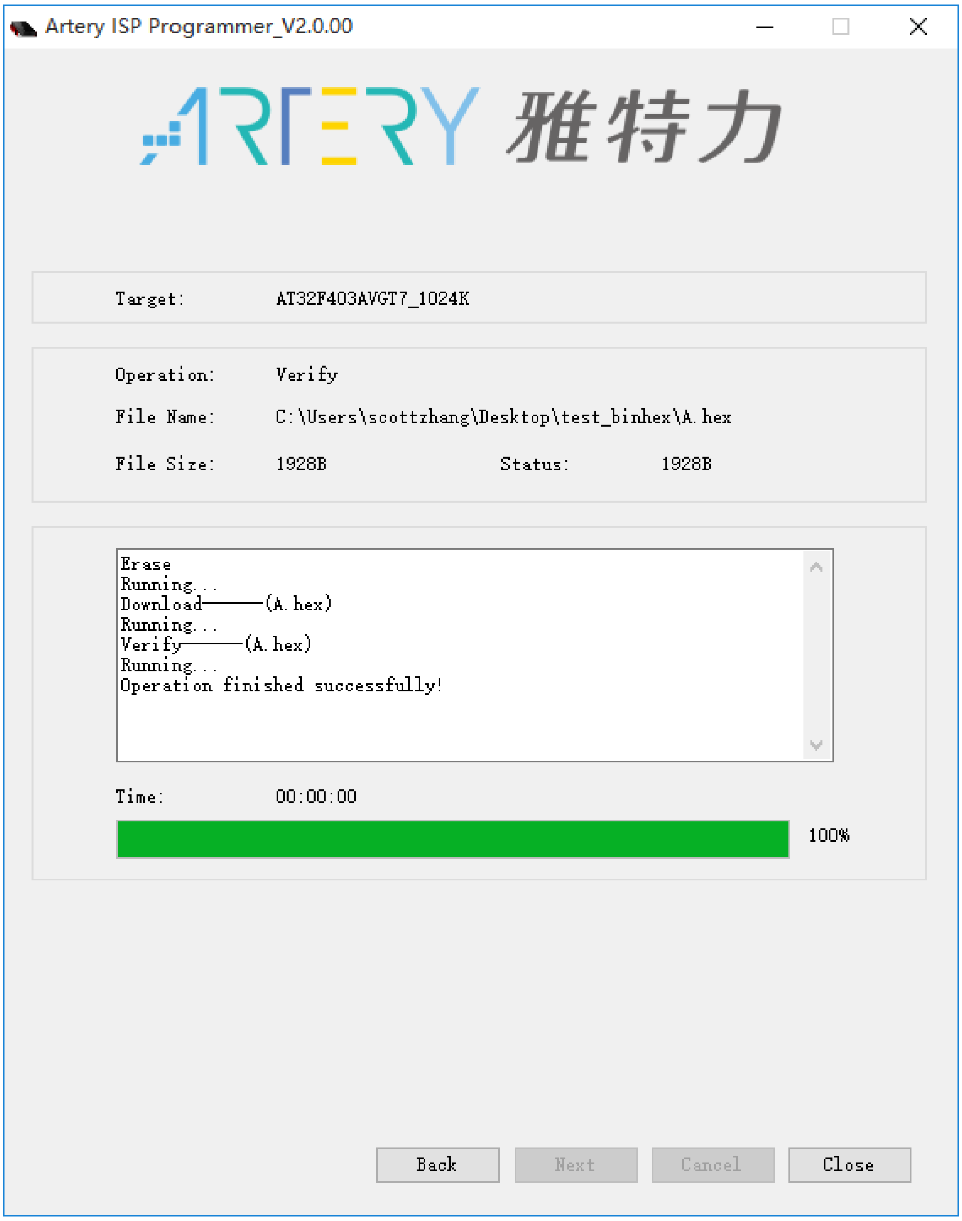
□ Select All

— Select "***Disable***" - "***Erase and program protection***" to disable the erase and program protection of the whole flash.

5.5 Operation progress page

This page displays information related to the operation progress. (As shown in Figure 32)

Figure 32. Operation progress display



Artery ISP Programmer_V2.0.00

Target: AT32F403AVGT7_1024K

Operation: Verify

File Name: C:\Users\scottzhang\Desktop\test_binhex\A.hex

File Size: 1928B Status: 1928B

```
Erase
Running...
Download———(A.hex)
Running...
Verify———(A.hex)
Running...
Operation finished successfully!
```

Time: 00:00:00

100%

5.6 SPIM encryption download

SPIM encryption principle:

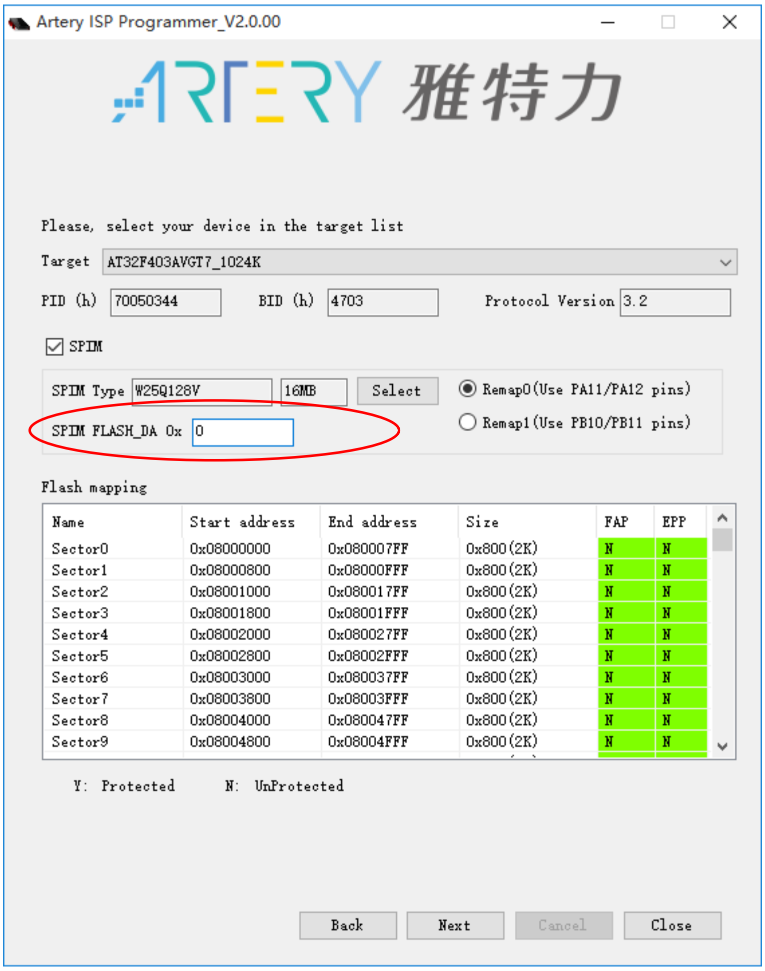
When SPIM encrypted download is required, users must first configure the SPIM FLASH_DA and SPIM encryption key (Key is set in the user system data), and then perform download operation. In this case, the MCU will encrypt the downloaded original data according to SPIM FLASH_DA and encryption key as well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

When users want to read the encrypted data in the SPIM, users also need to configure the SPIM FLASH_DA and encryption key. Based on the SPIM FLASH_DA and encryption key, the MCU uses the MCU's internal algorithm to decrypt the encrypted data and restore it to the correct original data.

When downloading files to SPIM, the following steps can be set to encrypt the downloaded contents (AT32F403/F413/F403A/F407 support SPIM)

Step 1: set the SPIM FLASH_DA (As shown in Figure 33).

Figure 33. Encryption range config



Please, select your device in the target list

Target: AT32F403AVGT7_1024K

PID (h): 70050344　BID (h): 4703　Protocol Version: 3.2

☑ SPIM

SPIM Type: W25Q128V　16MB　Select

● Remap0(Use PA11/PA12 pins)

○ Remap1(Use PB10/PB11 pins)

SPIM FLASH_DA 0x: 0

Flash mapping

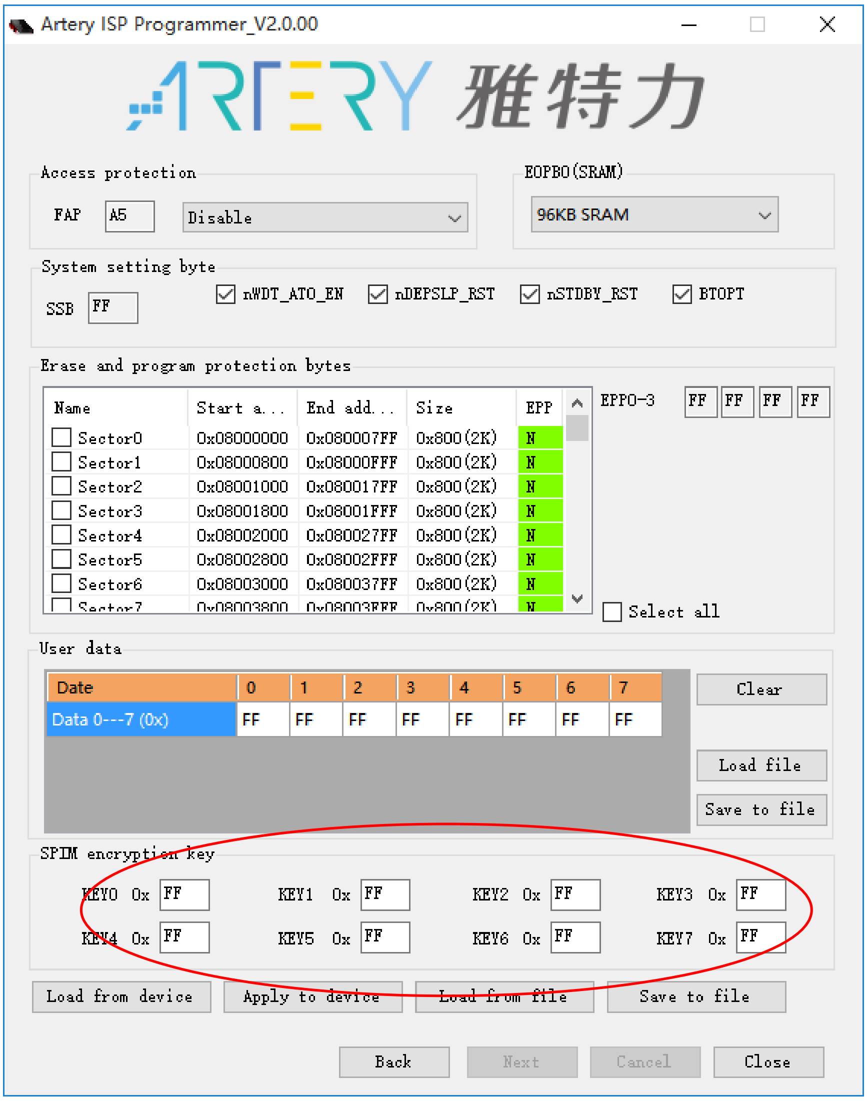
Name	Start address	End address	Size	FAP	EPP
Sector0	0x08000000	0x080007FF	0x800(2K)	N	N
Sector1	0x08000800	0x08000FFF	0x800(2K)	N	N
Sector2	0x08001000	0x080017FF	0x800(2K)	N	N
Sector3	0x08001800	0x08001FFF	0x800(2K)	N	N
Sector4	0x08002000	0x080027FF	0x800(2K)	N	N
Sector5	0x08002800	0x08002FFF	0x800(2K)	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

Starting from the address 0x08400000, plus the set FLASH_DA, it is the encryption area.
If encryption is not required, set to 0.

Step 2: set the SPIM encryption key through the "User system data". (As shown in Figure 34)

Figure 34. SPIM encryption key config



Artery ISP Programmer_V2.0.00

Access protection

FAP A5 Disable

EOPB0(SRAM)

96KB SRAM

System setting byte

SSB FF ☑ nWDT_ATO_EN ☑ nDEPSLP_RST ☑ nSTDBY_RST ☑ BTOPT

Erase and program protection bytes

Name	Start a...	End add...	Size	EPP
□ Sector0	0x08000000	0x080007FF	0x800(2K)	N
□ Sector1	0x08000800	0x08000FFF	0x800(2K)	N
□ Sector2	0x08001000	0x080017FF	0x800(2K)	N
□ Sector3	0x08001800	0x08001FFF	0x800(2K)	N
□ Sector4	0x08002000	0x080027FF	0x800(2K)	N
□ Sector5	0x08002800	0x08002FFF	0x800(2K)	N
□ Sector6	0x08003000	0x080037FF	0x800(2K)	N
□ Sector7	0x08003800	0x08003FFF	0x800(2K)	N

EPP0-3 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

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

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 is also erased.

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

6 Revision history

Table 19. Document revision history

Date	Revision	Changes
2022/08/25	V2.06	1. Support for AT32F4212C8T7.
2022/07/06	V2.04	1. Support for AT32L021 serial.
2022/01/26	V2.02	1. The serial port number supports a maximum of 1024.
2021/11/23	V2.01	1. Support for AT32F425 serial. 2. Support for AT32F403AVGW. 3. Support for AT32WB415 serial.
2021/10/09	V2.00	1. Initial release. Support for AT32F403/F413/F415/F421/F403A/F407/F435/F437.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

Purchasers understand and agree that purchasers are solely responsible for the selection and use of Artery's products and services.

Artery's products and services are provided "AS IS" and Artery provides no warranties express, implied or statutory, including, without limitation, any implied warranties of merchantability, satisfactory quality, non-infringement, or fitness for a particular purpose with respect to the Artery's products and services.

Notwithstanding anything to the contrary, purchasers acquires no right, title or interest in any Artery's products and services or any intellectual property rights embodied therein. In no event shall Artery's products and services provided be construed as (a) granting purchasers, expressly or by implication, estoppel or otherwise, a license to use third party's products and services; or (b) licensing the third parties' intellectual property rights; or (c) warranting the third party's products and services and its intellectual property rights.

Purchasers hereby agrees that Artery's products are not authorized for use as, and purchasers shall not integrate, promote, sell or otherwise transfer any Artery's product to any customer or end user for use as critical components in (a) any medical, life saving or life support device or system, or (b) any safety device or system in any automotive application and mechanism (including but not limited to automotive brake or airbag systems), or (c) any nuclear facilities, or (d) any air traffic control device, application or system, or (e) any weapons device, application or system, or (f) any other device, application or system where it is reasonably foreseeable that failure of the Artery's products as used in such device, application or system would lead to death, bodily injury or catastrophic property damage.