

AT32 Bootloader USB DFU Protocol

前言

本文档介绍USB DFU 协议在AT32微控制器Bootloader上的使用，以及支持的DFU协议命令。如需要了解更多USB的硬件资源或者当前型号是否支持 USB DFU 请参考如下文档：
《PM0007_AT32_Bootloader_Program_Manual》。

目录

1	USB DFU Bootloader 请求	5
2	DFU Bootloader 命令	6
3	DFU_UPLOAD 请求	8
3.1	Get	8
3.2	Read Memory	9
3.3	Get CRC	10
3.4	Get sLib Status 命令	10
4	DFU_DNLOAD 请求	12
4.1	Write Memory	14
4.2	Set Address	15
4.3	Access Protect	16
4.4	Access Unprotect	17
4.5	Erase and Program Protect	18
4.6	Erase and Program Unprotect	18
4.7	Erase	19
4.8	Jump	21
4.9	Firmware CRC	22
4.10	Enable SPIM	23
4.11	Enable sLib	24
4.12	Disable sLib	25
4.13	Reset Device	26
4.14	Advanced Access Protect	27
5	版本历史	29

表目录

表 1 DFU 请求命令	5
表 2 DFU class-specific requests.....	5
表 3 DFU Bootloader 命令列表	6
表 4 Get 命令返回数据	9
表 5 DNLOAD 命令条件	12
表 6 擦除索引表.....	20
表 7 文档版本历史	29

图目录

图 1 DFU_UPLOAD 请求设备端流程.....	8
图 2 DFU_UPLOAD 请求主机端流程.....	8
图 3 DFU_DNLOAD 请求设备端流程.....	13
图 4 DFU_DNLOAD 请求主机端流程.....	14
图 5 Write Memory 设备端流程	15
图 6 Set Address 设备端流程	16
图 7 Access Protect 设备端流程.....	17
图 8 Access Unprotect 设备端流程图.....	17
图 9 Erase and Program Protect 设备端流程图	18
图 10 Erase and Program Unprotect 设备端流程图.....	19
图 11 Erase 设备端流程图.....	21
图 12 Jump 设备端流程图	22
图 13 Firmware CRC 设备端流程图.....	23
图 14 Enable SPIM 设备端流程图.....	24
图 15 Enable sLib 设备端流程图.....	25
图 16 Disable sLib 设备端流程图	26
图 17 Reset Device 设备端流程图	27
图 18 Advanced Access Protect 设备端流程图.....	28

1 USB DFU Bootloader 请求

AT32 Bootloader支持USB DFU协议和请求，参考文档“Universal Serial Bus Device Upgrade Specification for Device Firmware Upgrade” Version 1.1。

表 1 DFU 请求命令

请求名	代码	描述
DFU_DETACH	0x00	设备退出 DFU 模式（没有实现该功能）
DFU_DNLOAD	0x01	从主机下载数据到设备的存储器里面
DFU_UPLOAD	0x02	从设备的存储器上传数据到主机
DFU_GETSTATUS	0x03	设备发送状态报告到主机
DFU_CLRSTATUS	0x04	清除设备当前的错误状态
DFU_GETSTATE	0x05	仅请求设备当前的状态
DFU_ABORT	0x06	请求设备退出当前状态，进入空闲状态

表 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 命令

DFU_DNLOAD和DFU_UPLOAD请求用于写存储器和读存储器的操作，同时也用于其它Bootloader的请求（如访问保护，擦写保护，擦除等），DFU_GETSTATUS命令用于触发命令的执行。

DFU 写存储器命令（DFU_DNLOAD）通过参数wValue来区分是写存储器还是Bootloader其它命令，如果wValue=0 表示主机请求了一个Bootloader的其它命令，收到的第一个字节表示具体请求的哪一个命令。

DFU读存储器命令（DFU_UPLOAD）通过参数wValue来区分是读存储器数据还是Bootloader的其它命令，如果wValue=0表示主机请求其它命令，收到的第一个字节表示具体请求的哪一个命令。

表 3 DFU Bootloader 命令列表

DFU请求	命令	代码	说明	支持型号
DFU_UPLOAD	Read Memory	无	读取存储器指定地址数据	Bootloader支持DFU的型号
	Get	无	获取Bootloader信息包括支持命令，Project ID, Product ID和版本号等	Bootloader支持DFU的型号
	Get CRC	0xAC	获取计算sector的CRC	Bootloader支持DFU的型号
	Get sLib Status	0xD2	获取sLib状态	Bootloader支持DFU的型号并且支持sLib的型号
DFU_DNLOAD	Write Memory	无	写数据到指定地址	Bootloader支持DFU的型号
	Set Address	0x21	设置读写地址	Bootloader支持DFU的型号
	Access Protect	0x82	开启访问保护	Bootloader支持DFU的型号
	Access Unprotect	0x92	关闭访问保护	Bootloader支持DFU的型号
	Erase and Program Protect	0x63	开启擦写保护	Bootloader支持DFU的型号
	Erase and Program Unprotect	0x73	关闭擦写保护	Bootloader支持DFU的型号
	Erase	0x41	擦除存储器	Bootloader支持DFU的型号
	Jump	0x18	跳转到指定地址执行	Bootloader支持DFU的型号
	Firmware CRC	0xAC	计算指定sector CRC	Bootloader支持DFU的型号
	Enable SPIM	0xB3	使能外部存储器	Bootloader支持DFU并且支持SPIM的型号
	Enable sLib	0xD0	使能sLib功能	Bootloader支持DFU并且支持sLib的型号
	Disable sLib	0xD1	解除sLib功能	Bootloader支持DFU并且支持sLib的型号
	Reset Device	0xD4	复位设备	除了AT32F403xx不支持

DFU请求	命令	代码	说明	支持型号
	Advanced Access Protect	0xD6	使能高级访问保护	Bootloader支持DFU并且支持高级访问保护的型号

3 DFU_UPLOAD 请求

DFU_UPLOAD请求根据参数wValue的值判断当前是读存储器还是Bootloader其它命令。

如下是DFU_UPLOAD请求设备和主机流程：

图 1 DFU_UPLOAD 请求设备端流程

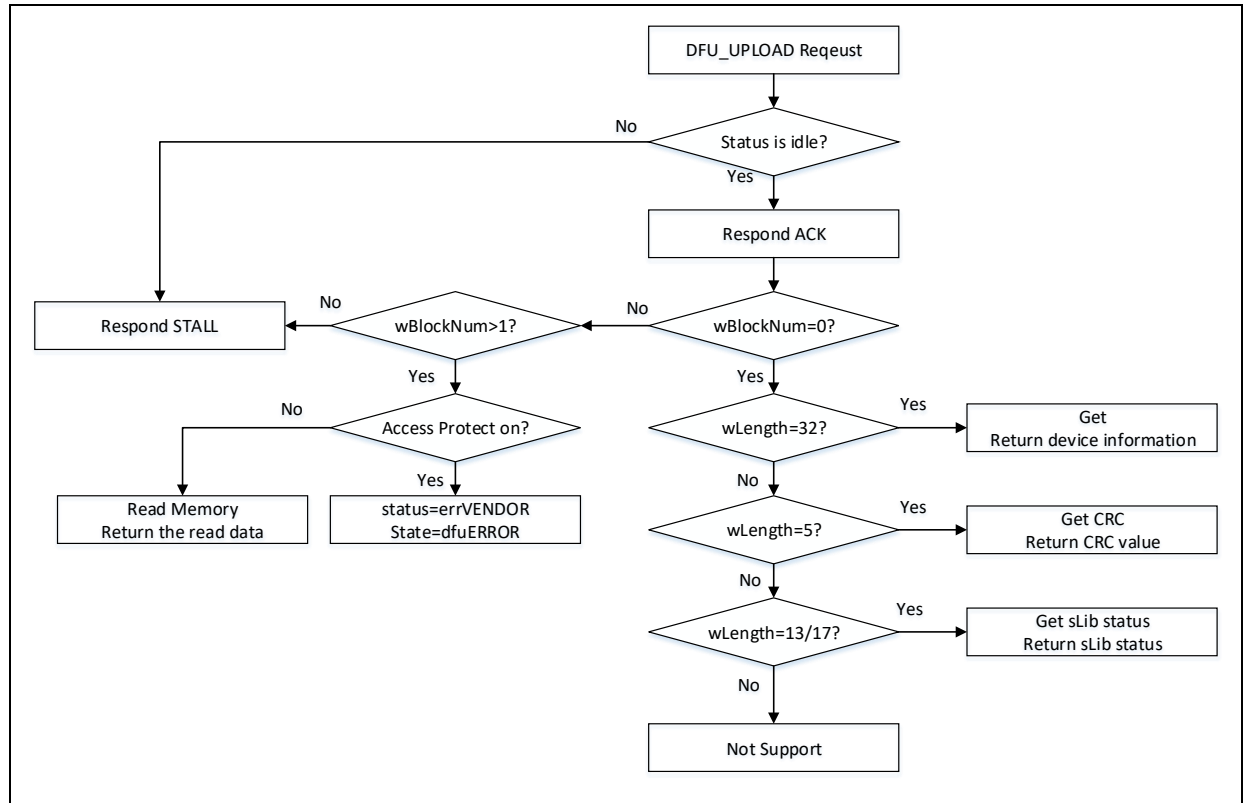
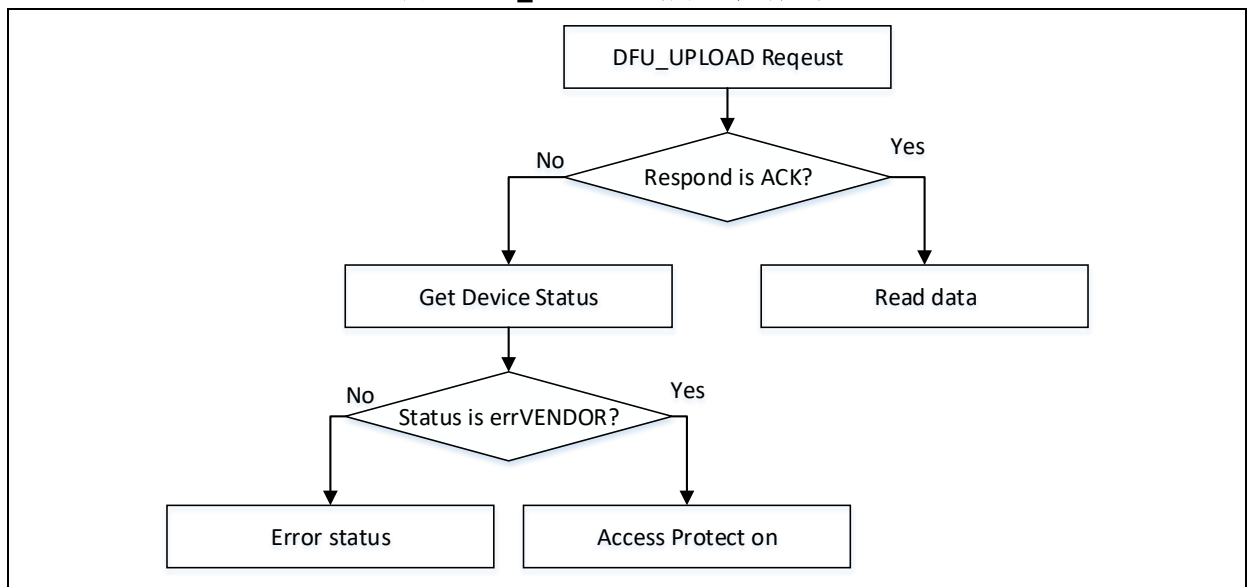


图 2 DFU_UPLOAD 请求主机端流程



3.1 Get

使用此命令时参数配置 wValue=0, wLength=32, 当访问保护开启之后, 此命令依然可以使用。

主机通过此命令获取设备支持的命令以及设备的基本信息, 设备的 Product ID 和 Project ID 固定第

12 byte~16 byte，上位机可通过 Product ID 和 Project ID 判断芯片型号，以确定芯片支持哪些命令。

注意：不同芯片支持的命令不同，Get 返回数据具体以芯片具体返回为准

表 4 Get 命令返回数据

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

3.2 Read Memory

当参数wValue > 1是表示要读存储器数据，当设备访问保护开启时，不能读取存储器数据。

主机通过发送一个有效的地址和长度（wLength），可以读取主存储器，片上SRAM以及用户系统数据。

- 从主存储器，片上SRAM，一次读取的字节数为2到2048.
- 用户系统数据读取大小必须与当前型号的微控制用户系统数据区大小相等。

注意：有效地址范围请参考对应型号的用户手册

读取的地址通过如下方式计算：

读取首地址 = (wValue - 2) * wlength + Address_Pointer

Address_Pointer通过DFU_DNLOAD请求的Set Address命令进行配置。

如果访问保护开启，将不允许读存储器，设备将返回错误状态 Status=dfuERROR, state=errVENDOR

3.3 Get CRC

使用此命令时参数配置 wValue = 0, wLength = 5，当设备访问保护开启时，依然可以使用此命令。

获取CRC用于校验固件的正确与否，CRC以Sector为单位进行计算，主机首先通过DFU_DNLOAD请求的Firmware CRC命令设置要计算的起始地址与Sector个数，计算完成之后通过DFU_UPLOAD请求的Get CRC命令获取计算的CRC。

Get CRC返回数据：

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

3.4 Get sLib Status 命令

使用此命令时参数配置 wValue = 0, wLength = 13/17，当设备访问保护开启时，不能使用此命令。

Get sLib status 命令用于获取当前 sLib 状态，此命令返回对应 sLib 寄存器值，寄存器值含义参考具体系列用户手册中关于 sLib 寄存器的说明。

AT32F435xx/AT32F437xx: (返回 17 bytes 数据)

当设备收到此命令后，会返回 4 字节的 SLIB_STS0 寄存器的值，4 字节的 SLIB_STS1 寄存器值，4 字节的 SLIB_STS2 寄存器值，4 字节 SLIB_MISC_STS 寄存器值。

其它系列: (返回 13 bytes 数据)

当设备收到此命令后，会返回 4 字节的 SLIB_STS0 寄存器的值，4 字节的 SLIB_STS1 寄存器值，4 字节 SLIB_MISC_STS 寄存器值。

注意：此命令在 AT32F403Axx, AT32F407xx, AT32F413xx 开启访问保护后不能使用，其它系列可以使用

AT32F435xx/AT32F437xx返回数据：

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)

Byte	Value	Description
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)

其它系列返回数据：

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 请求

DFU_DNLOAD 请求通过 wValue 的值和数据的第一个字节来确定是 Bootloader 哪条命令。

表 5 DNLOAD 命令条件

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

注意：在使用 DFU_DNLOAD 之前，主机先要检测设备端当前的状态是否在 *dfuIDLE* 或者 *dfuDNLOAD-IDLE*，如果不在这些空闲状态，主机需要发送 DFU_CLRSTATUS 请求然后在重新获取设备状态，直到设备状态为 *dfuIDLE* 状态。

如下是DFU_DNLOAD请求设备端和主机端流程：

图 3 DFU_DNLOAD 请求设备端流程

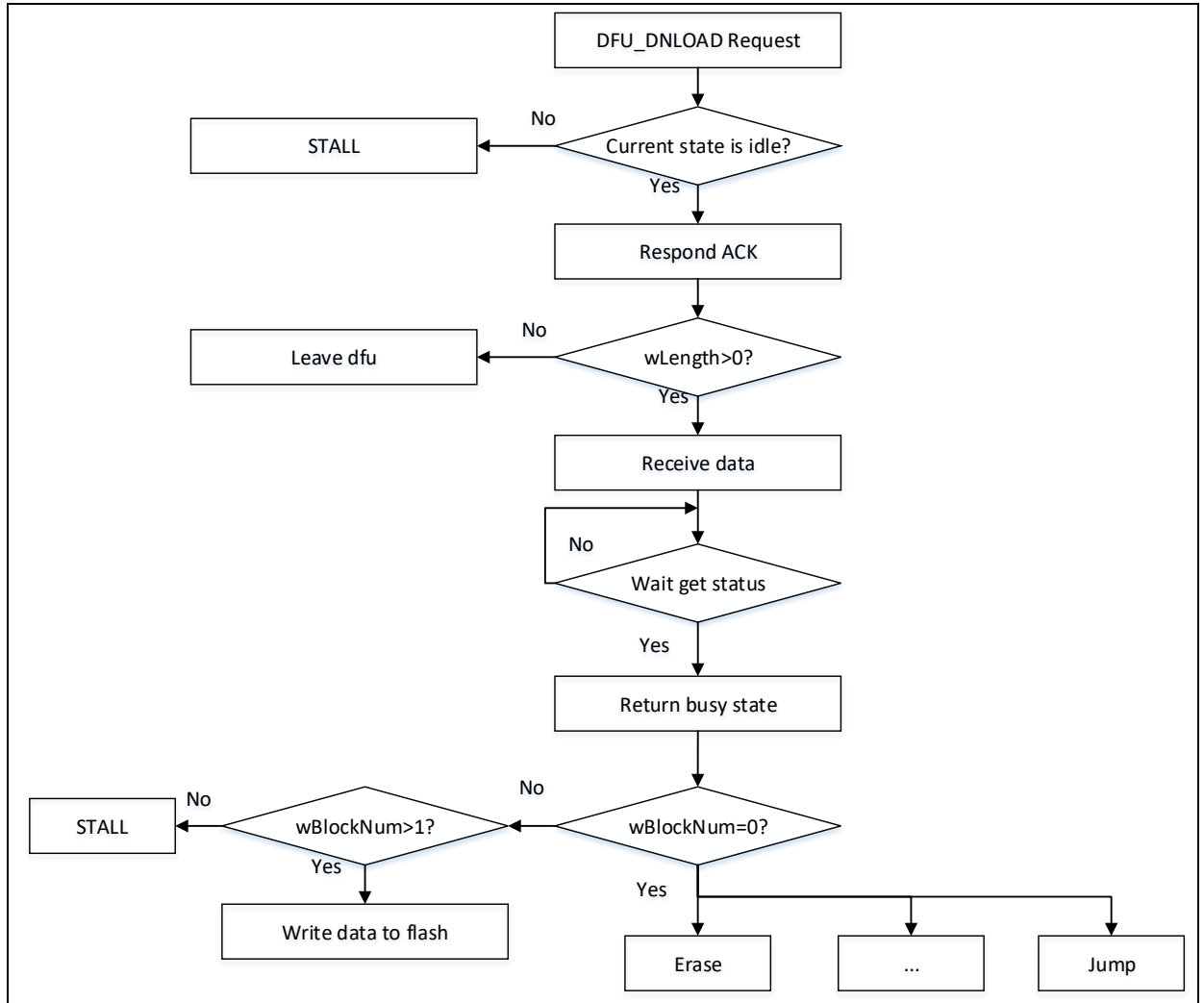
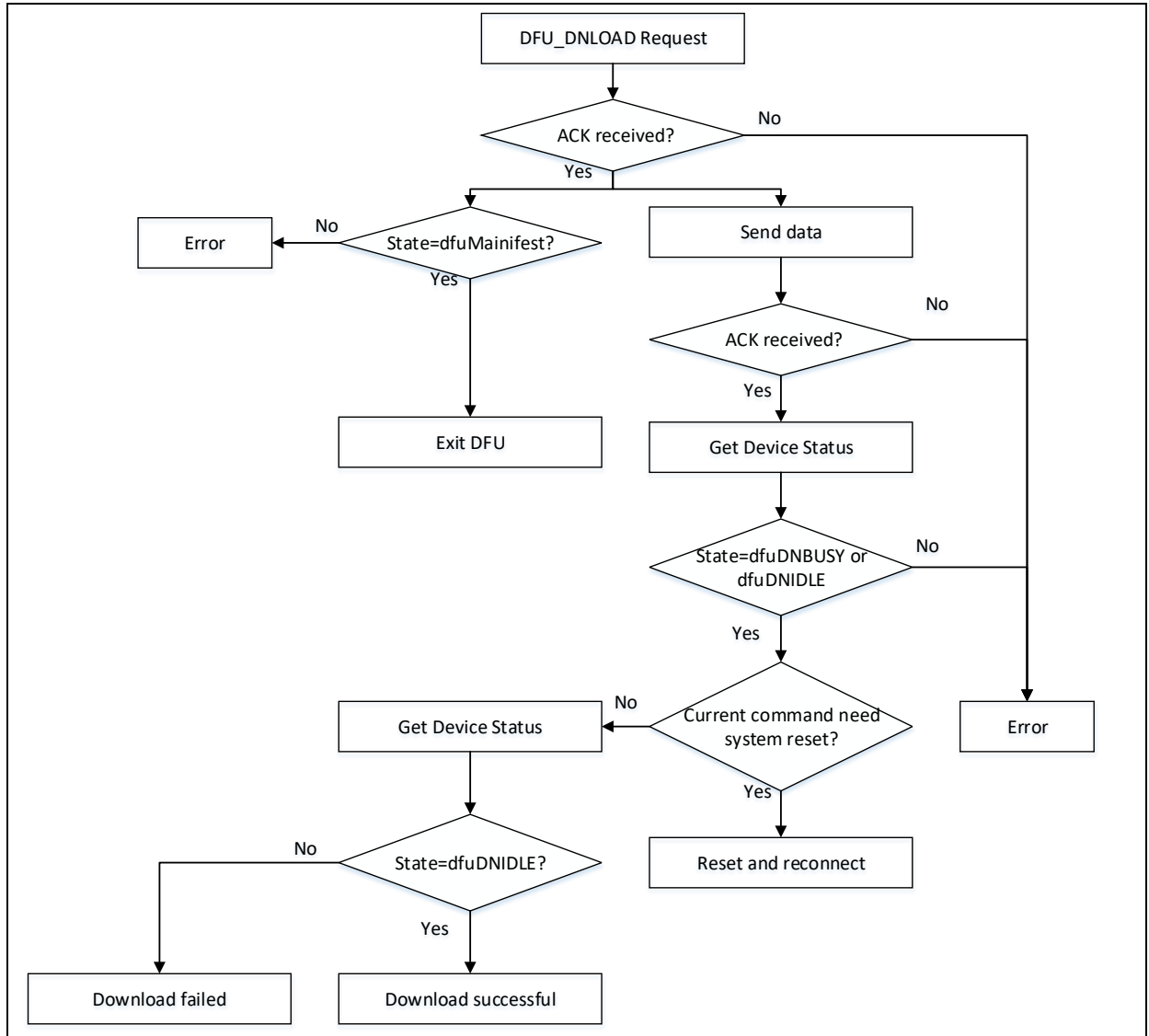


图 4 DFU_DNLOAD 请求主机端流程



4.1 Write Memory

当wValue > 1时表示Write Memory，设备访问保护开启时，不能使用此命令写存储器。

主机通过发送一个有效的地址(通过Set Address命令)和一个有效的数据长度（wLength）给设备，设备将数据写入到指定的地址。

写存储器支持的地址：

- 主闪存存储器和片上SRAM，一次支持写入2到2048个字节
- 用户系统数据，大小必须是用户系统数据区域的大小相等

注意：有效地址范围请参考对应芯片型号的说明文档

主机发送完写存储器和数据之后，需要发送DFU_GETSTATUS命令去让设备执行写操作，此时设备会返回状态为dfuDNBUSY（表示正在写数据到存储器），如果主机发送地址区域是选择字节，在数据写完之后，设备将会自动执行System Reset，此时上位机需要重新连接设备。

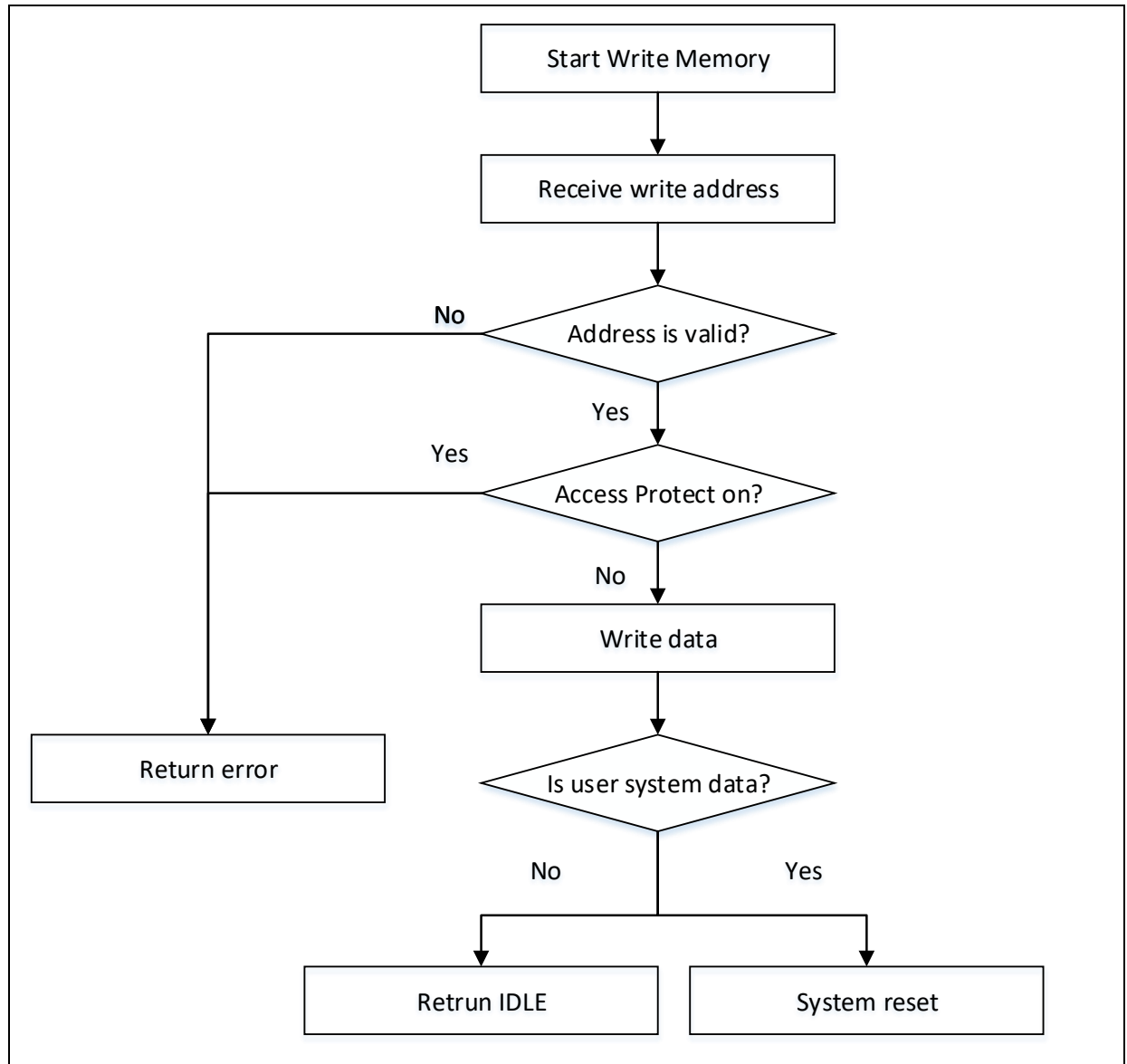
主机选择第二再次发送DFU_GETSTATUS命令去查询设备写存储器是否完成，如果发送是无效的地址，设备的状态会返回status=dfuERROR, state =STATUS_ERRADDRESS。

写数据地址通过如下计算：

写数据首地址 = (wValue - 2) * wlength + Address_Pointer

如果设备开始了访问保护，写存储器返回的状态status=dfuERROR, state=errVENDOR。

图 5 Write Memory 设备端流程



4.2 Set Address

当wValue=0 并且第一个字节值为0x21时表示Set Address命令，当设备访问保护开启时，不能使用此命令。

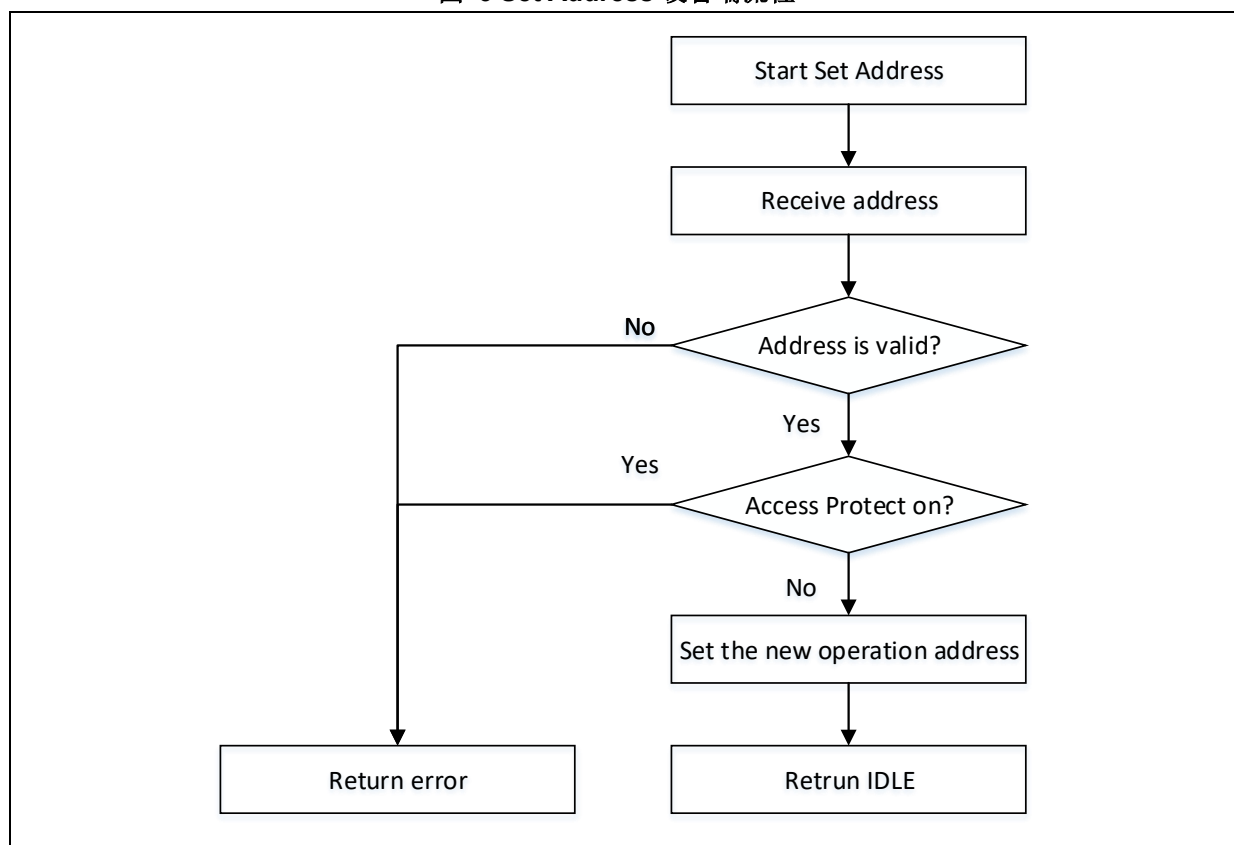
Write Memory命令和Read Memory命令在使用之前都需要用Set Address命令设置一个写或读的起始地址Address_Pointer，主机发送1字节设置地址命令加上4字节的地址。

主机发送设置地址命令之后，需要发送DFU_GETSTATUS请求触发设备执行设置地址操作，此时设备会返回状态为STATE_dfuDNLOAD_IDLE（表示设置完成）。

主机发送数据：

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

图 6 Set Address 设备端流程



4.3 Access Protect

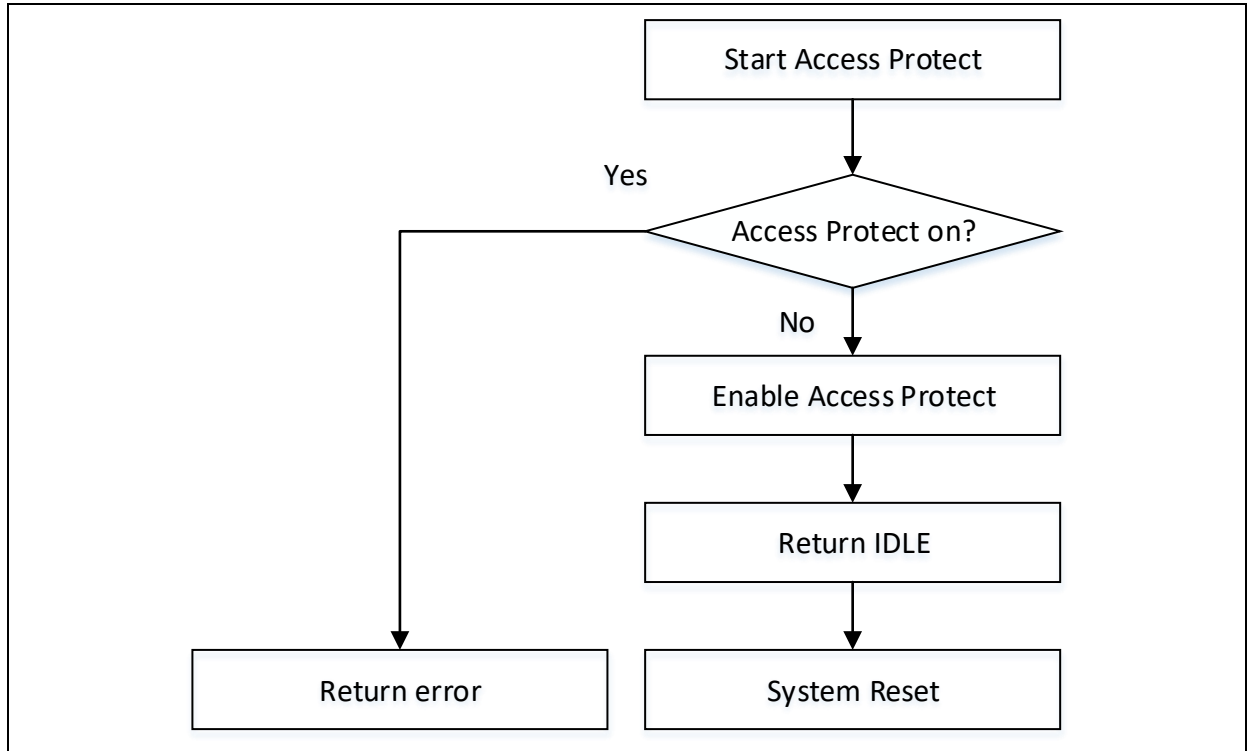
当wValue=0 并且第一个字节值为0x82时表示开启访问保护，当设备访问保护已经开始时，不能使用此命令。

主机发送1字节使能访问保护命令0x82之后，需要发送DFU_GETSTATUS请求触发设备执行使能访问操作。此时设备会返回STATE_dfuDNLOAD_IDLE。之后设备会执行系统复位，访问保护生效，此时主机需要重新连接设备。

主机发送数据：

Byte	Value	Description
0	0x82	Access Protect

图 7 Access Protect 设备端流程



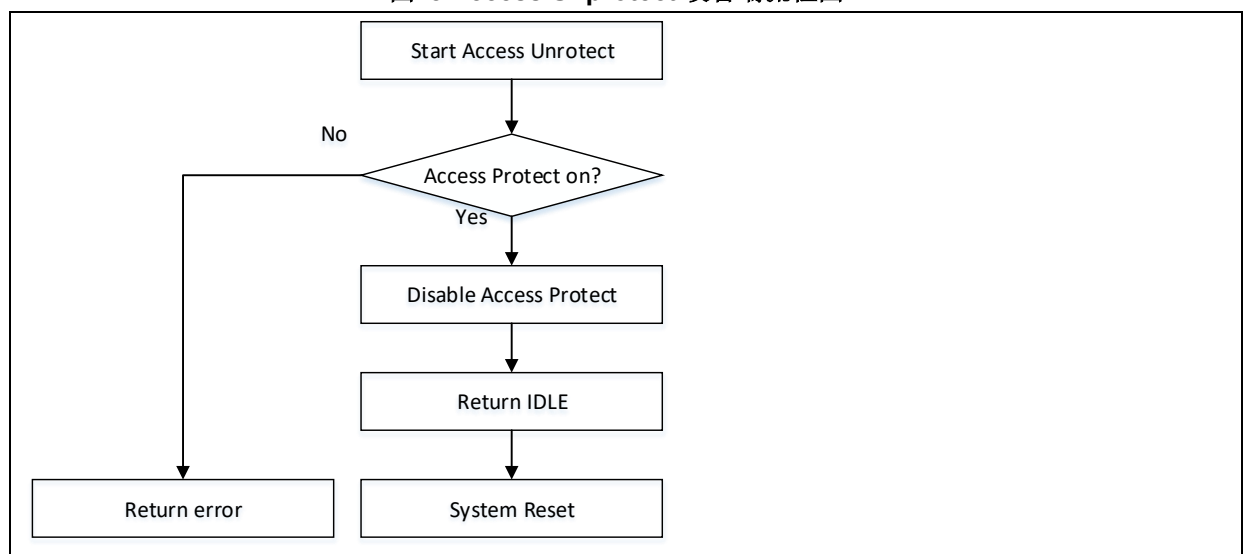
4.4 Access Unprotect

当wValue=0 并且第一个字节值为0x92时表示解除访问保护，当访问保护开始时，可以使用此命令。主机发送1字节解除访问保护命令0x92之后，需要发送DFU_GETSTATUS请求触发设备执行解除访问操作，此时设备会返回STATE_dfuDNLOAD_IDLE，系统会自动执行擦除主存储器数据，之后设备会执行系统复。

主机发送数据：

Byte	Value	Description
0	0x92	Access Unprotect

图 8 Access Unprotect 设备端流程图



4.5 Erase and Program Protect

当wValue=0 并且第一个字节值为0x63时表示设置擦写保护，当访问保护开始时，不能使用此命令。

主机需传送1个字节擦写保护命令，1个字节写保护Sector个数（对于435/437，发送的sector个数为实际个数减1，其它系列直接发送sector个数），加上n个字节Sector index，设备会按照主机端发送的Sector个数与Sector index写选择字节。

主机发送写保护命令之后，需要发送DFU_GETSTATUS请求触发设备执行使能写保护操作。设备执行命令成功之后会自动执行系统复位。

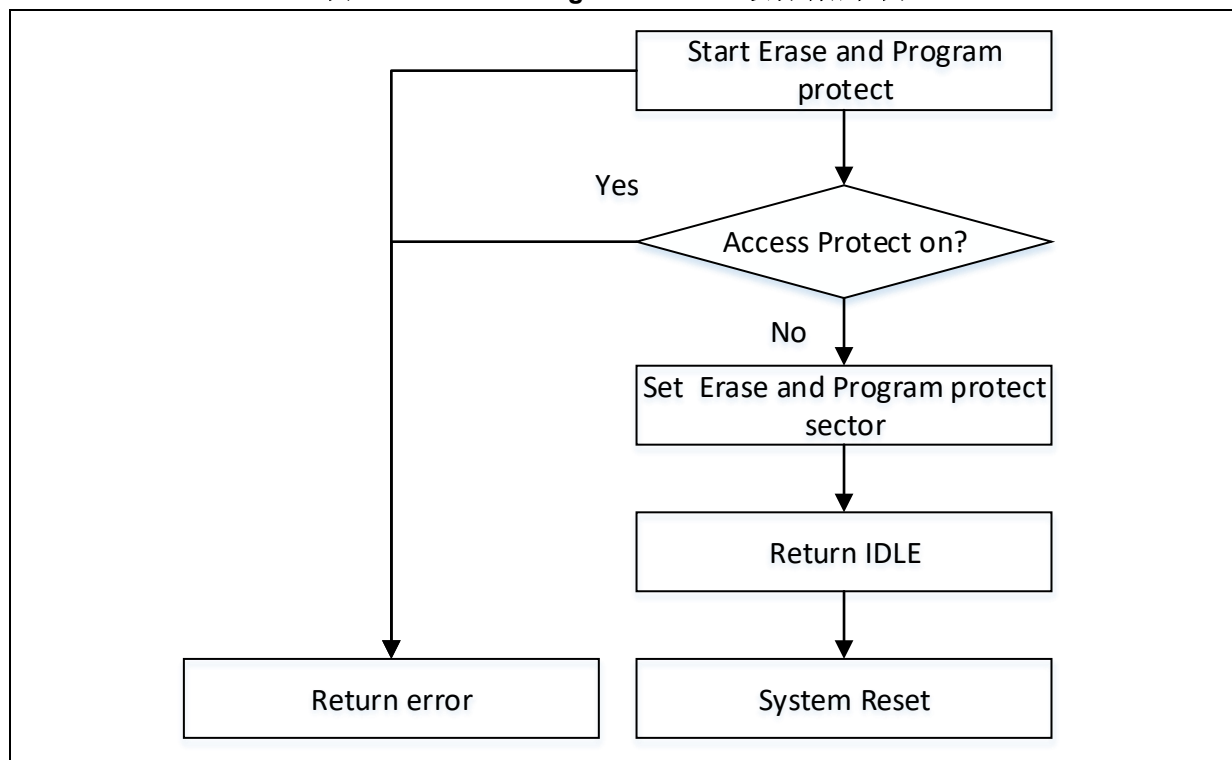
注意：擦写保护中的index bit 取值为(0,1,2...n),对应用户系统数据区中擦写保护字节(0-N)的bit位，更多详细内容可参考具体型号用户手册中的用户系统数据说明。

注意：对于435/437，发送的sector个数为实际个数减1，其它系列直接发送sector个数。

主机发送数据：

Byte	Value	Description
0	0x63	Erase and Program Protect
1	*	Number of sector index(435/437 need - 1)
2	*	Sector index 0
3	*	Sector index1
4

图 9 Erase and Program Protect 设备端流程图



4.6 Erase and Program Unprotect

当wValue=0 并且第一个字节值为0x73时表示设置解除擦写保护，当设备开启访问保护时，不能使用此命令。

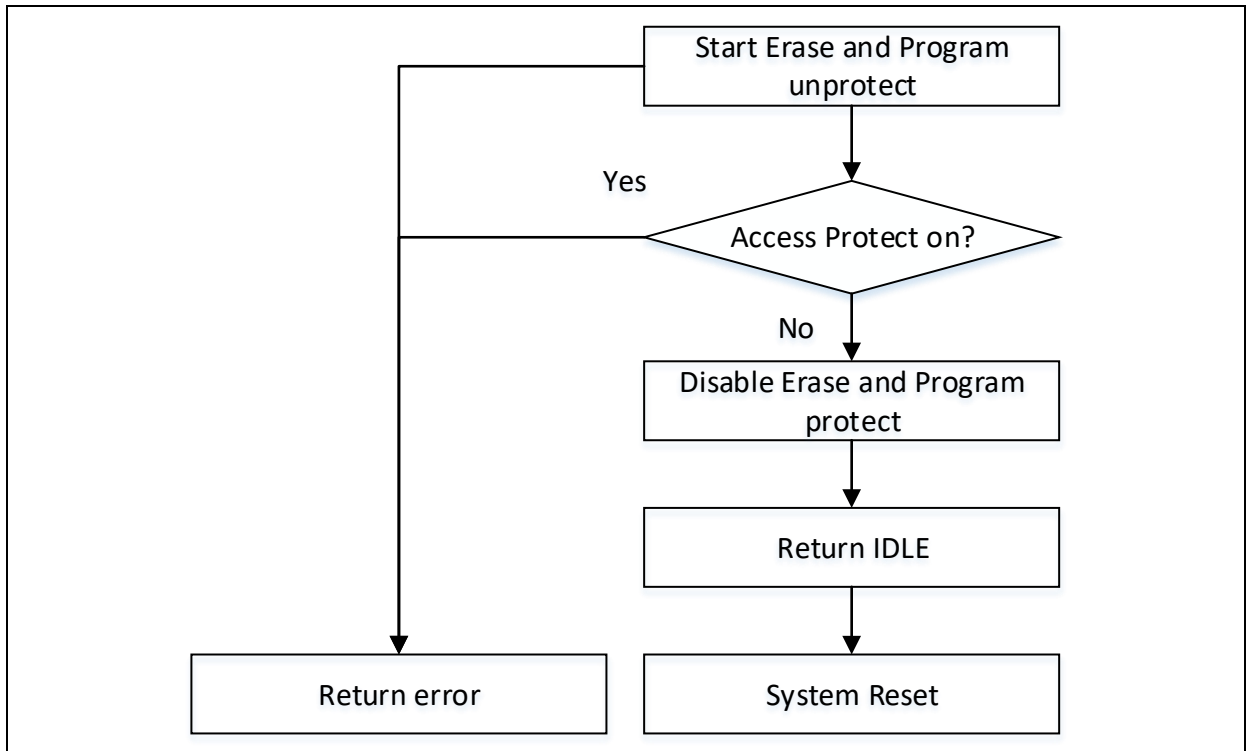
此命令用于解除存储器所有Sector的擦写保护，主机只需要传送1字节的解除擦写保护命令。

主机发送解除擦写保护命令0x73之后需要发送DFU_GETSTATUS请求触发设备执行解除擦写保护操作，设备执行解除擦写保护然后返回STATE_dfuDNLOAD_IDLE。之后设备会执行系统复位。

主机发送数据：

Byte	Value	Description
0	0x73	Erase and Program Unprotect

图 10 Erase and Program Unprotect 设备端流程图



4.7 Erase

当wValue=0 并且第一个字节值为0x41时表示设置擦除命令，当访问保护开启时，不能使用此命令。

此命令支持按照 Sector 擦除（Sector 大小根据具体型号有区别），全擦除等操作，对于存在 Bank2 的型号还支持 Bank1 擦除和 Bank2 擦除，支持 SPIM 的型号同时支持 Bank3 的擦除，擦除的地址必须是有效范围内的地址，各个型号的有效地址范围不同。

- 全擦除，wLength=1，只需要传输1字节擦除命令。
- Sector擦除，wLength=5，主机需传送1字节擦除命令和4字节擦除地址（需按sector对齐）。
- Bank擦除，wLength=3，主机需传送1字节擦除命令和2字节Bank index
- Block擦除，wLength=7，主机需传送1字节擦除命令,2字节的Block index和4字节擦除地址。

注意：AT32F435xx/AT32F437xx 支持 Block 擦除,Block 按照 64Kbyte 对齐。

表 6 擦除索引表

擦除类型	擦除索引 (index)	擦除位置
全擦除	0xFF 0xFF	All Flash
Bank 擦除	0xFF 0xFE	Bank1 Erase
	0xFF 0xFD	Bank2 Erase
	0xFF 0xFC	Bank3 Erase
	0xFF 0xFB	Block Erase

主机发送擦除命令之后，需要发送DFU_GETSTATUS请求，如果此时返回的状态不是dfuDNBUSY，此时可能是擦除错误，如无效的地址，或者设备开启了访问保护等。设备用DFU_GETSTATUS请求获取设备状态，直到返回的状态是dfuDNLOAD_IDLE表示擦除完成。

全擦除主机发送数据 wLength=1:

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

Sector擦除主机发送数据 wLength=5:

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

Bank擦除主机发送数据 wLength=3:

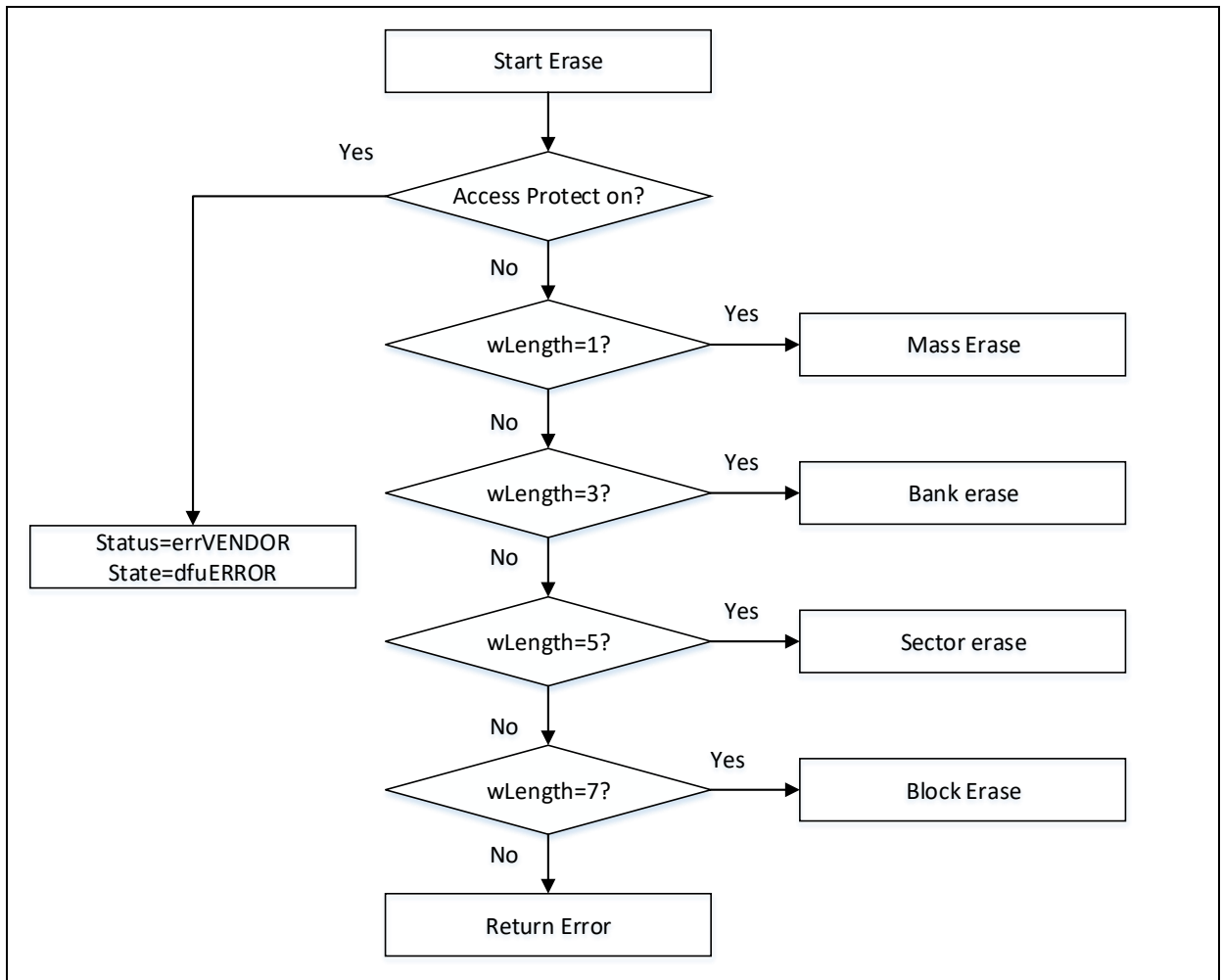
Byte	Value	Description
0	0x41	Erase
1	0xFF	Bank index
2	*	Bank index (0xFE 0xFD 0xFC)

Block擦除主机发送数据 wLength=7:

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

Byte	Value	Description
6	*	Address (MSB)

图 11 Erase 设备端流程图



4.8 Jump

当wValue=0 并且第一个字节值为0x18时表示跳转到用户程序执行的命令，当设备开启访问保护时，不能使用此命令。

此命令用来跳转到指定地址执行，主机需传送1字节的跳转命令和4字节有效的跳转地址。

主机发送跳转命令之后，需要发送DFU_GETSTATUS请求。此时设备返回状态dfuDNLOAD_IDLE表示即将跳转到用户代码执行。

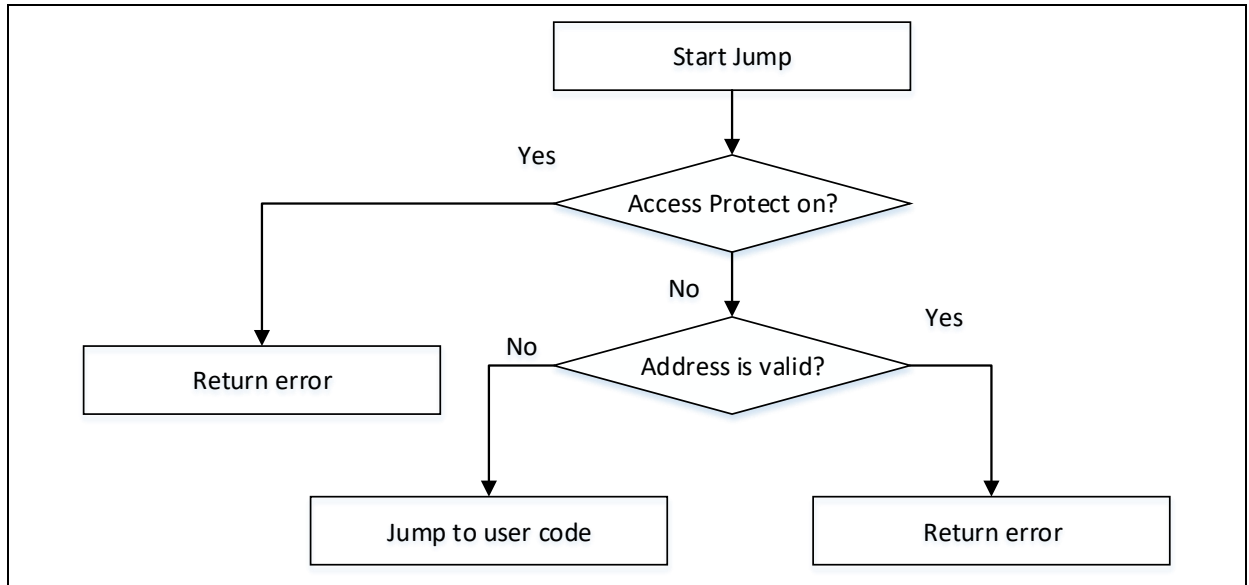
注意：有效地址包括SRAM地址，Flash地址，SPIM(BANK3)地址，具体地址范围请参考对应芯片型号说明文档。

主机发送数据：

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

Byte	Value	Description
4	*	Address
5	*	Address (MSB)

图 12 Jump 设备端流程图



4.9 Firmware CRC

当wValue=0 并且第一个字节值为0xAC时表示CRC计算的命令，当设备访问保护开启时，可以使用此命令。

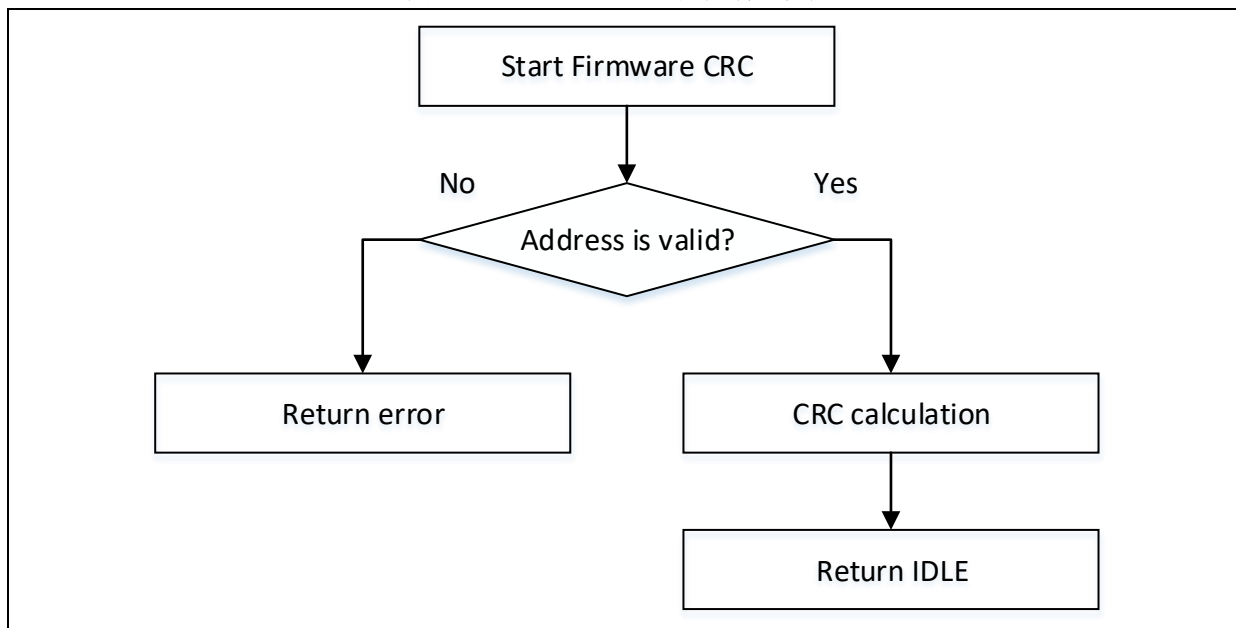
CRC按照Sector为单位进行计算，主机需传送1字节CRC计算命令，4字节的起始地址（需要Sector对齐）和2字节Sector个数。

主机发送命令之后，需要发送DFU_GETSTATUS请求。如果设备返回dfuDNLOAD_IDLE表示设备计算完成，此时需要使用DFU_UPLOAD里面的Get CRC的命令来获取当前计算的CRC值。

主机发送数据：

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)

图 13 Firmware CRC 设备端流程图



4.10 Enable SPIM

当wValue=0 并且第一个字节值为 0xB3时表示Enable Bank3的命令，当设备访问保护开启时，可以使用此命令。

使用此命令时需要支持 SPIM Flash type，Flash size，以及 Flash 的加密范围，详细信息可参考支持 SPIM 的用户手册。

Flash type list:

- 0x90: 一般 Flash
Dummy cycle 为 4
- 0x91: 一般 Flash，加上 Quad Enable 设定
Dummy cycle 为 4
以 Volatile 格式启用 Quad Enable

Flash size 支持的上限为 16M Bytes，设备端会根据主机传入的 Flash size 进行后续擦除与读写。

为了防止 SPIM Flash 数据直接被读出，SPIM 提供加密功能，Flash controller 写入 SPIM 时可写加密数据，读出时对应解密。对 MCU 而言，读写数据任是明文，但存储在外部 Flash 的数据，是经过加密后的数据。用户可以自行设定加密密码和加密范围，加密密码位于用户系统数据区，在写入 SPIM(bank3)要设定完成。

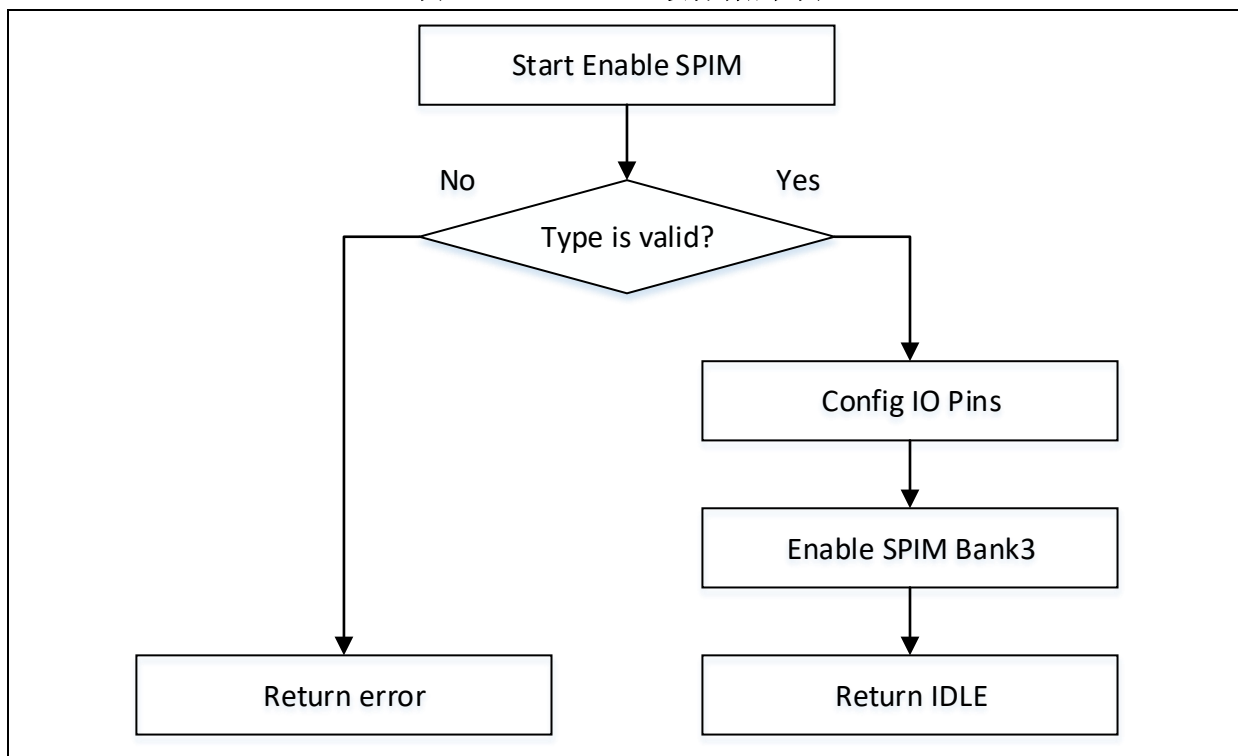
加密范围（FLASH_FDA）是每次启用 SPIM(bank3)时要提供的设定，用来指定从 SPIM(bank3)起始地址开始有多少 Bytes 需要被加密，当设定大于 16M Bytes 时，表示 SPIM(bank3)全部加密，当设定为 0 时，表示 SPIM（bank3）全部不加密。

主机发送数据：

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

Byte	Value	Description
6	*	Flash FDA (LSB)
7	*	Flash FDA
8	*	Flash FDA
9	*	Flash FDA (MSB)

图 14 Enable SPIM 设备端流程图



4.11 Enable sLib

当wValue=0 并且第一个字节值为 0xD0时表示Enable sLib的命令，当访问保护开启时，不能使用此命令。

使用此命令需要4字节sLib密码，2字节的sLib起始sector，2字节的sLib数据/指令起始sector， 2字节的sLib结束sector，sLib的设定需要系统复位之后才能生效。

主机发送Enable sLib命令之后，需要发送DFU_GETSTATUS请求触发设备执行使能sLib的操作。sLib的配置需要Reset才会生效，因此在使能sLib，下载数据到存储器之后，需要调用复位设备命令。

注意：1. 更多关于SLIB使用方法请参考《安全库区(SLIB)使用指南》

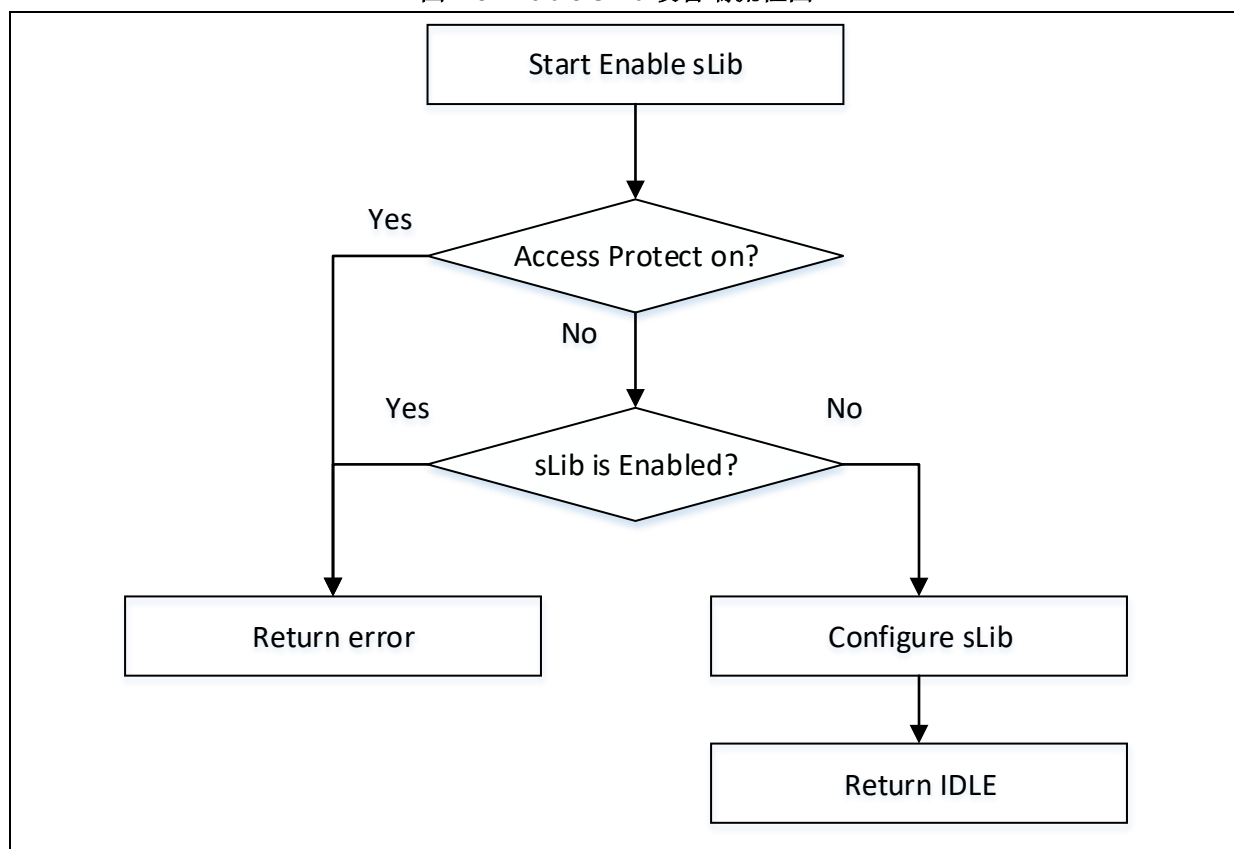
2. 在调用使能SLIB之前，需先获取sLib状态，如果sLib已经开启，则需要先解除sLib之后才能再次使能SLIB功能。

主机发送数据：

Byte	Value	Description
0	0xD0	Enable sLib
1	*	Password (LSB)
2	*	Password

Byte	Value	Description
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)

图 15 Enable sLib 设备端流程图



4.12 Disable sLib

当wValue=0 并且第一个字节值为 0xD1时表示Disable sLib的命令，当访问保护开启时，不能使用此命令。

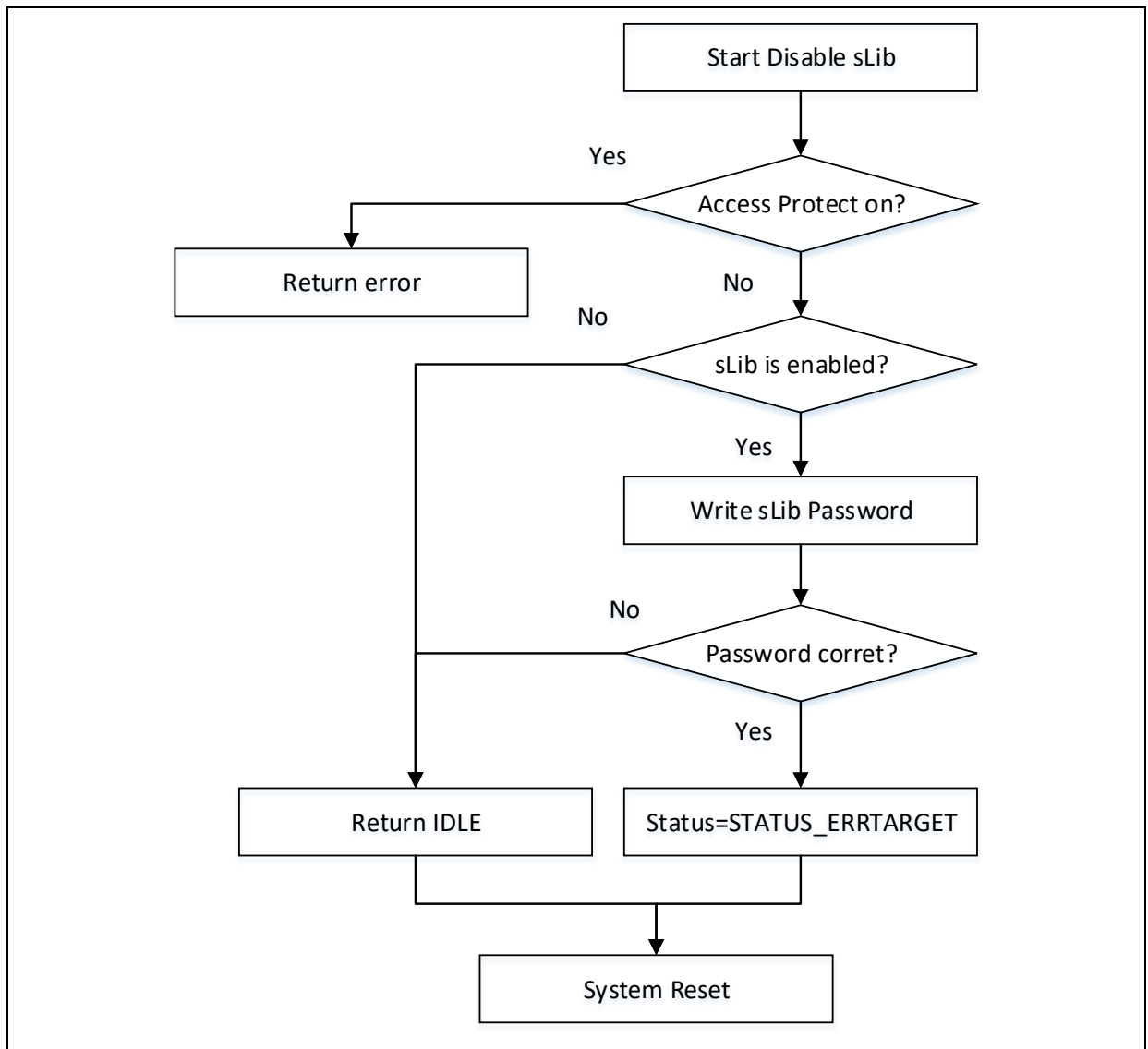
Disable sLib需要主机传送1字节解除sLib命令和4字节Password，如果Password正常，则设备会解除sLib并执行System Reset，如果Passwrod错误，返回错误状态status= STATUS_ERRTARGET。

主机发送Disable sLib之后，需要发送DFU_GETSTATUS请求触发设备支持解除sLib操作。不管密码是否正确，调用Disable sLib之后，设备端会自动执行System Reset，主机端需要重新连接。

主机发送数据：

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

图 16 Disable sLib 设备端流程图



4.13 Reset Device

当wValue=0 并且第一个字节值为 0xD4时表示Reset Device的命令，当设备访问保护开启时，不能使用此命令

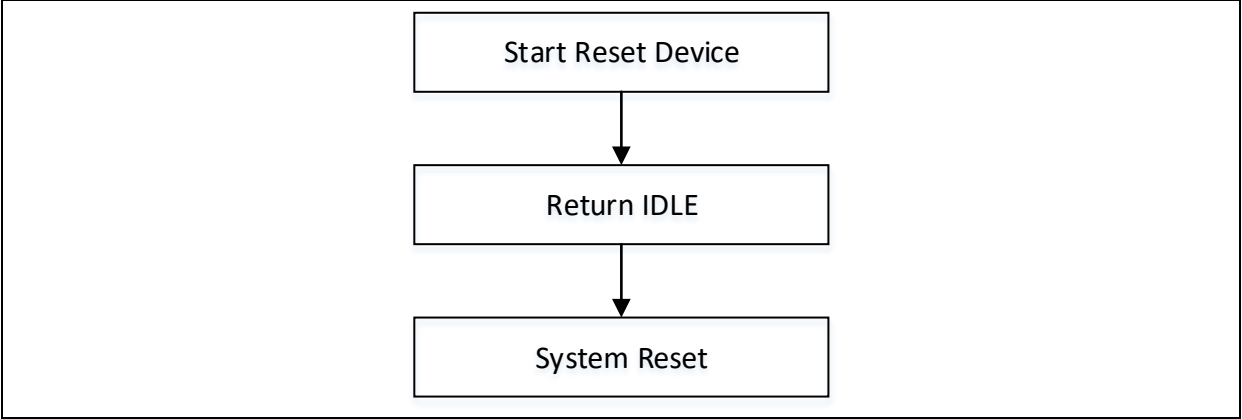
此命令用于主机在某些操作完成之后，需要让设备端复位时使用。

主机发送Reset Device之后，需要发送DFU_GETSTATUS请求触发设备端执行复位设备。

主机发送数据：

Byte	Value	Description
0	0xD4	Reset Device

图 17 Reset Device 设备端流程图



4.14 Advanced Access Protect

当 wValue=0 并且第一个字节值为 0xD6 时表示设置使能高级访问保护的命令，如果设备访问保护开启时，不能使用此命令，高级读保护功能具体功能请参考对应型号用户手册中对高级读保护的说明。

注意：高级访问保护在部分型号上不能再被解除，具体参考对应型号用户手册。

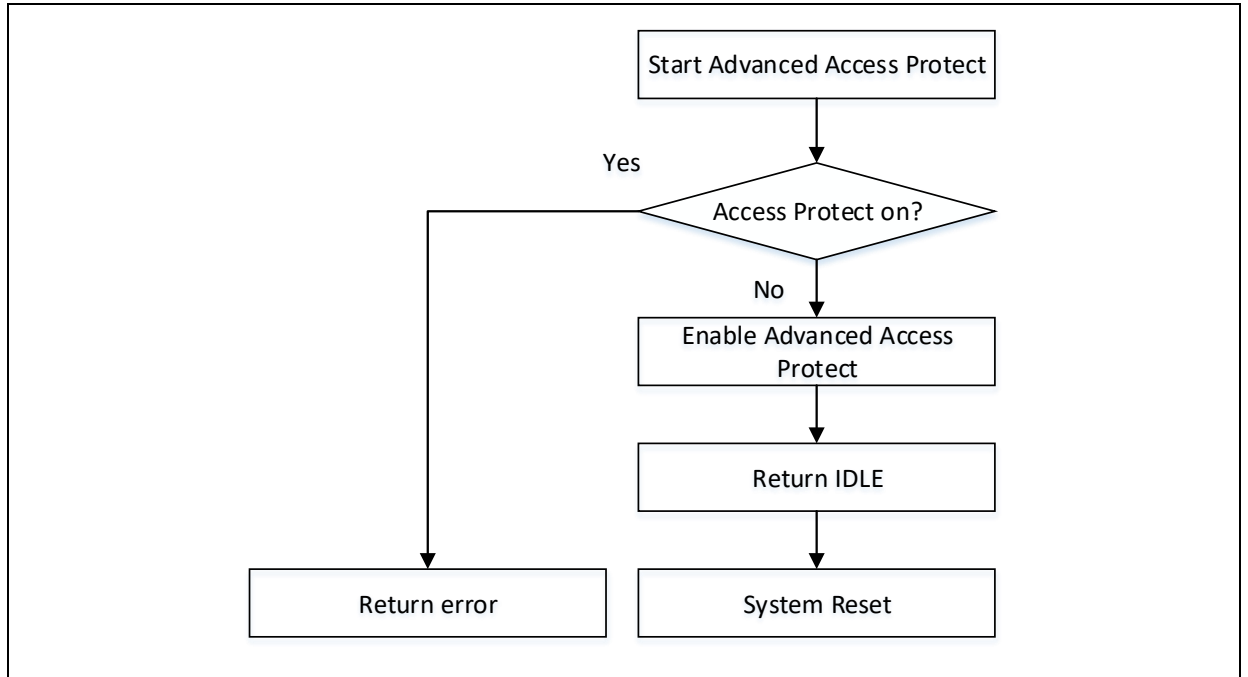
使用此命令主机需传送 1 字节使能高级访问保护命令和 1 字节标志。

主机发送使能高级访问保护命令之后，需要发送DFU_GETSTATUS请求触发设备执行使能高级访问保护，之后设备会执行System Reset，主机需要重新连接设备。

主机发送数据：

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

图 18 Advanced Access Protect 设备端流程图



5 版本历史

表 7 文档版本历史

日期	版本	变更
2021.12.07	2.0.0	最初版本
2023.01.05	2.0.1	1. 添加AT32F423的支持 2. 修改435/437 Erase and Program Protect 命令中sector个数需要减1描述
2023.06.30	2.0.2	1. 添加AT32F405/402支持 2. 添加AT32A403A支持
2024.03.20	2.0.3	1. 添加AT32A423支持
2024.04.28	2.0.4	1. 删除型号支持列表 2. 删除命令支持的型号列表

重要通知 - 请仔细阅读

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有过任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和 / 或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：（A）对安全性有特别要求的应用，例如：生命支持、主动植入设备或对产品功能安全有要求的系统；（B）航空应用；（C）航天应用或航天环境；（D）武器，且/或（E）其他可能导致人身伤害、死亡及财产损害的应用。如果采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险及法律责任仍将由采购商单独承担，且采购商应独力负责在前述应用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和 / 或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

。

© 2024 雅特力科技 保留所有权利