PM0006
User Manual

AT32 Bootloader USB DFU Protocol

Introduction

This guideline describes how to use USB DFU protocol in the AT32 microcontroller bootloader, including the supported commands..

Applicable products:

MCUs	AT32F403xx
	AT32F413xx
	AT32F415xx
	AT32F403Axx
	AT32F407xx
	AT32F435xx
	AT32F437xx

Contents

List of tables

List of figures

1 USB DFU Bootloader requests

AT32 Bootloader supports USB DFU protocols and requests. Refer to *Universal Serial Bus Device Upgrade Specification for Device Firmware Upgrade Version 1.1* for more information.

Table 1 DFU requests

Request	Code	Description
DFU_DETACH	0x00	Device exits DFU mode (This function is not performed)
DFU_DNLOAD	0x01	Download data from host to device memory
DFU_UPLOAD	0x02	Upload data from device memory to host
DFU_GETSTATUS	0x03	Report device status to host
DFU_CLRSTATUS	0x04	Clear current error status
DFU_GETSTATE	0x05	Get current device status
DFU_ABORT	0x06	Request device to exit current status and go to idle mode

Table 2 DFU class-specific requests

bmRequest	bRequest	wValue	wIndex	wLength	Data
00100001b	DFU_DETACH	wTimeout	interface	Zero	None
00100001b	DFU_DNLOAD	wBlockNum	interface	Length	Firmware
10100001b	DFU_UPLOAD	Zero	interface	Length	Firmware
00100001b	DFU_GETSTATUS	Zero	interface	6	Status
00100001b	DFU_CLRSTATUS	Zero	interface	Zero	None
00100001b	DFU_GETSTATE	Zero	interface	1	State
00100001b	DFU_ABORT	Zero	interface	Zero	None

2 DFU Bootloader commands

DFU_DNLOAD and DFU_UPLOAD commands are used to write/read memory. They can also serve other Bootlaoder requests (access protection, erase and program protection, erase. etc). DFU_GETSTATUS command is used to trigger the execution of commands.

DFU write memory command (DFU_DNLOAD) uses the parameter wValue to determine whether to write memory or other Bootloader command. If wValue=0, it indicates that the host is requesting Bootloader command rather than writing memory, the received 1st byte representing the command name.

DFU read memory command (DFU_UPLOAD) uses the parameter wValue to determine whether to read memory or other Bootloader command. If wValue=0, it indicates that the host is requesting a Bootloader command rather than reading memory, the received first byte representing the command name.

Table 3 DFU Bootloader command list

DFU request	Commands	Code	Description	Applicable products
DFU_UPLOAD	Read Memory	NA	Read memory data at a given address	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Get	NA	Get Bootloader information such as commands, project ID, product ID and version	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Get CRC	0xAC	Get sector CRC	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Get sLib Status	0xD2	Get sLib status	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
DFU_DNLOAD	Write Memory	NA	Write data to a designated address	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Set Address	0x21	Set read/write address	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Access Protect	0x82	Enable access protection	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx,

DFU request	Commands	Code	Description	Applicable products
DFU_DNLOAD				AT32F437xx
	Access Unprotect	0x92	Disable access protection	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Erase and Program Protect	0x63	Enable erase/program protection	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Erase and Program Unprotect	0x73	Disable erase/program protection	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Erase	0x41	Erase memory	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Jump	0x18	Jump to a designated address	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Firmware CRC	0xAC	Calculate sector CRC	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Enable SPIM	0xB3	Enable external memory	AT32F413xx, AT32F403Axx, AT32F407xx
	Enable sLib	0xD0	Enable sLib	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Disable sLib	0xD1	Disable sLib	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Reset Device	0xD4	Reset device	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F435xx, AT32F437xx
	Advanced Access Protect	0xD6	Enable advanced access protection	AT32F415xx

3 DFU_UPLOAD requests

DFU_UPLOAD request uses the parameter wValue to determine whether to read memory or other Bootloader command.

Figure 1 DFU_UPLOAD request flow chart on device side

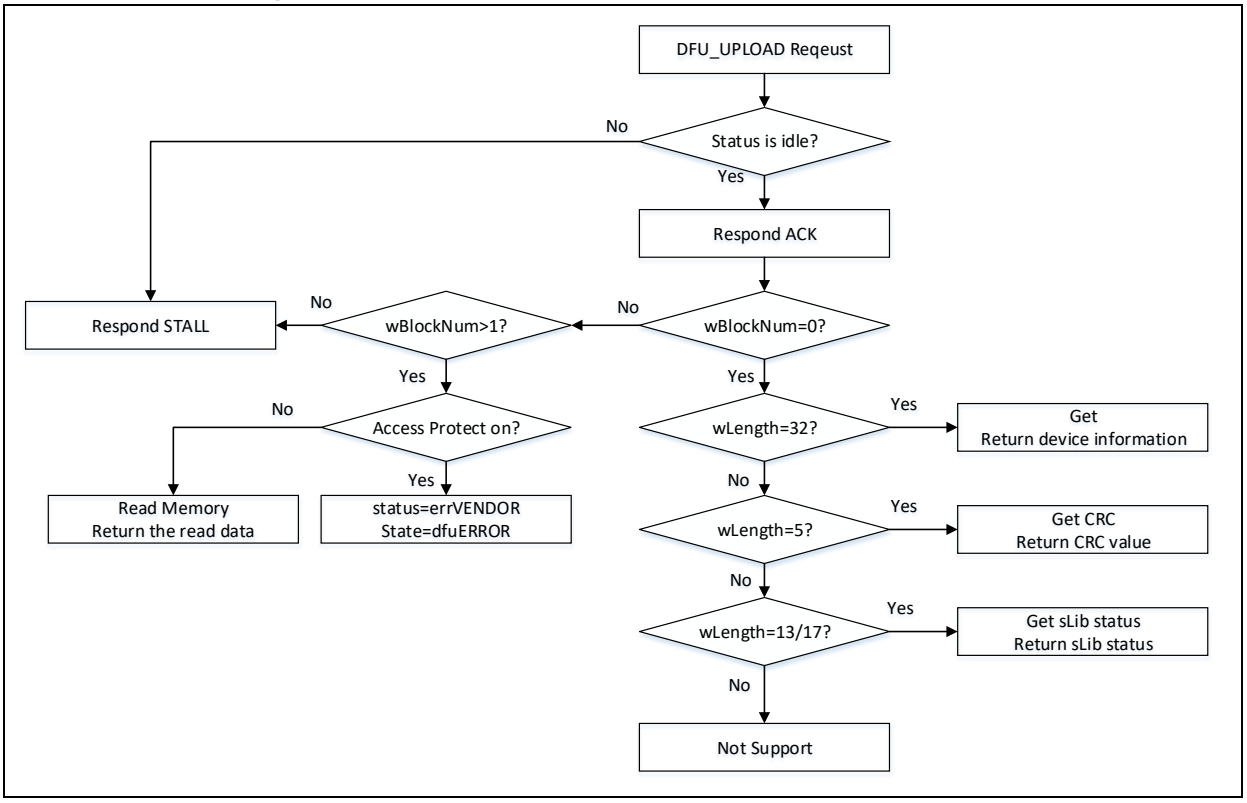
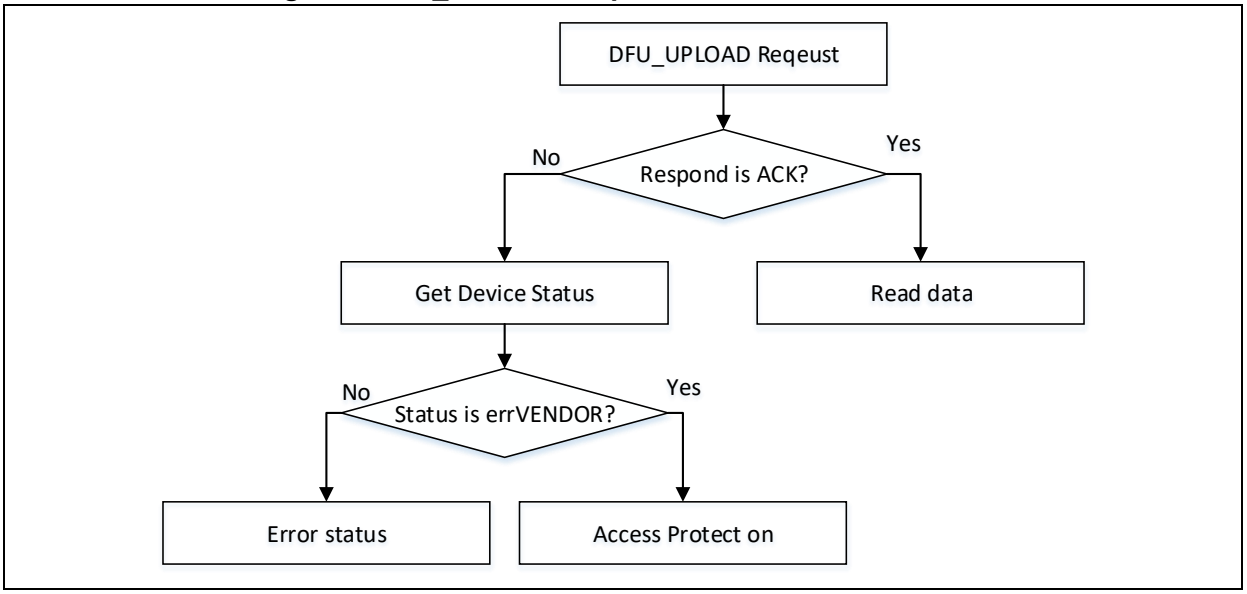


Figure 2 DFU_UPLOAD request flow chart on host side



3.1 Get

To use this command, wValue=0 and wLength=32 must be set. This command is still valid even if access protection is enabled.

With this command, the host can get device information and the supported commands. The product ID and project ID is fixed 12 ~16 bytes, which can be used to obtain MCU part number and its available commands.

Note: The supported commands depend on the microcontrollers. Thus data returned from Get command is subject to the particular microcontroller.

Table 4 Get command return command

Byte	Value	Description
0	0x00	Command
1	0x21	Set Address
2	0x41	Erase
3	0x92	Access Unprotect
4	0x82	Access Protect
5	0x63	Erase and Program protect
6	0x73	Erase and Program unprotect
7	0x18	Jump
8	0xAC	Firmware CRC
9	0xFF	Reserve
10	0x32	Protocol Version
11	0x00	Reserve
12	*	Product ID (LSB)
13	*	Product ID
14	*	Product ID
15	*	Product ID (MSB)
16	*	Project ID
17	*	Bootloader ID
18	*	Bootloader ID
19	0xD0	Enable sLib（if not support reserve）
20	0xD1	Disable sLib（if not support reserve）
21	0xD2	Get sLib status（if not support reserve）
22	0xB3	Enable SPIM（if not support reserve）
23	*	Reserve
24	0xD4	Reset Device（if not support reserve）
25	*	Reserve
26	0xD6	Advanced Access Protect（if not support reserve）
27~31	*	Reserve

Table 5 AT32 project ID list

MCU family	Project ID
AT32F403xx	0x02
AT32F413xx	0x04
AT32F415xx	0x05
AT32F403Axx	0x07
AT32F407xx	0x08
AT32F421xx	0x09
AT32F435xx	0x0D
AT32F437xx	0x0E
AT32F425xx	0x0F

3.2 Read memory

When wValue > 1, it indicates read memory command. When access protection is enabled, read memory is prohibited.

By sending a valid address and wLength by host, main memory, SRAM and user system data can be read.

- Main memory and SRAM: 2 ~ 2048 bytes for a single read operation
- User system data: read length must be aligned with the user system data area size.

Note: Refer to the particular user manual for more information on valid addresses.

Read address is calculated based on the following formula:

Initial address = (wValue – 2)* wlength + Address_Pointer

The Address_Pointer is configured through Set Address command of DFU_DNLOAD.

When access protection is enabled, read memory is prohibited, and device reports Status=dfuERROR, state=errVENDOR.

3.3 Get CRC

To use this command, wValue = 0 and wLength = 5 must be set. This command is still valid even if access protection is enabled.

The Get CRC is used to check if firmware is correct or not. The CRC is calculated based on sector level. The host uses Firmware CRC of DFU_DNLOAD to configure the start address and number of sectors that are to be calculated, and then uses Get CRC of DFU_UPLOAD to get CRC result.

Get CRC return data:

Byte	Value	Description
0	0xAC	Get CRC
1	*	CRC (LSB)
2	*	CRC
3	*	CRC
4	*	CRC (MSB)

3.4 Get sLib status

To use this command, wValue = 0 and wLength = 13/17 must be set. This command cannot be used when access protection is enabled.

Get sLib status is used to get the current sLib status and the corresponding sLib register value. Refer to the particular reference manual for more information on sLib registers.

AT32F435xx/AT32F437xx: (return 17-byte data)
After receiving this command, the device sends an ACK to host, and receives 4-byte SLIB_STS0 register value, 4-byte SLIB_STS1 value, 4-byte SLIB_STS2 value and 4-byte SLIB_MISC_STS value, and then sends an ACK to host.

Others: return 13-byte data
After receiving this command, the device sends an ACK to host, and receives 4-byte SLIB_STS0 value, 4-byte SLIB_STS1 value, and 4-byte SLIB_MISC_STS value, and then sends an ACK to host.
Note: This command cannot be used in *AT32F403Axx, AT32F407xx, AT32F413xx in access protection enable mode. Others are not affected by this.*

AT32F435xx/AT32F437xx return data:

Byte	Value	Description
0	0xD2	Get sLib Status
1	*	SLIB_STS0 (LSB)
2	*	SLIB_STS0
3	*	SLIB_STS0
4	*	SLIB_STS0 (MSB)
5		SLIB_STS1 (LSB)
6		SLIB_STS1
7		SLIB_STS1
8		SLIB_STS1 (MSB)
9		SLIB_STS2 (LSB)
10		SLIB_STS2
11		SLIB_STS2
12		SLIB_STS2 (MSB)
13		SLIB_MISC_STS (LSB)
14		SLIB_MISC_STS
15		SLIB_MISC_STS
16		SLIB_MISC_STS (MSB)

For others:

Byte	Value	Description
0	0xD2	Get sLib Status
1	*	SLIB_STS0 (LSB)
2	*	SLIB_STS0
3	*	SLIB_STS0
4	*	SLIB_STS0 (MSB)
5	*	SLIB_STS1 (LSB)
6	*	SLIB_STS1
7	*	SLIB_STS1
8	*	SLIB_STS1 (MSB)
9	*	SLIB_MISC_STS (LSB)
10	*	SLIB_MISC_STS
11	*	SLIB_MISC_STS
12	*	SLIB_MISC_STS (MSB)

4 DFU_DNLOAD request

The DFU_DNLOAD request uses wValue parameter and the first byte of data to determine which kind of Bootloader command is being used.

Table 6 DNLOAD commands

Command	Byte0 Value	wValue
Write Memory	*	>1
Set Address	0x21	0
Access Protect	0x82	0
Access Unprotect	0x92	0
Erase and Program protect	0x63	0
Erase and Program unprotect	0x73	0
Erase	0x41	0
Jump	0x18	0
Firmware CRC	0xAC	0
Enable sLib	0xD0	0
Disable sLib	0xD1	0
Reset Device	0xD4	0
Advanced Access Protect	0xD6	0

Note: Before starting DFU_DNLOAD, the host must check if the device is in dfuIDLE or dfuDNLOAD-IDLE state. If the device is not in idle state, the host must send DFU_CLRSTATUS status and re-obtain the device status and waits until the device enters dfuIDLE state.

Figure 3 shows DFU_DNLOAD request flow chart on host and device side

Figure 3 DFU_DNLOAD request flow chart on device side

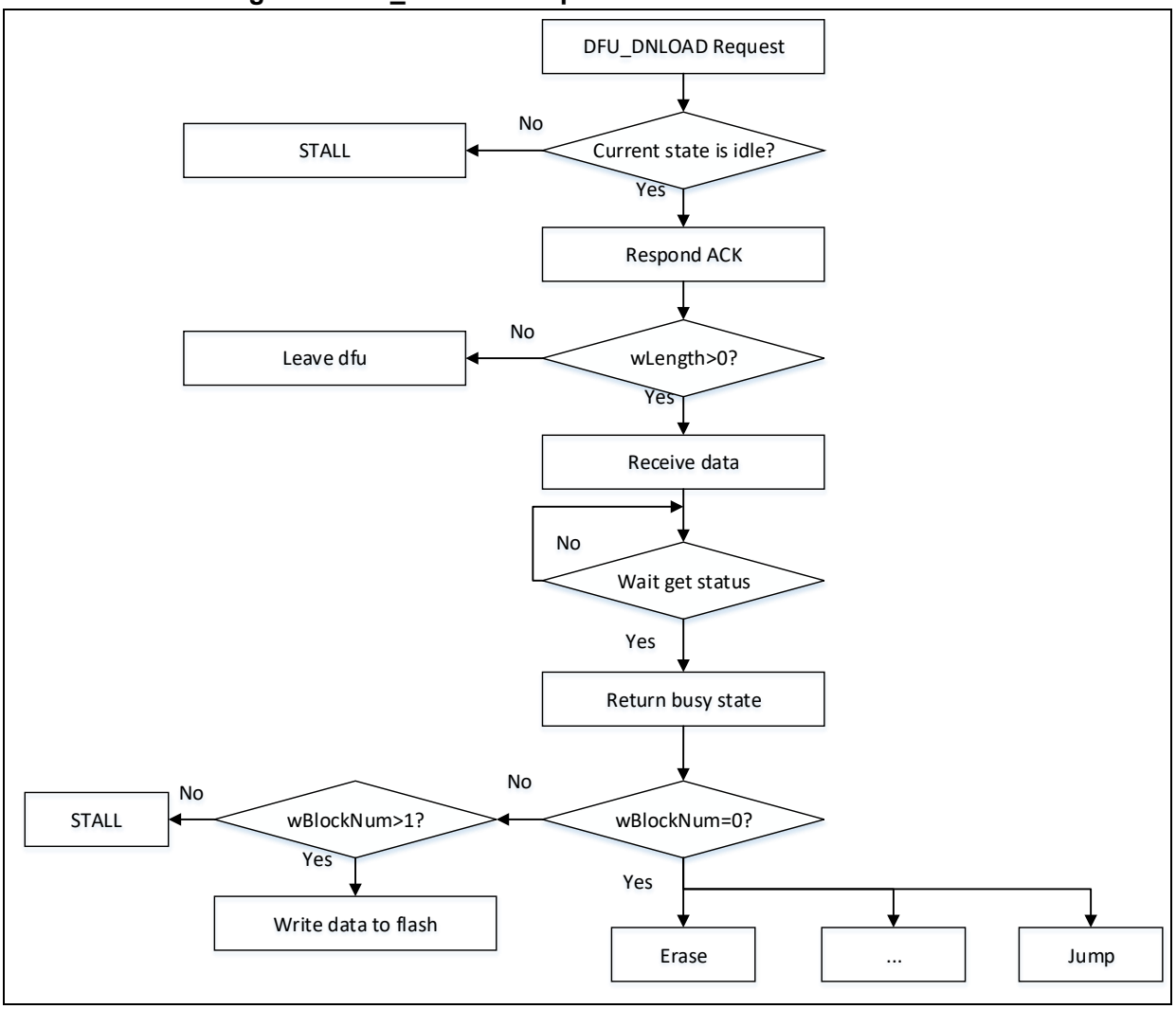
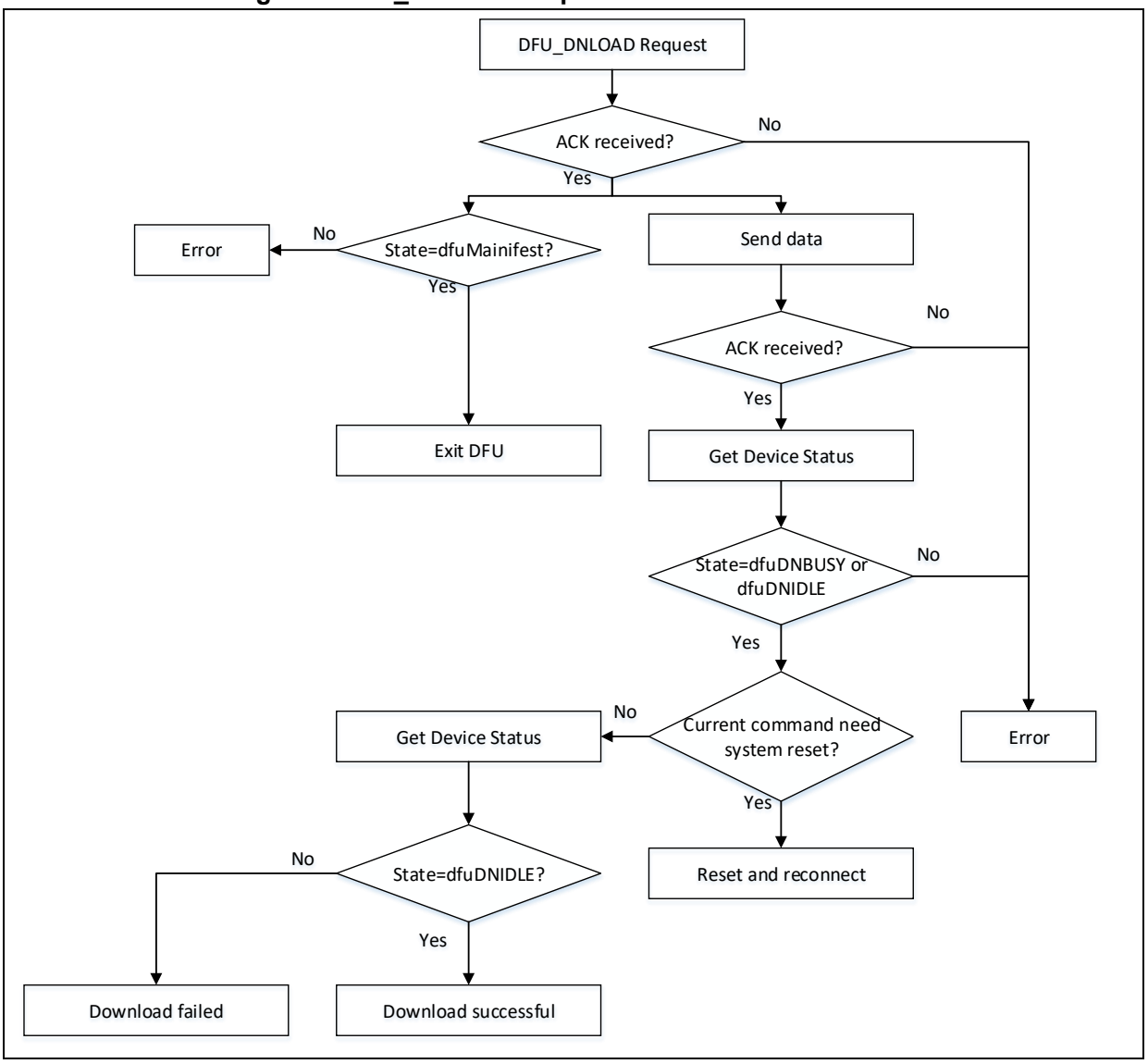


Figure 4 DFU_DNLOAD request flow chart on host side



4.1 Write memory

When wValue > 1, it indicates Write memory command. This command cannot be used when access protection is enabled.

The host sends a valid address (through Set Address command) and data length (wLength) to device so that the device writes data to a given address.

Write memory address:

- Flash memory and SRAM: 2 ~ 2048 bytes for a single write operation
- User system data: its size must be aligned with the user system area size

Note: Refer to the particular reference manual for more information on valid addresses.

After sending write memory, the host must send DFU_GETSTATUS command to allow device to perform write access. At this point, the device returns dfuDNBUSY state (indicating that data is being written to memory). If the address area is user system data, the device will perform system reset at the end of write data, and the host must be reconnected to the device.

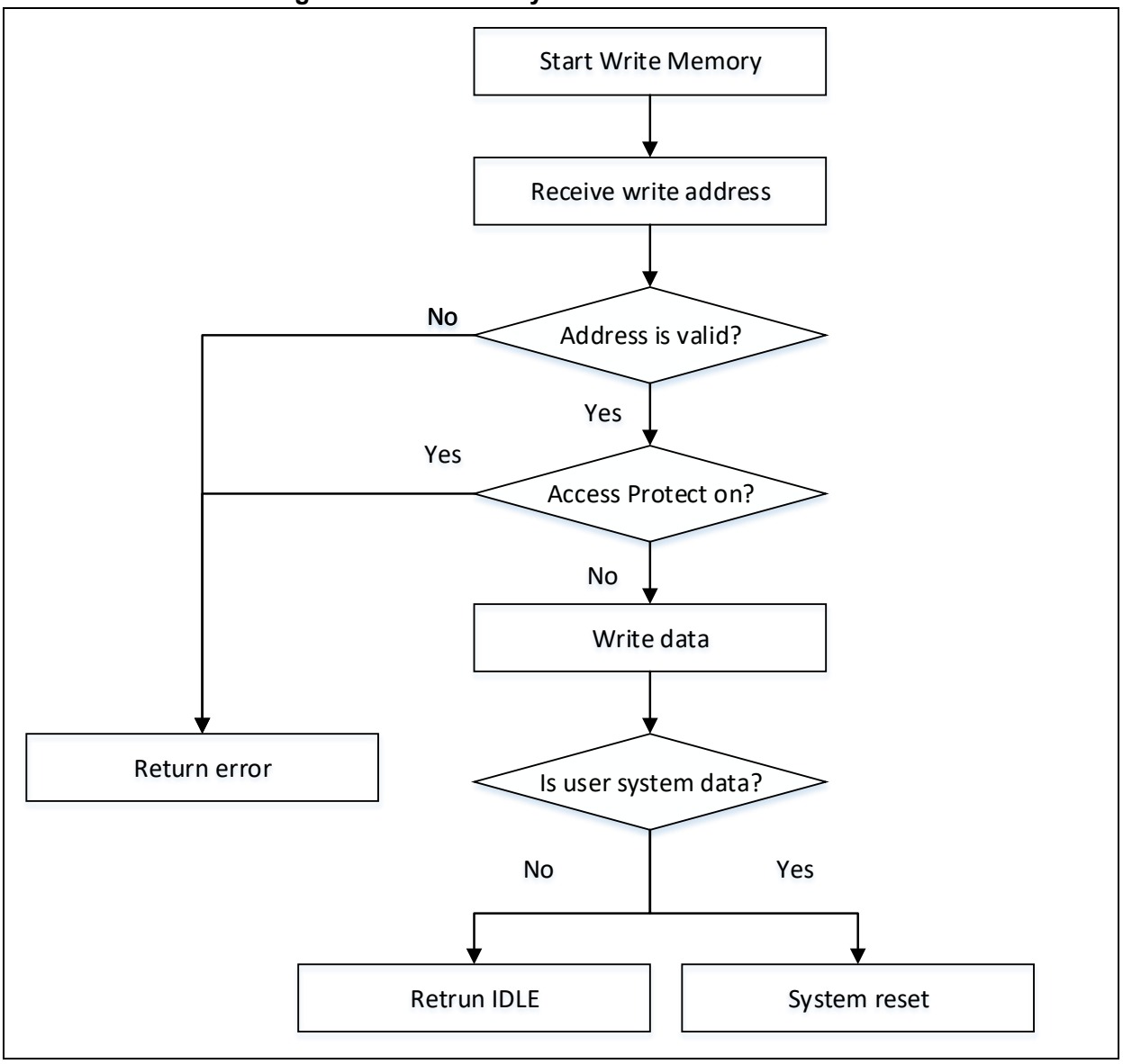
The host then sends DFU_GETSTATUS command once more to check if write memory is complete or not. If the transmit address is valid, the device returnsstatus=dfuERROR, state =STATUS_ERRADDRESS.

Write address is calculated based on the following formula:

Initial address = (wValue – 2)* wlength + Address_Pointer

If access protection is enabled, writing memory returns status=dfuERROR, state=errVENDOR.

Figure 5 Write Memory flow chart on device side



4.2 Set address

When wValue=0 and the first byte is 0x21, it indicates a Set Address command. This command cannot be used when access protection is enabled.

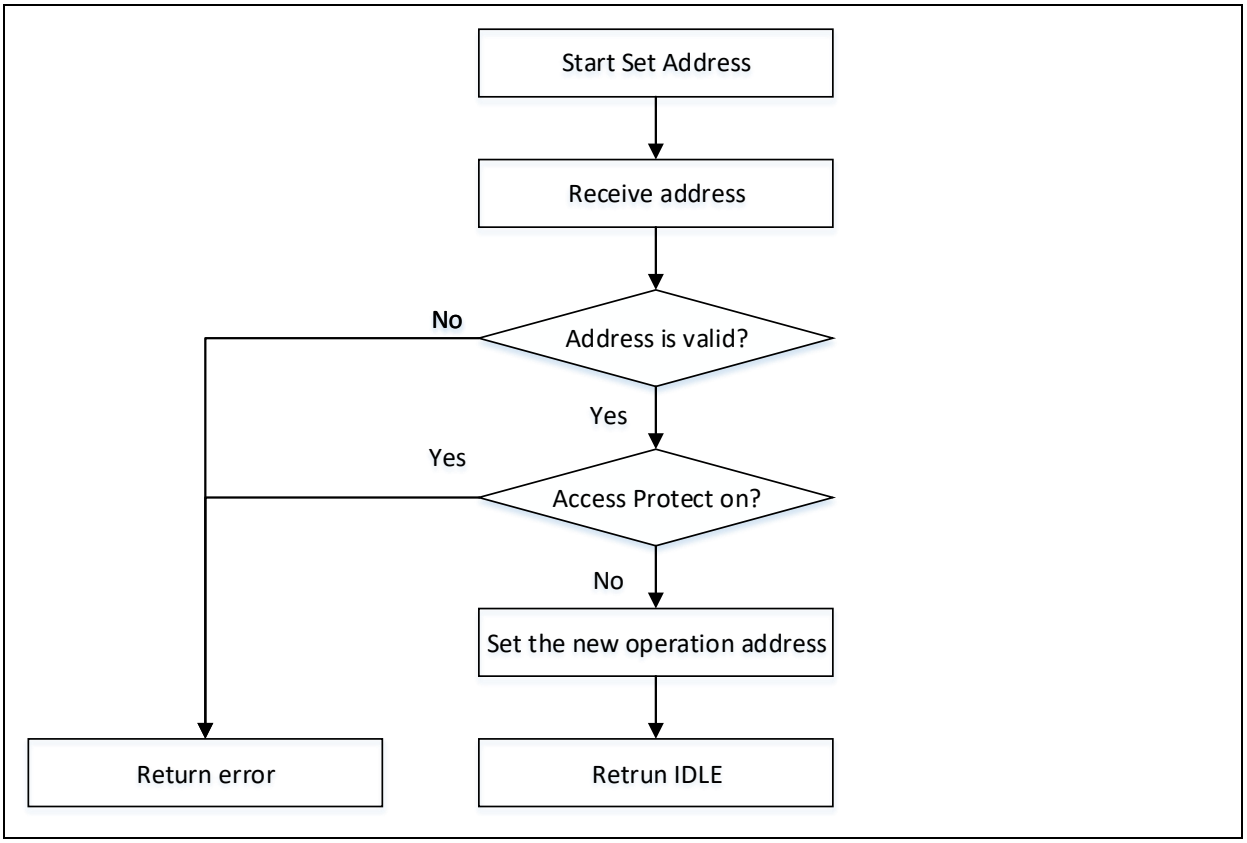
Before starting Write Memory and Read Memory, it is necessary to set a start address Address_Pointer for write or read operation with Set Address command. The host sends 1-byte Set Address command and 4-byte address.

After sending Set Address command, the host must send DFU_GETSTATUS request to drive device to set address, in this case, the device returns STATE_dfuDNLOAD_IDLE (indicating that set address is complete).

Host transmit data:

Byte	Value	Description
0	0x21	Set Address command
1	*	Address (LSB)
2	*	Address
3	*	Address
4	*	Address (MSB)

Figure 6 Get Commands flow chart on device side



4.3 Access protect

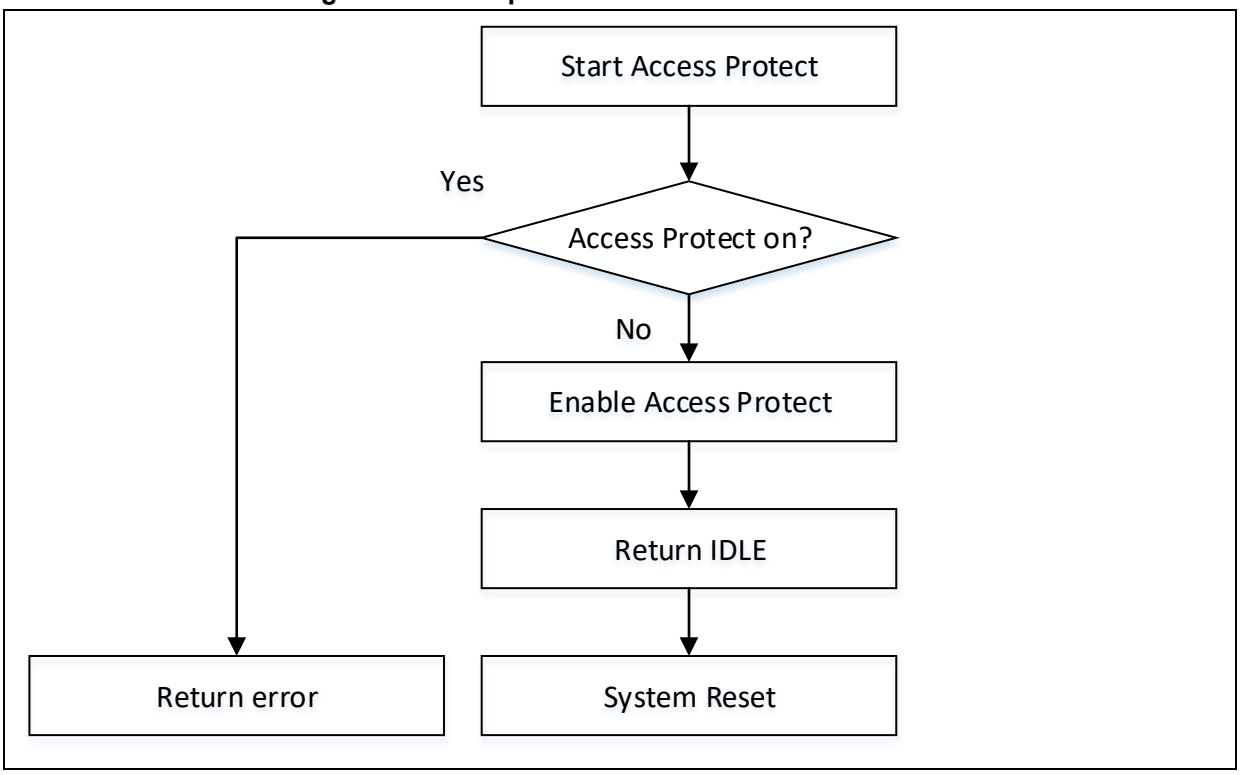
When wValue=0 and the first byte is 0x82, it indicates that access protection is enabled. In this case, this command cannot be used.

After sending 1-byte access protection enable command 0x82, the host must send DFU_GETSTATUS request to drive the device to perform access protection. In this case, the device returns STATE_dfuDNLOAD_IDLE, and then performs a system reset so that access protection operation takes effect. The host must be reconnected to the device at this time.

Host transmit data:

Byte	Value	Description
0	0x82	Access Protect

Figure 7 Access protect flow chart on device side



4.4 Access unprotect

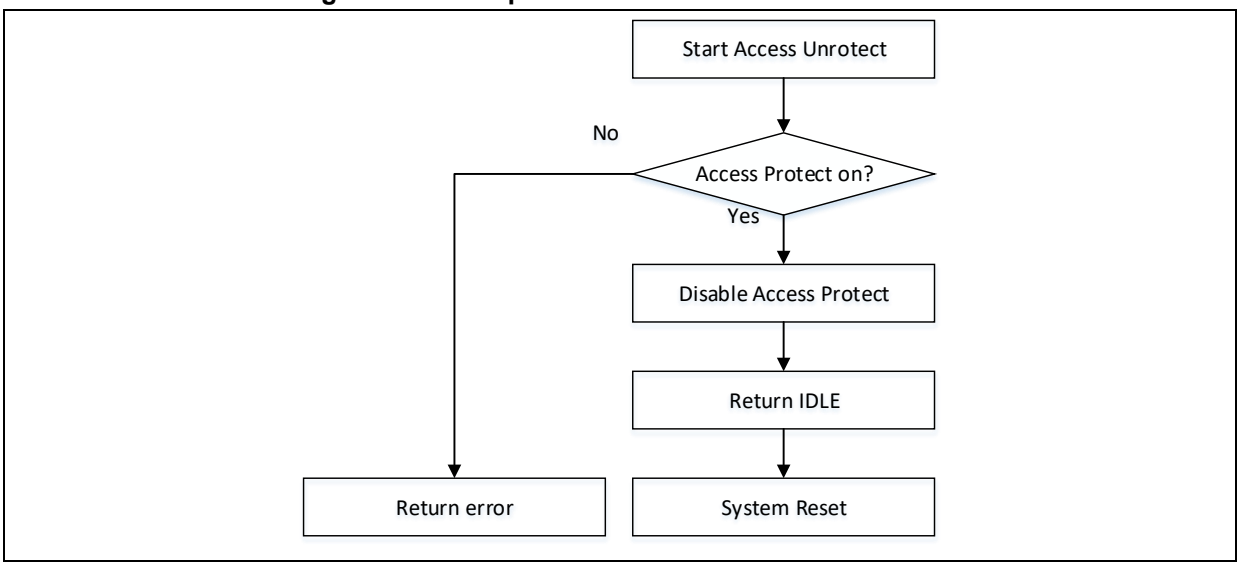
When wValue=0 and the first byte is 0x92, it indicates that access protection is unlocked. This command is used when access is protected.

After sending 1-byte access unprotect command 0x92, the host must send DFU_GETSTATUS request to drive the device to unlock access protection. In this case, the device returns STATE_dfuDNLOAD_IDLE, and main memory data is automatically erased, before performing a system reset.

Host transmit data:

Byte	Value	Description
0	0x92	Access Unprotect

Figure 8 Access protect flow chart on device side



4.5 Erase and program protect

When wValue=0 and the first byte value is 0x63, it indicates that erase and program operation is protected. This command cannot be used when access protection is enabled.

The host sends 1-byte erase and program protect command, 1-byte write-protected sector and n-byte sector index. Then the device writes user system data area accordingly.

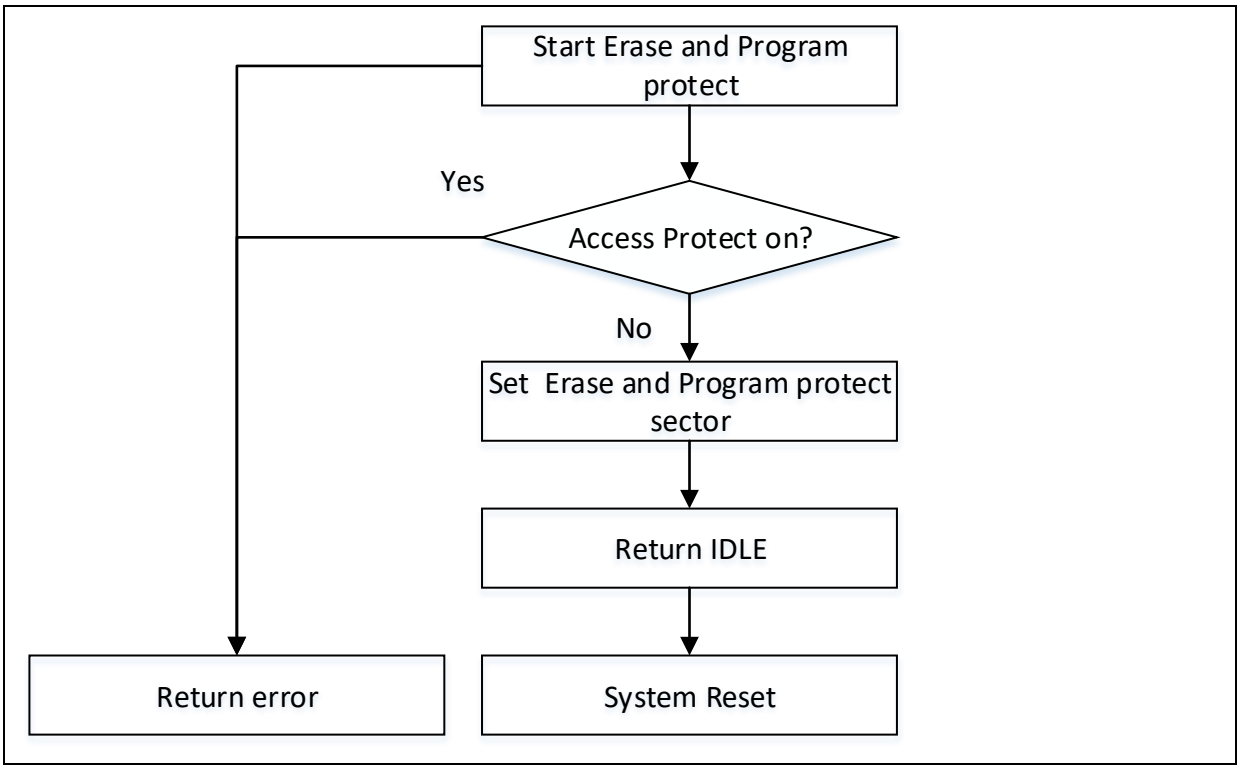
After sending write protection command, the host must send DFU_GETSTATUS request to drive the device to enable write protection. A system reset is performed at the end of the operation.

Note: The value of index bit (0, 1, 2…n) in the erase and program protection corresponds to (0-N) bit of the erase and program protection byte in the user system data. Refer to the particular reference manual for more information on user system data.

Host transmit data:

Byte	Value	Description
0	0x63	Erase and Program Protect
1	*	Number of sector index
2	*	Sector index 0
3	*	Sector index1
4	…	…

Figure 9 Erase and program protect flow chart on device side



4.6 Erase and program unprotect

When wValue=0 and the first byte is 0x73, it indicates that erase and program is unprotected. This command cannot be used when access protection is enabled.

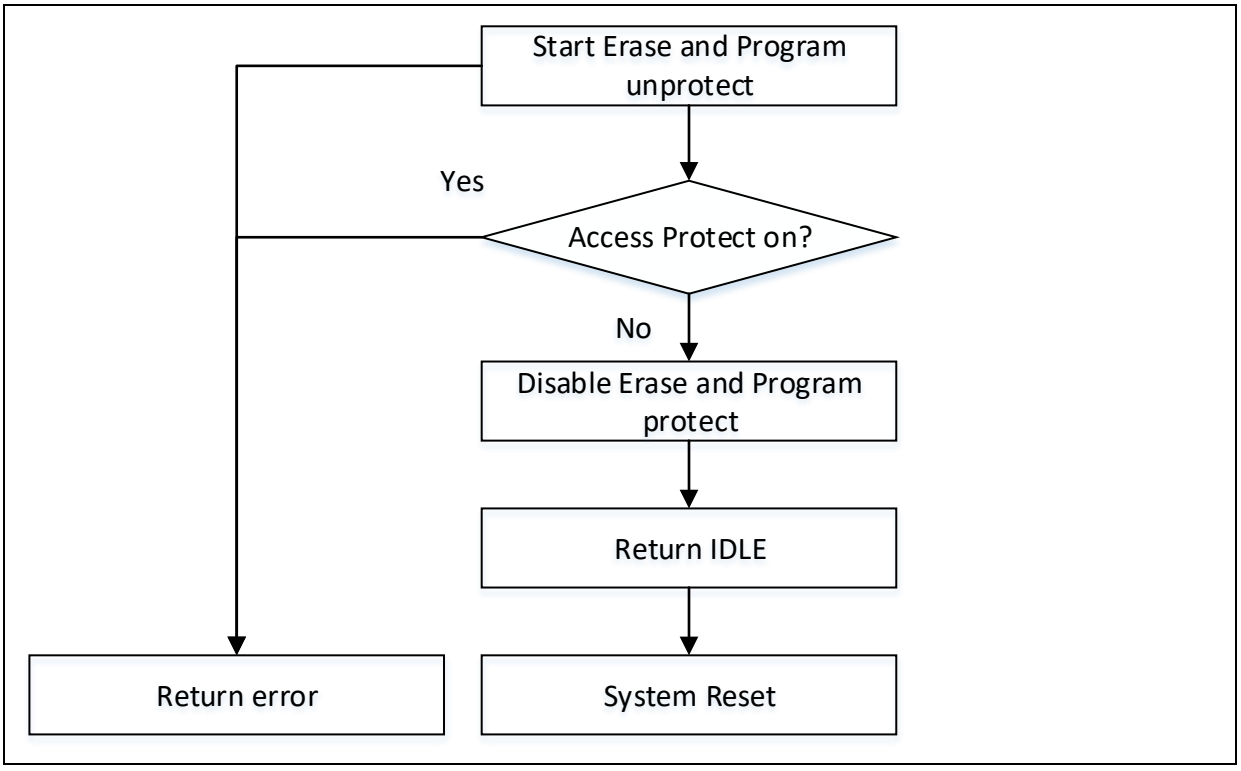
This command is used to disable erase and program protection for all sectors. The host only needs to send 1-byte unlock command.

After sending erase and program protect disable command, the host must send DFU_GETSTATUS request to drive the device to unlock erase and program protection. The device then disables erase and program protection and returns STATE_dfuDNLOAD_IDLE. A system reset is performed by device.

Host transmit data:

Byte	Value	Description
0	0x73	Erase and Program Unprotect

Figure 10 Erase and program unprotect flow chart on device side



4.7 Erase

When wValue=0 and the first byte is 0x41, it indicates that erase command is enabled. This command cannot be used when access protection is enabled.
This command supports section erase (sector size depends on the microcontrollers) and mass erase. Even bank 1 and bank2 erase are supported for devices with bank2. Bank3 erase is applicable for devices with SPIM support. The erase addresses must be valid, and the valid address range depends on the microcontrollers.

- Mass erase: wLength=1, 1-byte erase command is sent.
- Sector erase: wLength=5. The host sends 1-byte erase command and 4-byte erase address (sector aligned)

- Bank erase: wLength=3. The host sends 1-byte erase command and 2-byte bank index.
- Block erase: wLength=7. The host sends 1-byte erase command, 2-byte block index and 4-byte erase address.

Note: *AT32F435xx/AT32F437xx support block erase (64 Kbytes aligned).*

Table 7 Erase index table

Erase type	Erase index	Erase location
Mass erase	0xFF 0xFF	All Flash
Bank erase	0xFF 0xFE	Bank1 Erase
	0xFF 0xFD	Bank2 Erase
	0xFF 0xFC	Bank3 Erase
Block erase	0xFF 0xFB	Block Erase

After sending an erase command, the host must send DFU_GETSTATUS request. If it is not dfuDNBUSY state, it indicates that an erase error may occur, such as, invalid address, access protection is enabled. DFU_GETSTATUS command is used to get the current device status. The dfuDNLOAD_IDLE state indicates the completion of an erase operation.

Host transmit data for mass erase: wLength=1

Byte	Value	Description
0	0x41	Erase
1	*	Number of sector index
2	*	Sector index 0
3	*	Sector index1
4	…	…

Host transmit data for sector erase: wLength=5

Byte	Value	Description
0	0x41	Erase
1	*	Address (LSB)
2	*	Address
3	*	Address
4	*	Address
5	*	Address (MSB)

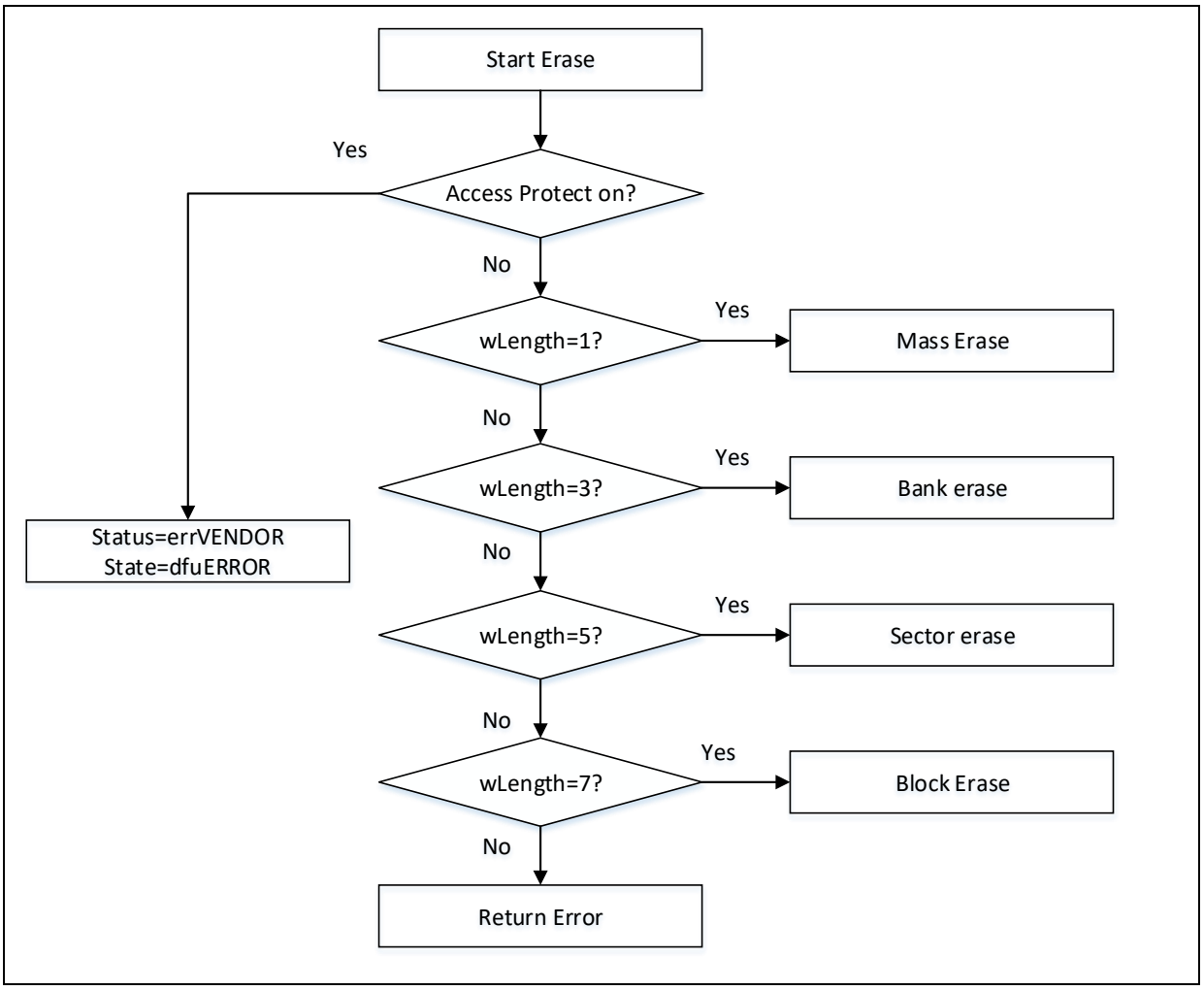
Host transmit data for bank erase: wLength=3

Byte	Value	Description
0	0x41	Erase
1	0xFF	Bank index
2	*	Bank index (0XFE 0xFD 0xFC)

Host transmit data for block erase: wLength=7

Byte	Value	Description
0	0x41	Erase
1	0xFF	Block index
2	0xFB	Block index
3		Address (LSB)
4	*	Address
5	*	Address
6	*	Address (MSB)

Figure 11 Erase flow chart on device side



4.8 Jump

When wValue=0 and the first byte is 0x18, it indicates a jump to user applications. This command cannot be used when access protection is enabled.

This command is used to jump to a given address. The host must send 1-byte jump command and 4-byte valid jump address.

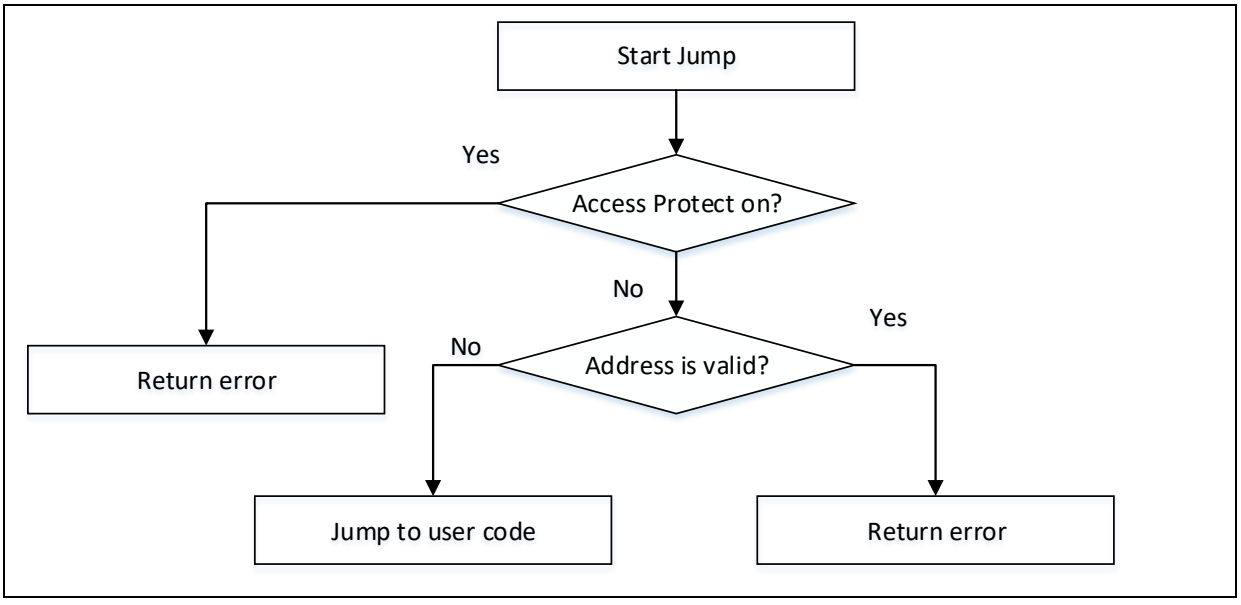
After sending a jump command, the host must send DFU_GETSTATUS request. Then the device returns dfuDNLOAD_IDLE, indicating that it is ready to jump to user applications.

Note: Valid addresses include SRAM, Flash and SPIM(BANK3 addresses. Refer to the particular reference manual for more information.

Host transmit data:

Byte	Value	Description
0	0x18	Jump
1	*	Address (LSB)
2	*	Address
3	*	Address
4	*	Address
5	*	Address (MSB)

Figure 12 Jump flow chart on device side



4.9 Firmware CRC

When wValue=0 and the first byte is 0xAC, it indicates a CRC command. This command can be used even if access protection is enabled.

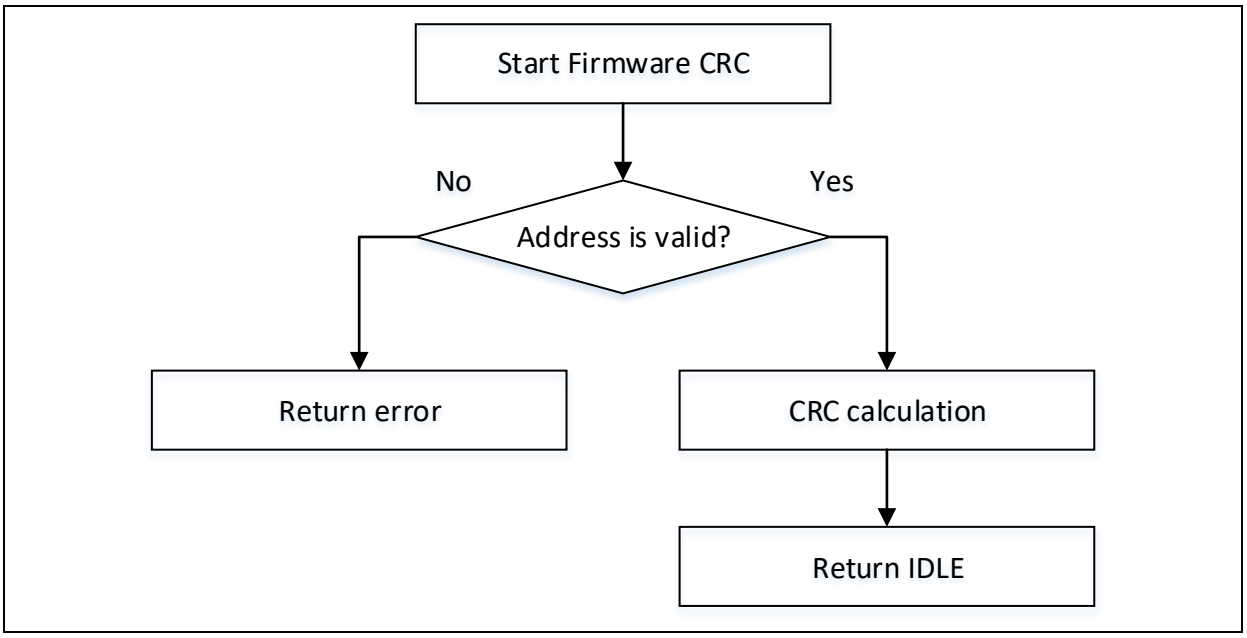
The CRC check is based on sector level. The host sends 1-byte CRC command, 4-byte start address (sector aligned) and 2-byte sector.

After sending CRC command, the host must send DFU_GETSTATUS request. If the device returns dfuDNLOAD_IDLE, it indicates the completion of CRC calculation. The Get CRC of DFU_UPLOAD is used to get the current CRC result.

Host transmit data:

Byte	Value	Description
0	0xAC	Erase
1	*	Address (LSB)
2	*	Address
3	*	Address
4	*	Address (MSB)
5	*	Number of sector (LSB)
6	*	Number of sector (MSB)

Figure 13 Firmware CRC flow chart on device side



4.10 Enable SPIM

When wValue=0 and the first byte is 0xB3, it indicates Enable Bank3 command. This command can be used even if access protection is enabled.

To use this command, SPIM Flash type, Flash size and Flash encryption range must be set. Refer to SPIM user manual for more information.

Flash type list:

- 0x90: General Flash
  Dummy cycle is 4
- 0x91: General Flash, Quad Enable
  Dummy cycle is 4
  Quad Enable in Volatile format

Flash size supported is up to 16Mbytes. The device performs erase and programming operations based on the Flash size sent by host.

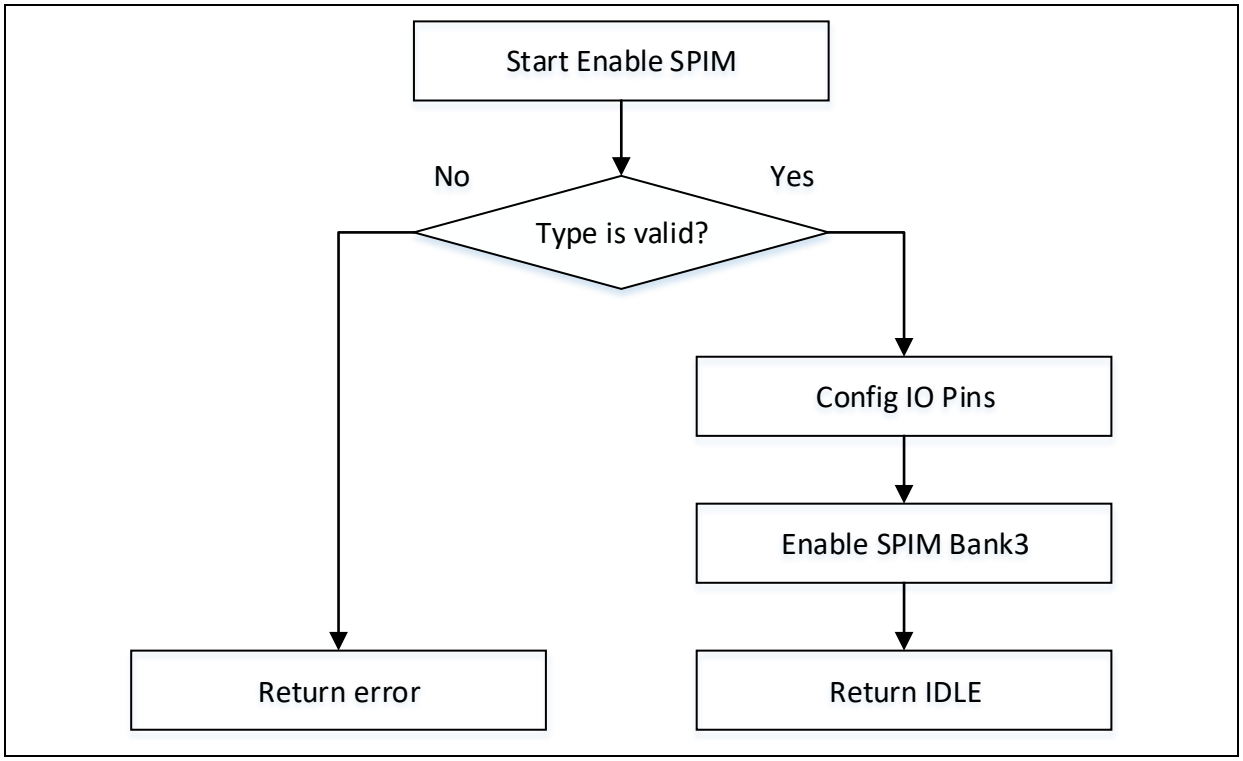
The SPIM has an encryption feature to protect SPIM Flash data against read. In this case, the Flash controller writes the encrypted data into SPIM, but can read it in plaintext form. For MCU, read and write are performed in plaintext format, but the data stored in external Flash are encrypted. It is up to the user to set a password and range for encryption operation. The password is located in user system area, and must be configured before writing SPIM (bank3).

Encrypted range (FLASH_FDA) must be set whenever enabling SPIM (bank3). It is used to define the number of bytes to be encrypted from the start address of SPIM (bank3). If the value is greater than 16 M Bytes, it indicates that the entire SPIM (bank3) must be encrypted; if the value is 0, it indicates that the SPIM (bank3) is not encrypted.

Host transmit data:

Byte	Value	Description
0	0xB3	Enable SPIM
1	*	Flash type
2	*	Flash size (LSB)
3	*	Flash size
4	*	Flash size
5	*	Flash size (MSB)
6	*	Flash FDA (LSB)
7	*	Flash FDA
8	*	Flash FDA
9	*	Flash FDA (MSB)

Figure 14 SPIM enable flow chart on device side



4.11 Enable sLib

When wValue=0 and the first byte is 0xD0, it indicates enable sLib command. This command cannot be used when access protection is enabled.

To enable this command, it is necessary to set 4-byte sLib password, 2-byte sLib start sector, 2-byte sLib data/instruction start sector and 2-byte sLib end sector. sLib settings take effect only after a system reset.

After sending Enable sLib command, the host must send DFU_GETSTATUS request to drive the device to enable sLib. A system reset must be performed to have sLib settings take effect after enabling sLib and downloading data to memory.

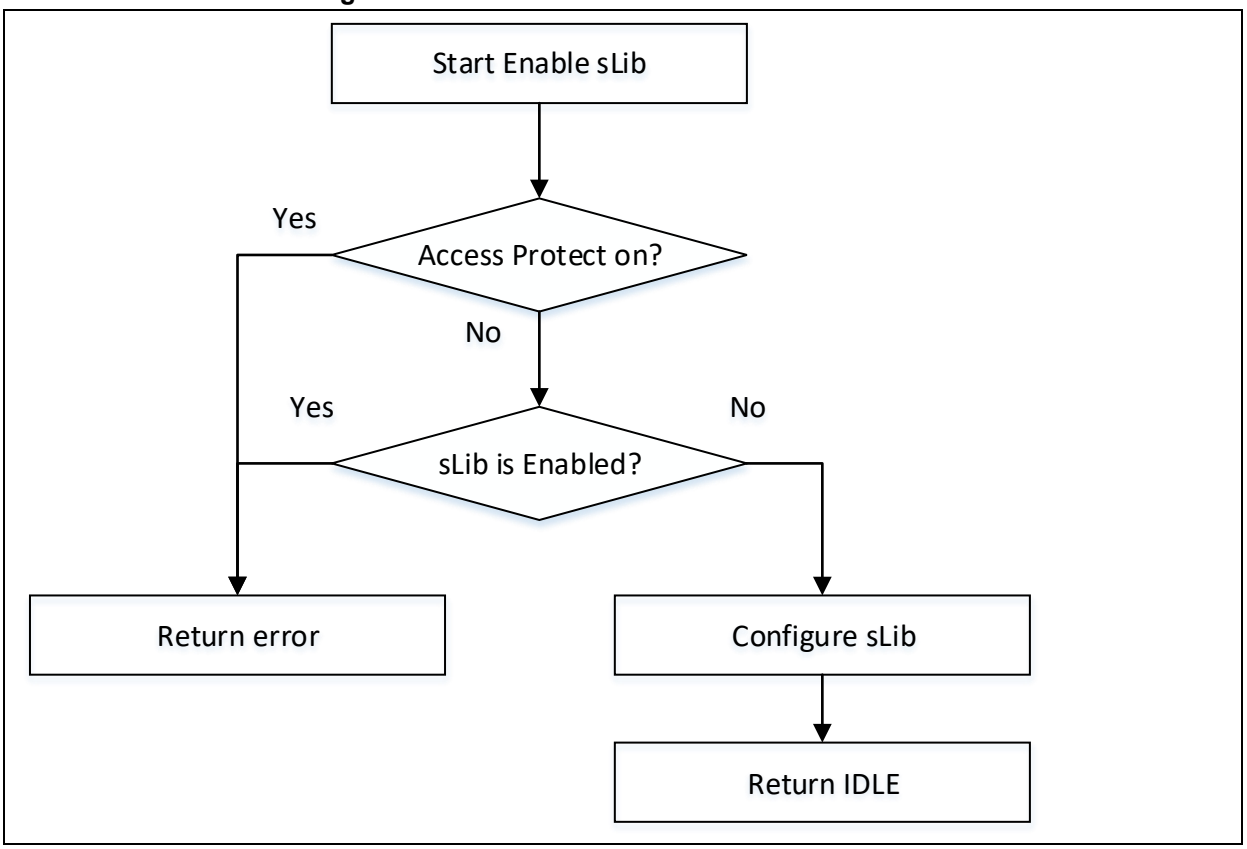
Note: 1. Refer to Security library user guideline for more information on sLib.

2. The sLib status must be obtained prior to sLib enable. If sLib is already enabled, it is necessary to disable sLib before re-enabling sLib.

Host transmit data:

Byte	Value	Description
0	0xD0	Enable sLib
1	*	Password (LSB)
2	*	Password
3	*	Password
4	*	Password (MSB)
5	*	sLib start sector (LSB)
6	*	sLib start sector (MSB)
7	*	sLib data/instruction start sector (LSB)
8	*	sLib data/instruction start sector (MSB)
9	*	sLib end sector (LSB)
10	*	sLib end sector (MSB)

Figure 15 Enable sLib flow chart on device side



4.12 Disable sLib

When wValue=0 and the first byte is 0xD1, it indicates Disable sLib command. This command cannot be used when access protection is enabled.

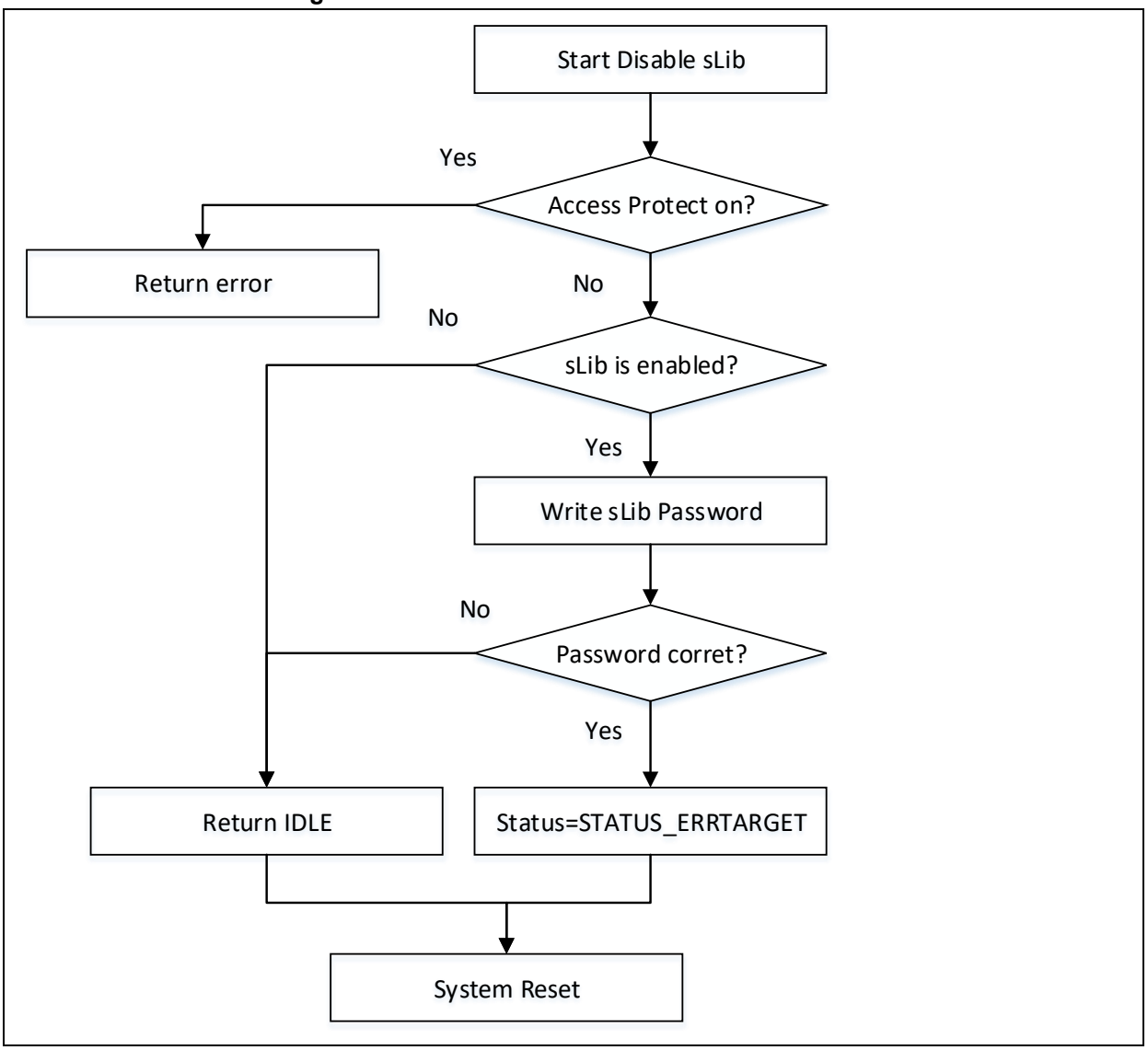
To start this command, the host must send 1-byte disable sLib command and 4-byte password. If a correct password, the device disables sLib and performs a system reset. For a wrong password, the device returns status= STATUS_ERRTARGET.

After sending a disable sLib command, the host must send DFU_GETSTATUS request to drive the device to disable sLib. After disabling sLib, the device performs a system reset automatically (regardless of the correctness of password), and the host must be reconnected to the device.

Host transmit data:

Byte	Value	Description
0	0xD1	Disable sLib
1	*	Password (LSB)
2	*	Password
3	*	Password
4	*	Password (MSB)

Figure 16 Disable sLib flow chart on device side



4.13 Reset device

When wValue=0 and the first byte is 0xD4, it indicates Reset Device command. This command cannot be used when access protection is enabled.

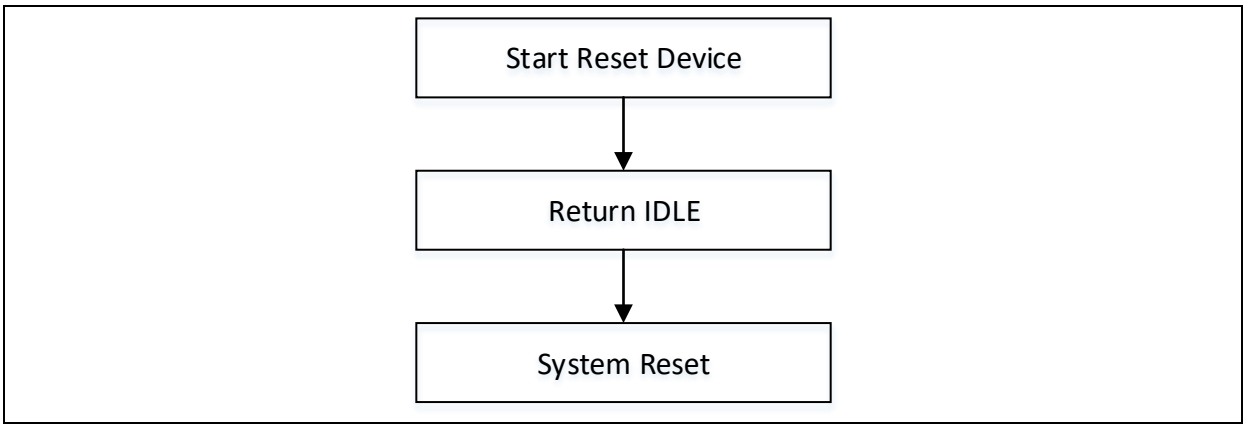
This command is used to reset device after some operations by host.

After sending Reset Device command, the host must send DFU_GETSTATUS request to drive the device to perform reset.

Host transmit data:

Byte	Value	Description

Figure 17 Reset device flow chart on device side



4.14 Advanced Access Protect

When wValue=0 and the first byte is 0xD6, it indicates advanced access protect enable. This command cannot be used when access protection is enabled. Refer to the particular reference manual for more information on high-level access protection.

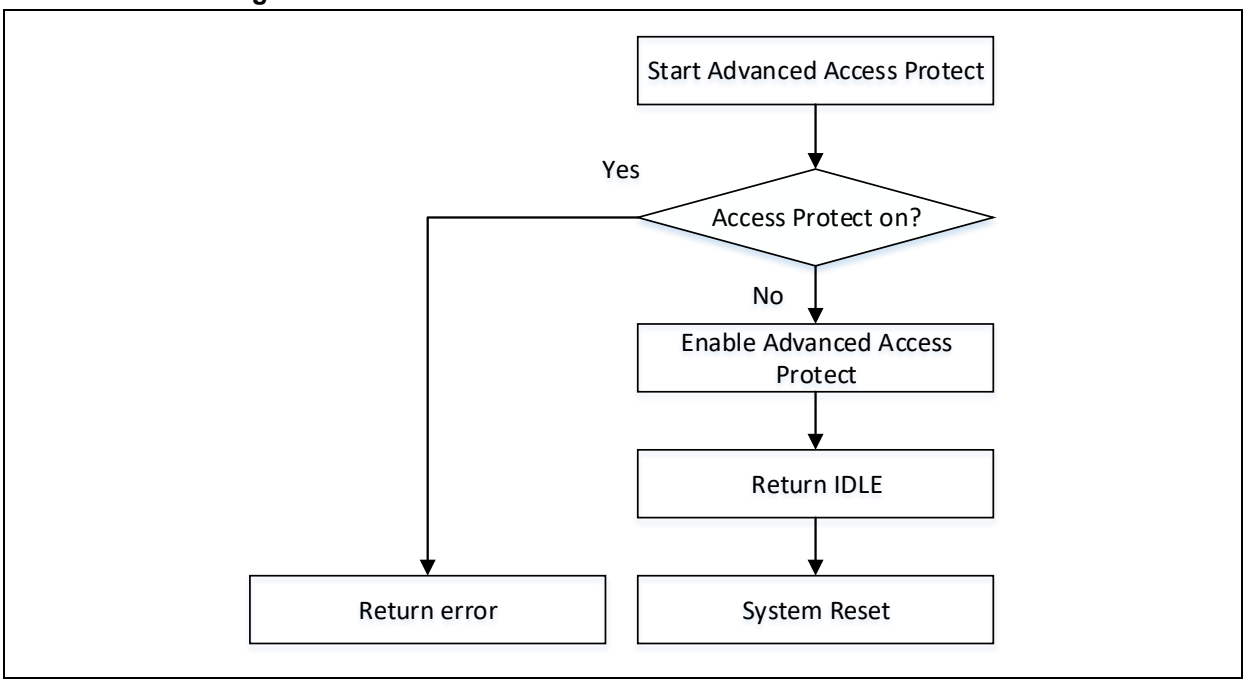
Note: Advanced access protection cannot be unlocked for some devices. Refer to the particular reference manual for more information.

To start this command, the host must send 1-byte access protect enable command, and 1-byte flag. After sending this command, the host must send DFU_GETSTATUS request to drive the device to enable advanced access protection, and then the host performs a system reset. The host must be reconnected to device.

Host transmit data:

Byte	Value	Description
0	0xD6	Advanced Access Protect
1	0x02	Flag

Figure 18 Advanced Access Protect flow chart on device side



5 Revision history

Table 8 Document revision history

Date	Revision	Changes
2021.12.07	2.0.0	Initial release

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.