PM0005
User Manual

AT32 Bootloader UART Protocol

Introduction

This guideline describes the UART protocol used in the AT32 microcontroller bootloader, including the supported commands and how to use this protocol.

Applicable products:

MCUs	AT32F403xx
	AT32F413xx
	AT32F415xx
	AT32F403Axx
	AT32F407xx
	AT32F421xx
	AT32F435xx
	AT32F437xx
	AT32F425xx

Contents

List of tables

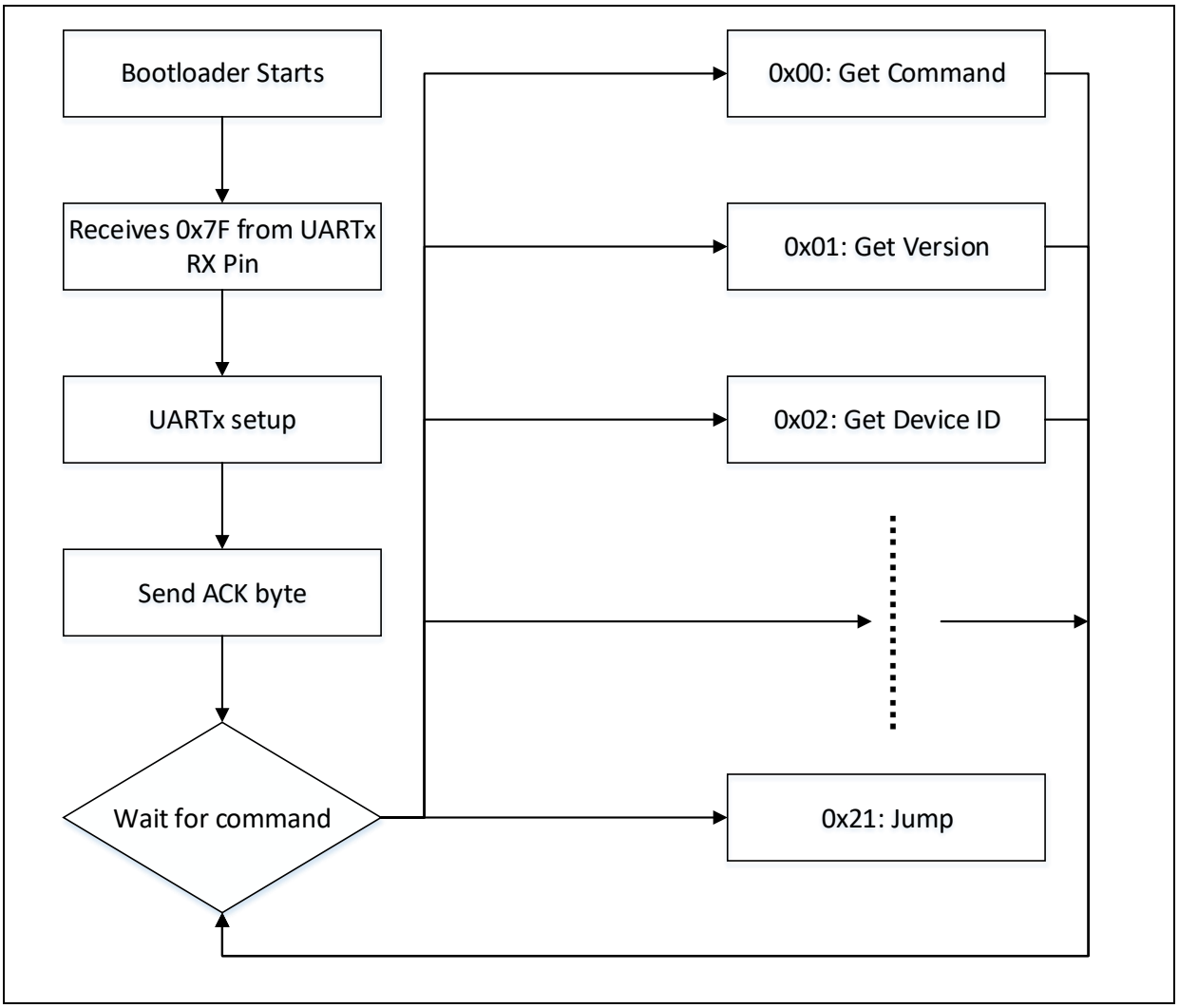
List of figures

1 UARTx boot mode

While using UART boot mode, the bootloader, once enabled, starts detecting the UARTx_Rx pin and waits to receive 0x7F data frame: a start bit, 0x7F data bit, even parity bit and a stop bit.

The data frame is measured using the Systick timer. And the value of the timer is used to calculate the baud rate of UARTx. An ACK byte is transmitted to the host 0x79, indicating that the connection is successful and the microcontroller is ready to receive commands from the host.

Figure 1 Bootloader Main Loop



2 UARTx baud rate

Baud rate settings:

AT32 Bootloader supports auto detection of baud rate, with a deviation lower than 2.5%. The minimum baud rate is 1200, while the maximum one is 256000.

Deviation rate=$\left|\frac{Device\ \ baud\ \ rate-Host\ \ baud\ \ rate}{Device\ \ baud\ \ rate}\right| \times 100\%$. Error is less than 2.5%

Baud rate detection through bootloader:

After the device is activated, the host sends 0x7F data to the device, including 1 start bit, 8 data bit (0x7F), even parity bit and 1 stop bit.

After detecting the data sent by the host, the device will set the corresponding UARTx baud rate and send 0x79 to the host. If the host received 0x79, it signals that the connection between the device and the host is successful and the host will continue issuing commands.

ACK = 0x79

NACK = 0x1F

3 Bootloader commands

The number of commands supported depends on the microcontrollers (due to their functionality difference). However, all of MCUs are compatible for the same commands, such as, read, write, erase and so on.

The supported commands are shown in Table 2 below.

Command	Value	Description	Applicable products
Set ISP	0xFA	Set host identification number	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx
Get Commands(1)	0x00	Get device command list	AT32F403xx, AT32F413xx(1), AT32F415xx(1), AT32F403Axx(1), AT32F407xx(1), AT32F421xx(1), AT32F435xxm AT32F437xx, AT32F425xx
Get Version	0x01	Get bootloader version	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Get Device ID(1)	0x02	Get device ID	AT32F403xx, AT32F413xx(1), AT32F415xx(1), AT32F403Axx(1), AT32F407xx(1), AT32F421xx(1), AT32F435xx, AT32F437xx, AT32F425xx
Read memory	0x11	Read memory at a given address	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Jump	0x21	Jump to a given memory address	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Write memory	0x31	Write to memory at a given address	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Erase	0x44	Erase memory	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Erase and program protect	0x63	Erase/program protection enable	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Erase and program unprotect	0x73	Erase/program protection disable	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Access Protect	0x82	Access protection enable	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Access Unprotect	0x92	Access protection disable	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Firmware CRC	0xAC	Calculate sector CRC	AT32F403xx, AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx

Command	Value	Description	Applicable products
Enable SPIM(bank3)	0xB3	SPIM enable	AT32F403xx, AT32F413xx, AT32F403Axx, AT32F407xx
Enable sLib	0xD0	sLib enable	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Disable sLib	0xD1	sLib disable	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Get sLib Status	0xD2	Get sLib status	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
SPIM Remap	0xD3	SPIM IO pin remap	AT32F413xx, AT32F403Axx, AT32F407xx
Reset Device	0xD4	Devise reset	AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, AT32F435xx, AT32F437xx, AT32F425xx
Advanced Access Protect	0xD6	Advanced access protection enable	AT32F415xx, AT32F421xx, AT32F425xx

Note: For AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421xx, a SET ISP command must be sent before enabling Get Commands and Get Device ID. If a Set ISP command is sent to a device that does not support it, the device issues NACK signal. In this case, the host could ignore NACK and continue subsequent operations.

4 Bootloader commands

This section gives a description of each Bootloader command and of how to use it in detail.

4.1 Set ISP

Set ISP command is used to identify a host. For the microcontrollers supporting this command, it is necessary to send Set ISP command before Get Commands and Get Device ID execution. This requirement has no effect on those devices that do not support Set ISP command. For the sake of compatibility, a host can handle this command in two ways:

- Start Set ISP command by host, issue ACK response by device, send a 4-byte data (fixed 0x02, 0x03, 0x54, 0x41) and 1-byte checksum by host, issue an ACK by device, and this command finishes, and it is ready to start the next command.
- Start Set ISP command by host, issue a NACK by device, then this command finishes, and it is ready to start the next command.

This command can also be used even if access protection is enabled.

4.1.1 Bootloader command flow chart

Figure 2 Set ISP flow chart on host side

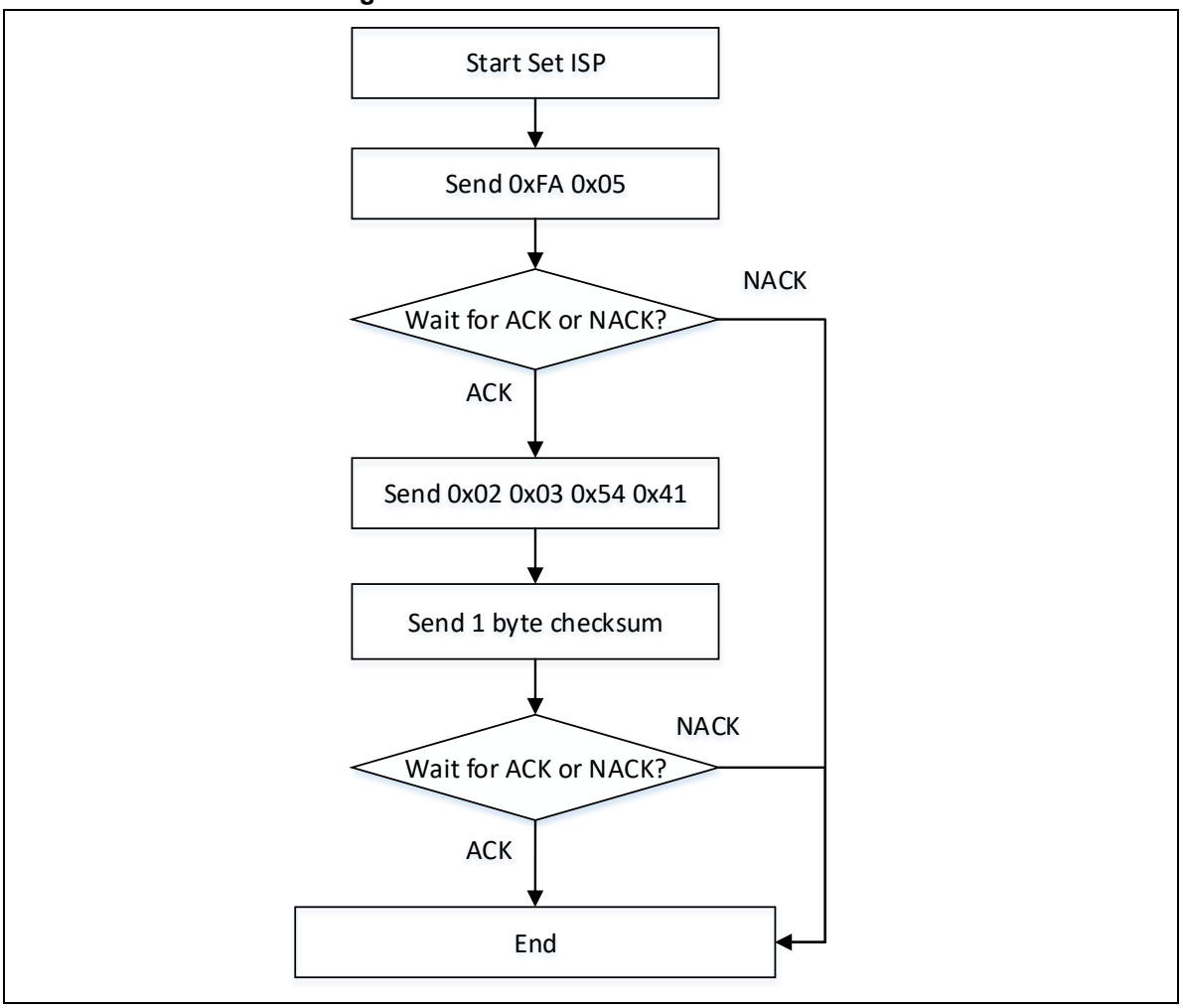
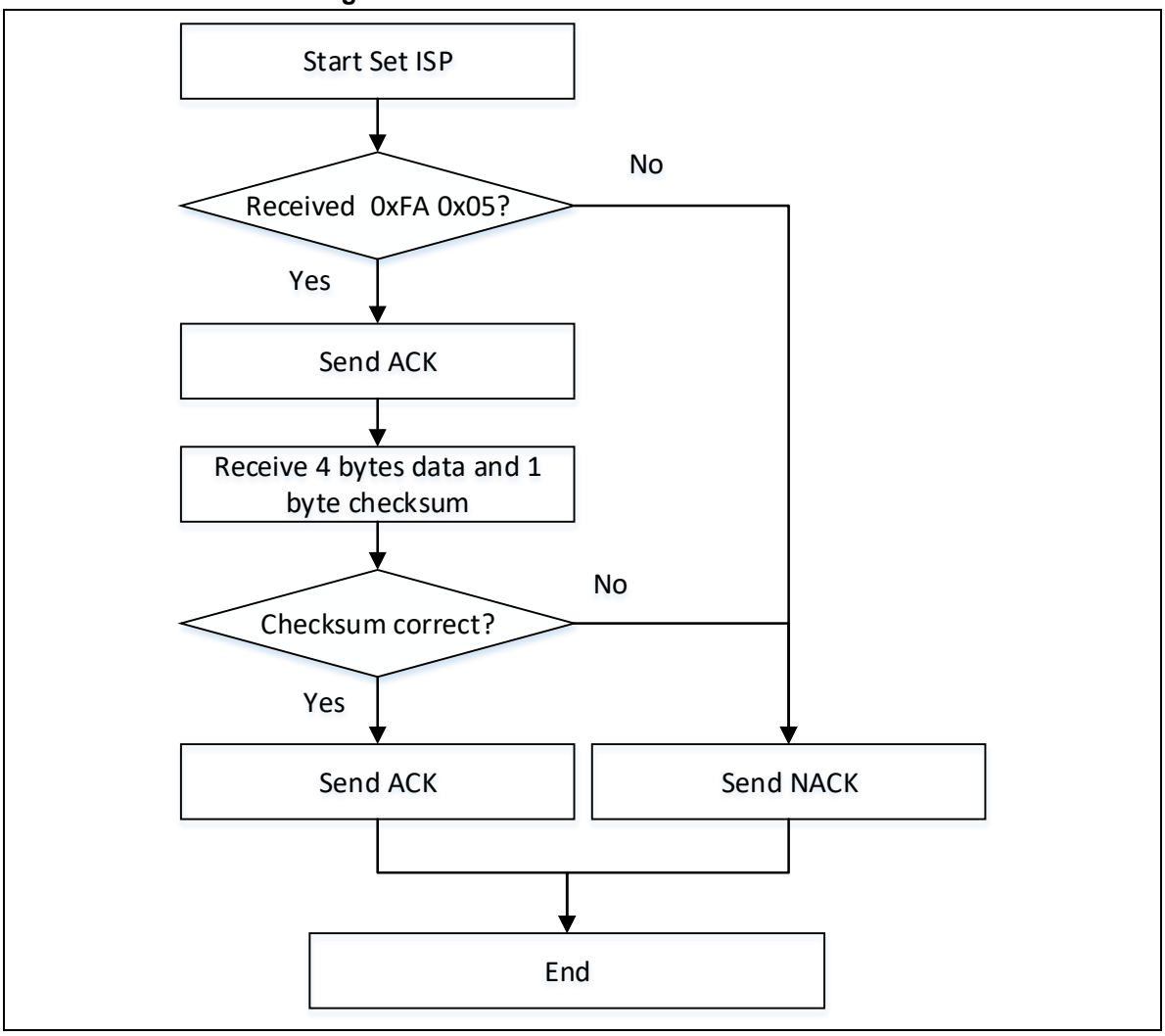


Figure 3 Set ISP flow chart on device side



4.1.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xFA	
2		0x05	
	1	ACK/NACK	This command ended when a NACK is received.
3		0x02	
4		0x03	
5		0x54	
6		0x41	
7		*	CheckSum(Byte3~Byte6)
	2	ACK	End of command

4.2 Get Commands

The Get Commands is used to get the bootloader protocol version and the supported commands. The number of commands and the command list depends on the microcontrollers.

After receiving this command, the bootloader sends an ACK, and the number of bytes (version and commands) to the host before sending version and the supported commands, then sends ACK to

the host once more.

Get Commands can also be used even if access protection is enabled.

Note: It is necessary to send Set ISP command before issuing Get Commands for AT32F413xx, AT32F415xx, AT32F403Axx, AT32F407xx, AT32F421x.

4.2.1 Get Commands flow chart

Figure 4 Get Commands flow chart on host side

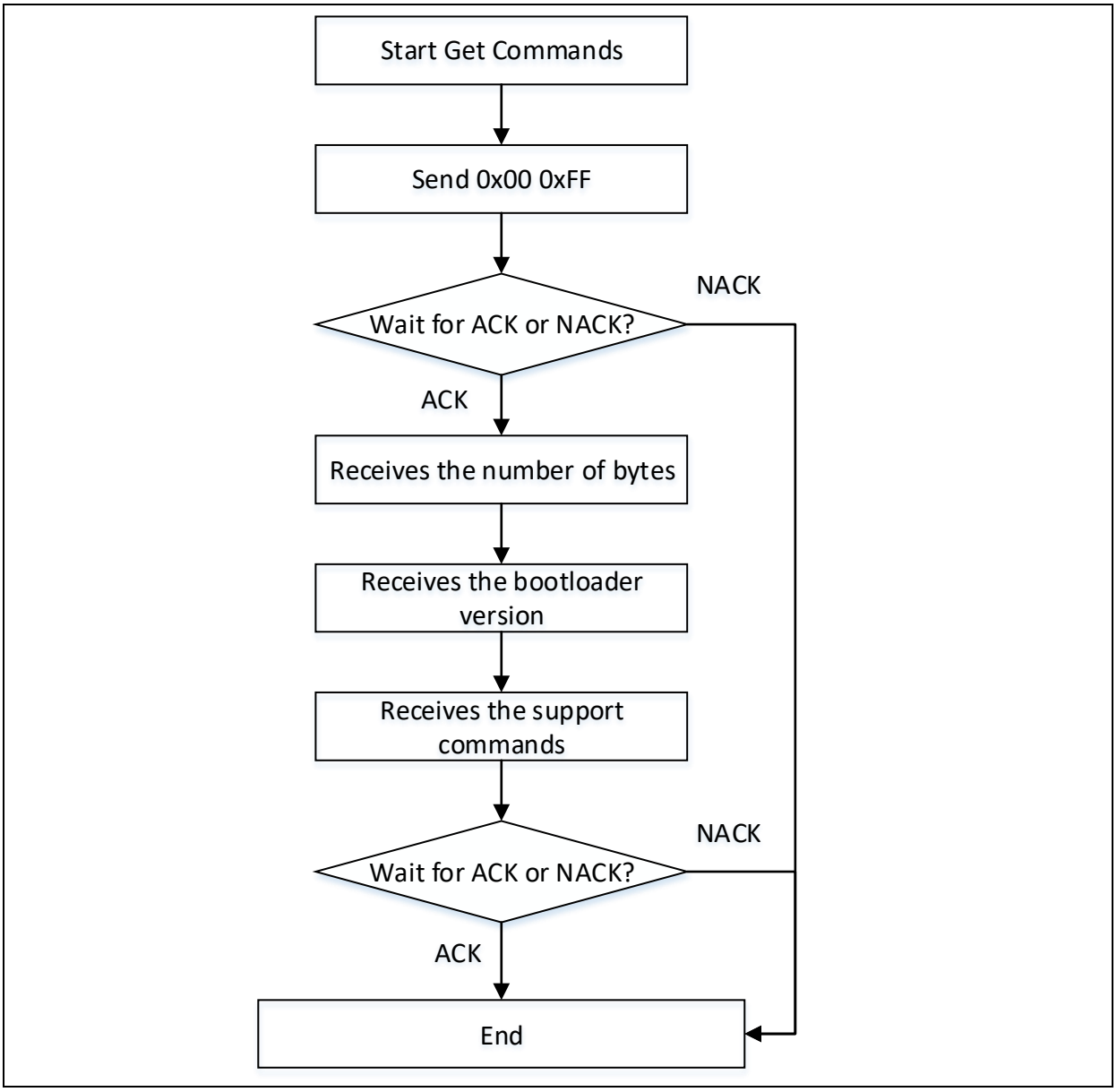
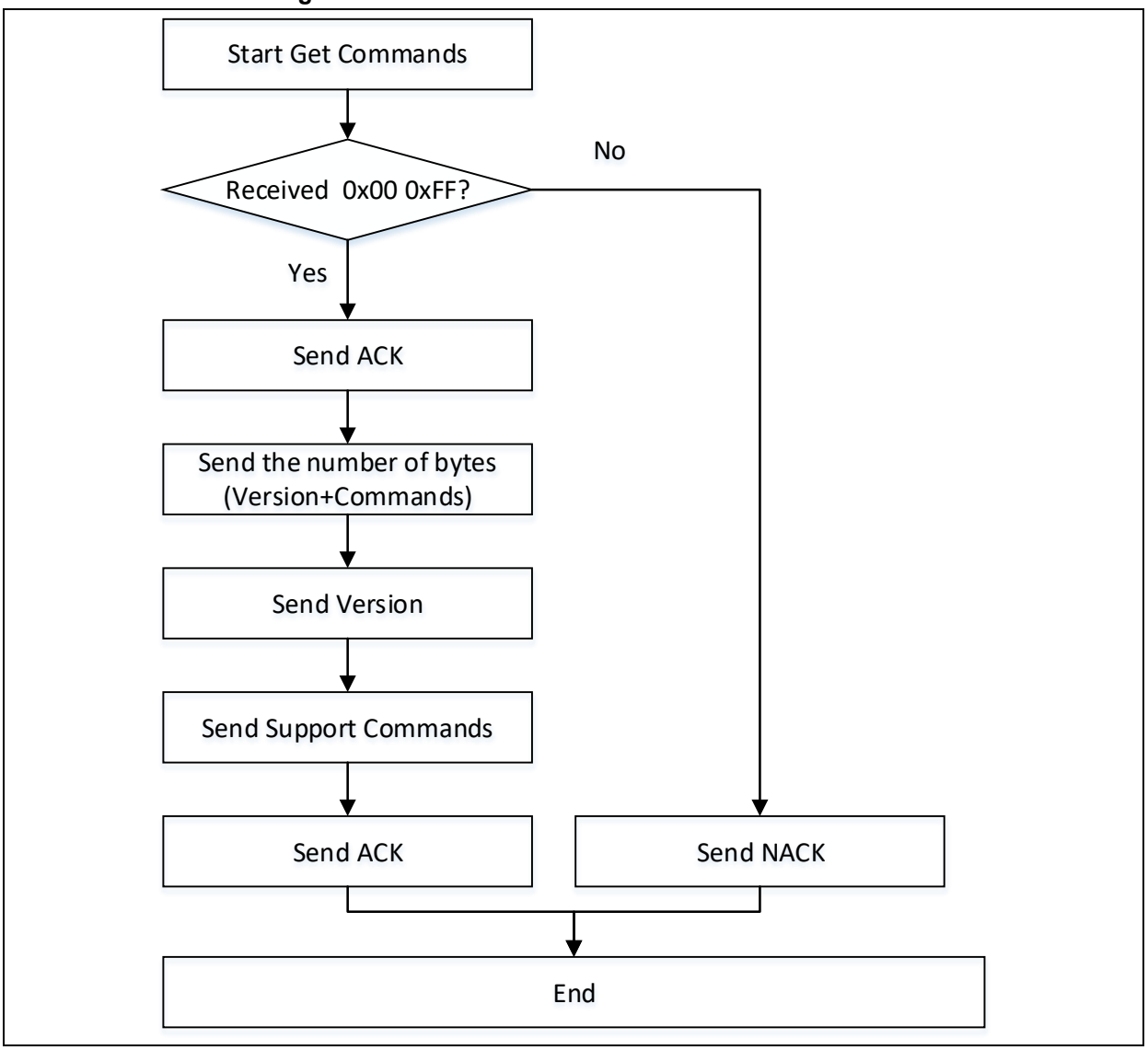


Figure 5 Get Commands flow chart on device side



4.2.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x00	
2		0xFF	
	1	ACK/NACK	When a NACK is received, this command stops.
	2	n	Indicates that n-byte data are received.
	3	*	Bootloader version
	4	0x00	First command: Get Commands
	5	0x01	Second command: Get Version
	…	…	…
	n+2	*	N-numbered command
	n+3	ACK	End of command

4.3 Get Version

The Get Version command is used to get the bootloader version of device. After receiving this command, the device sends an ACK to the host, and 1-byte Bootloader protocol version and 2-byte Bootloader version (BID), before sending ACK to host once more, which indicates the end of command.

This command is still valid even if access protection is enabled.

4.3.1 Get Version flow chart

Figure 6 Get Version flow chart on host side

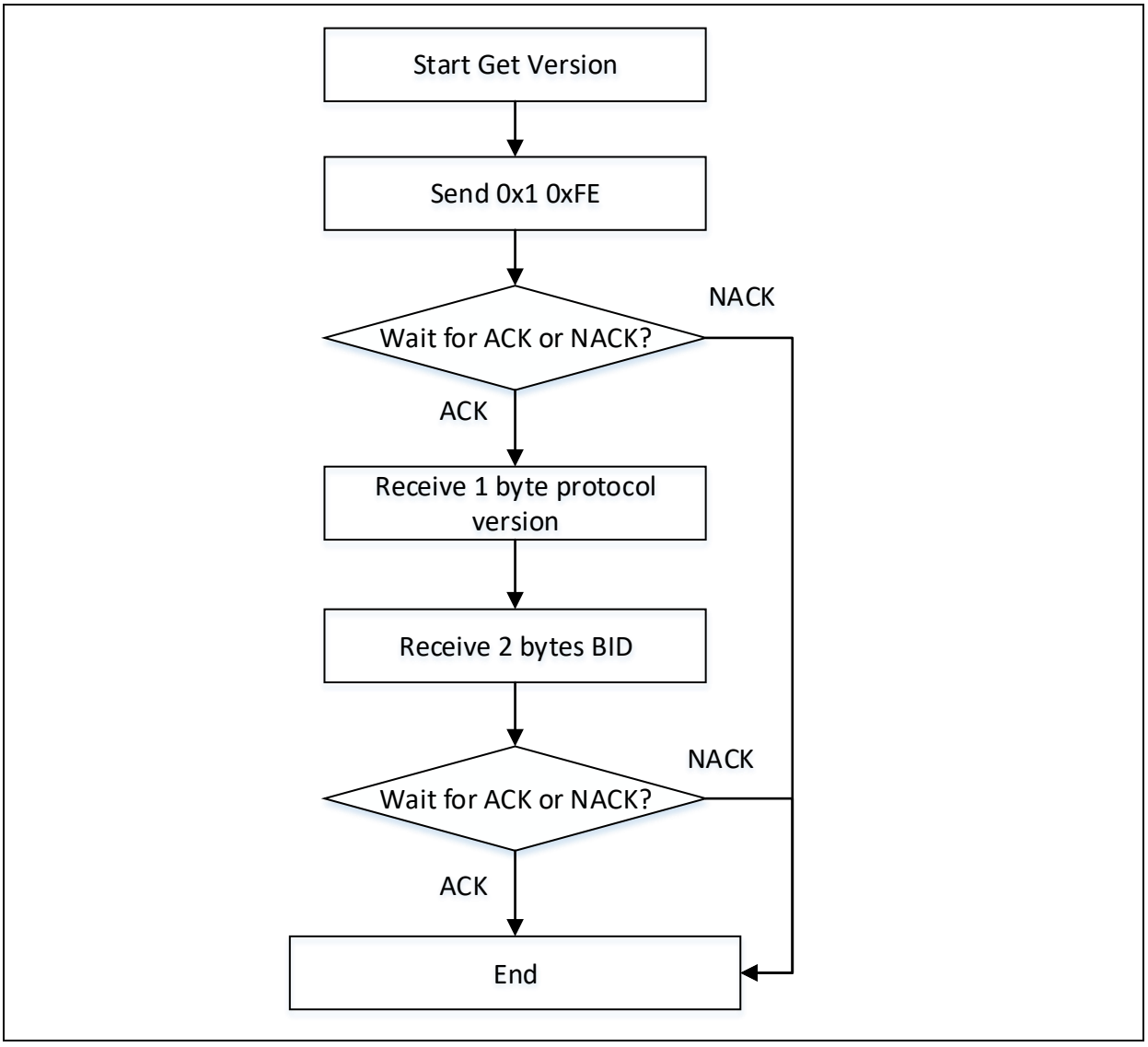
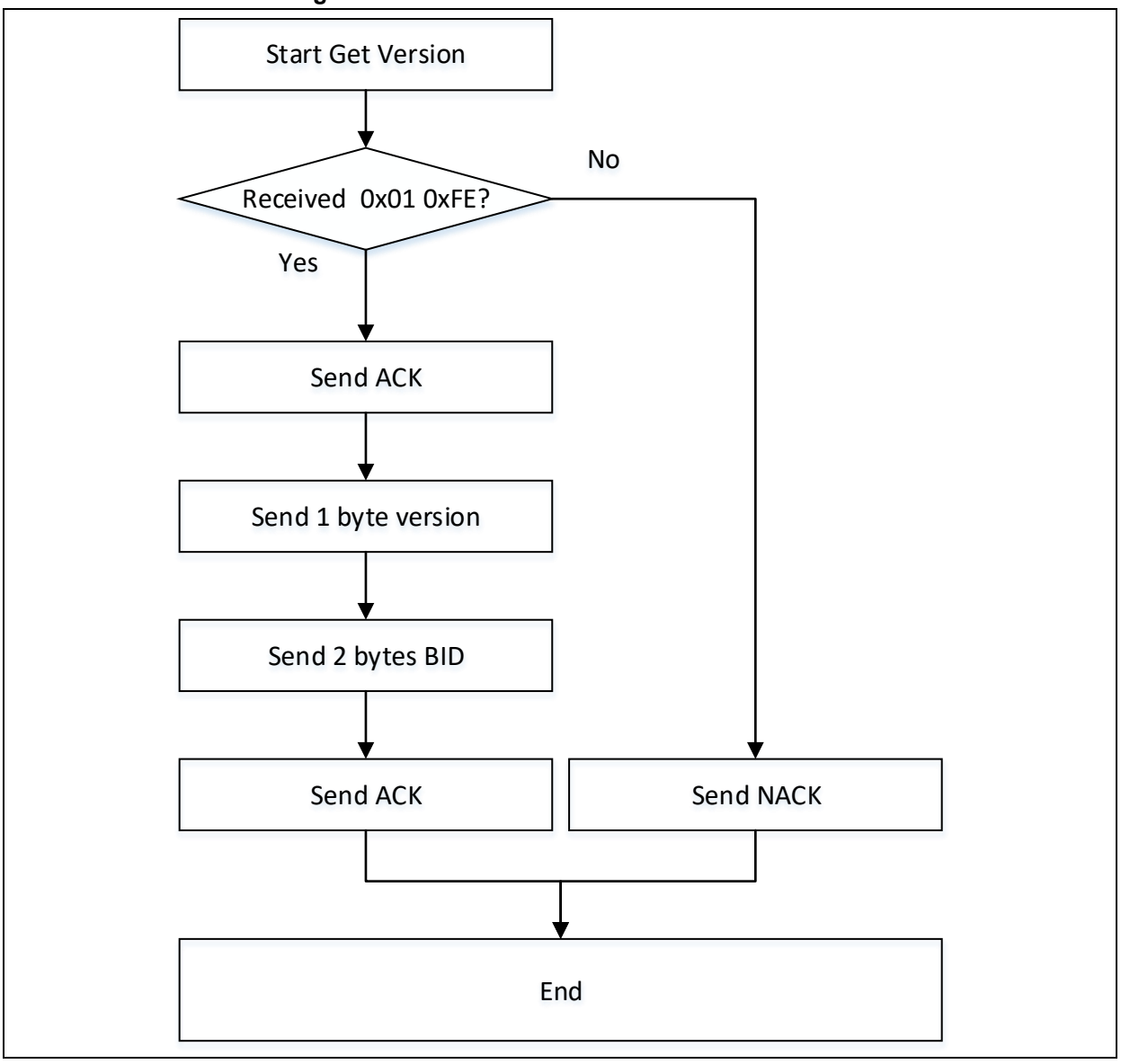


Figure 7 Get Version flow chart on device side



4.3.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x01	
2		0xFE	
	1	ACK/NACK	When NACK is received, this command stops.
	2	*	Protocol version
	3	*	Bootloader version (BID)
	4	*	Bootloader version (BID)
	5	ACK	End of command

4.4 Get Device ID

The Get Device ID command is used to read MCU part number. This ID consists of 4-byte product ID and 1-byte project ID. The 1-byte project ID indicates which one of MCU series is running, while the 4-byte product ID indicates the concrete part number under a certain MCU series.

After receiving this command, the device sends an ACK to host, and 1-byte data (its value is data transfer size decremented by 1, for example, data transfer size is 5, then this value is 4), then sends 4-byte product ID and 1-byte project ID, before sending an ACK once more.

This command is still valid even if access protection is enabled.

Table 1 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

4.4.1 Get Device ID flow chart

Figure 8 Get Device ID flow chart on host side

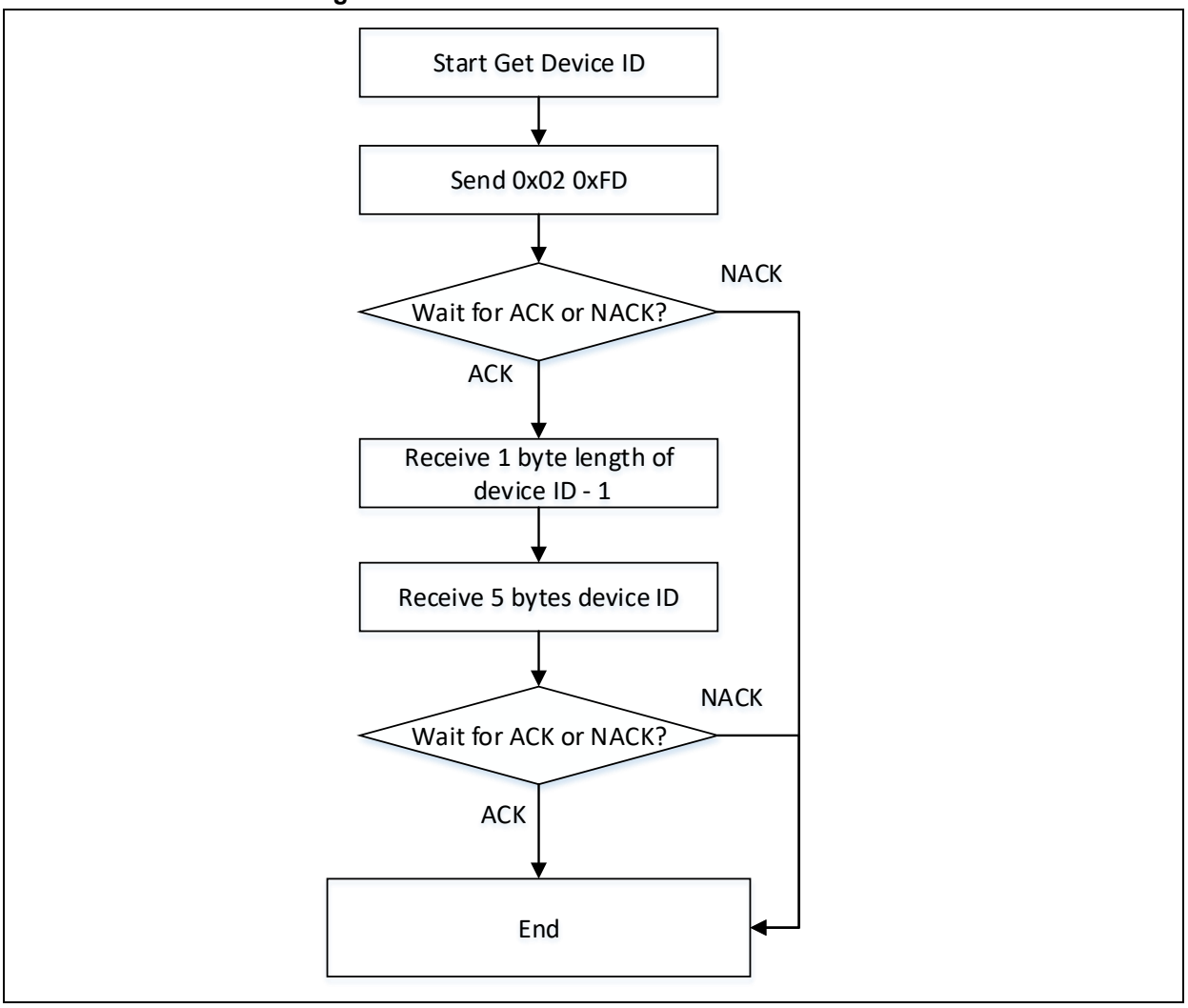
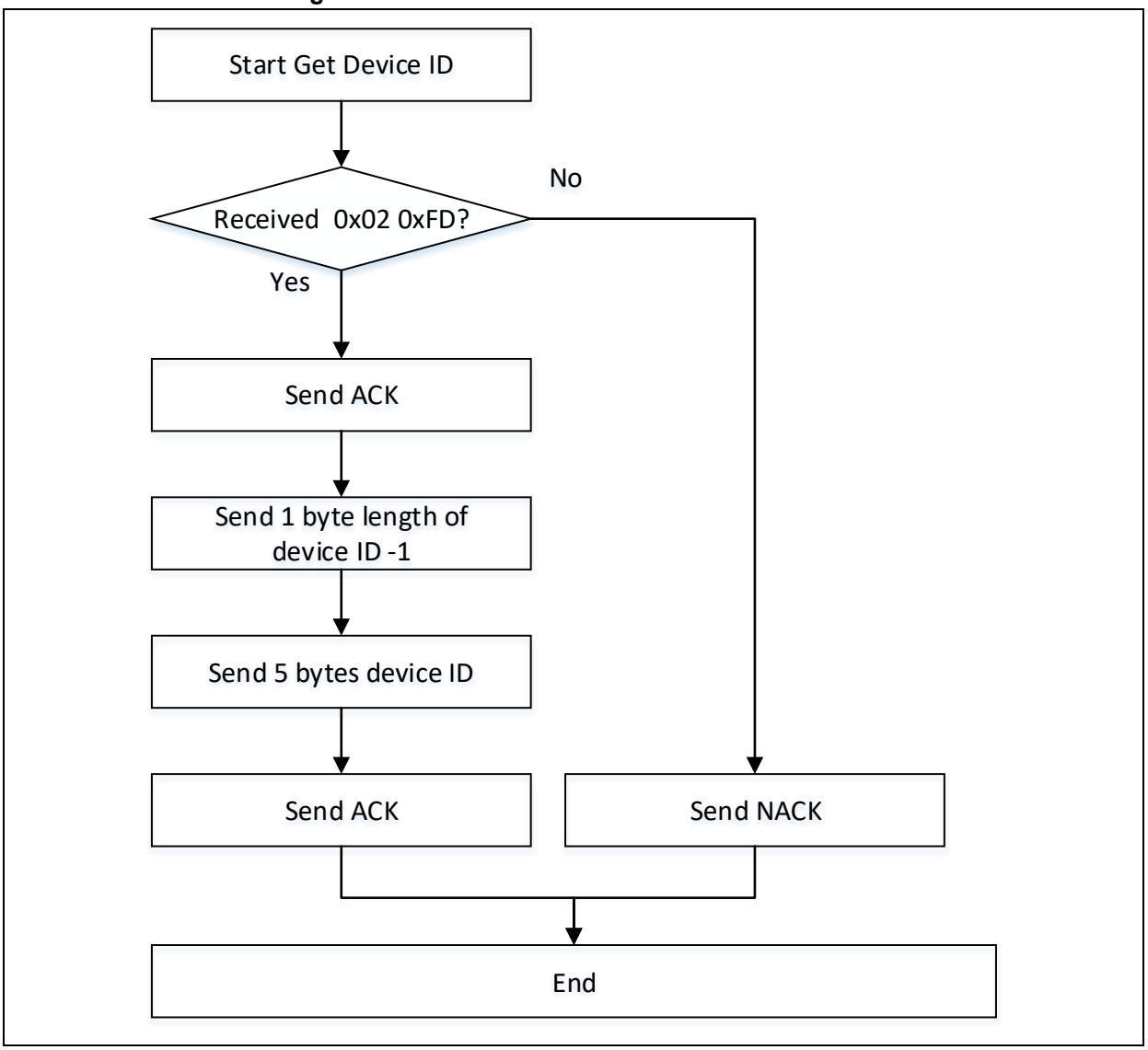


Figure 9 Get Device ID flow chart on device side



4.4.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x02	
2		0xFD	
	1	ACK/NACK	When a NACK is received, this command stops.
	2	0x04	Device ID (data transfer size decremented by 1)
	3	*	Product ID [8-15]
	4	*	Product ID [0-7]
	5	*	Product ID [24-31]
	6	*	Product ID [16-23]
	7	*	Project ID
	8	ACK	End of command

4.5 Read Memory

The Read memory command is used to read memory, SRAM, bootloader code, user system data and other data from any valid address. The range of read access depends on the microcontrollers. To read SPIM data (for devices with SPIM), it is necessary to enable SPIM in advance (before sending Enable SPIM command)

After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits to receive 4-byte address and 1-byte checksum. If both checksum and address are correct, the device issues an ACK to host, and waits to receive 1-byte read length (its value is to-be-read value decremented by 1, for example, if read 10-byte data, this value is 9) and its checksum. If checksum is correct, the device sends an ACK to host before sending data.
This command is invalid when access protection is enabled.

4.5.1 Read Memory flow chart

Figure 10 Read Memory flow chart on host side

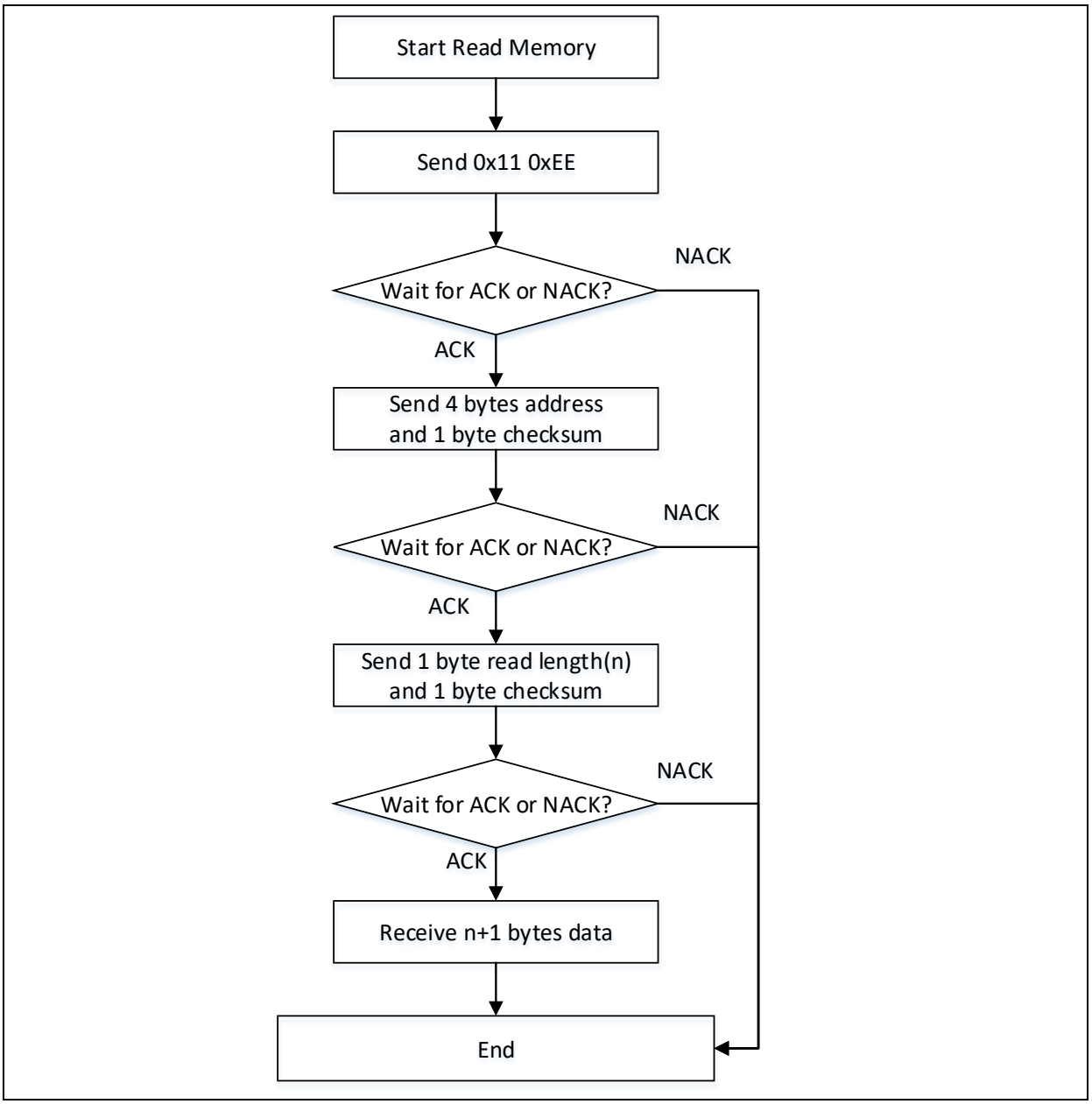
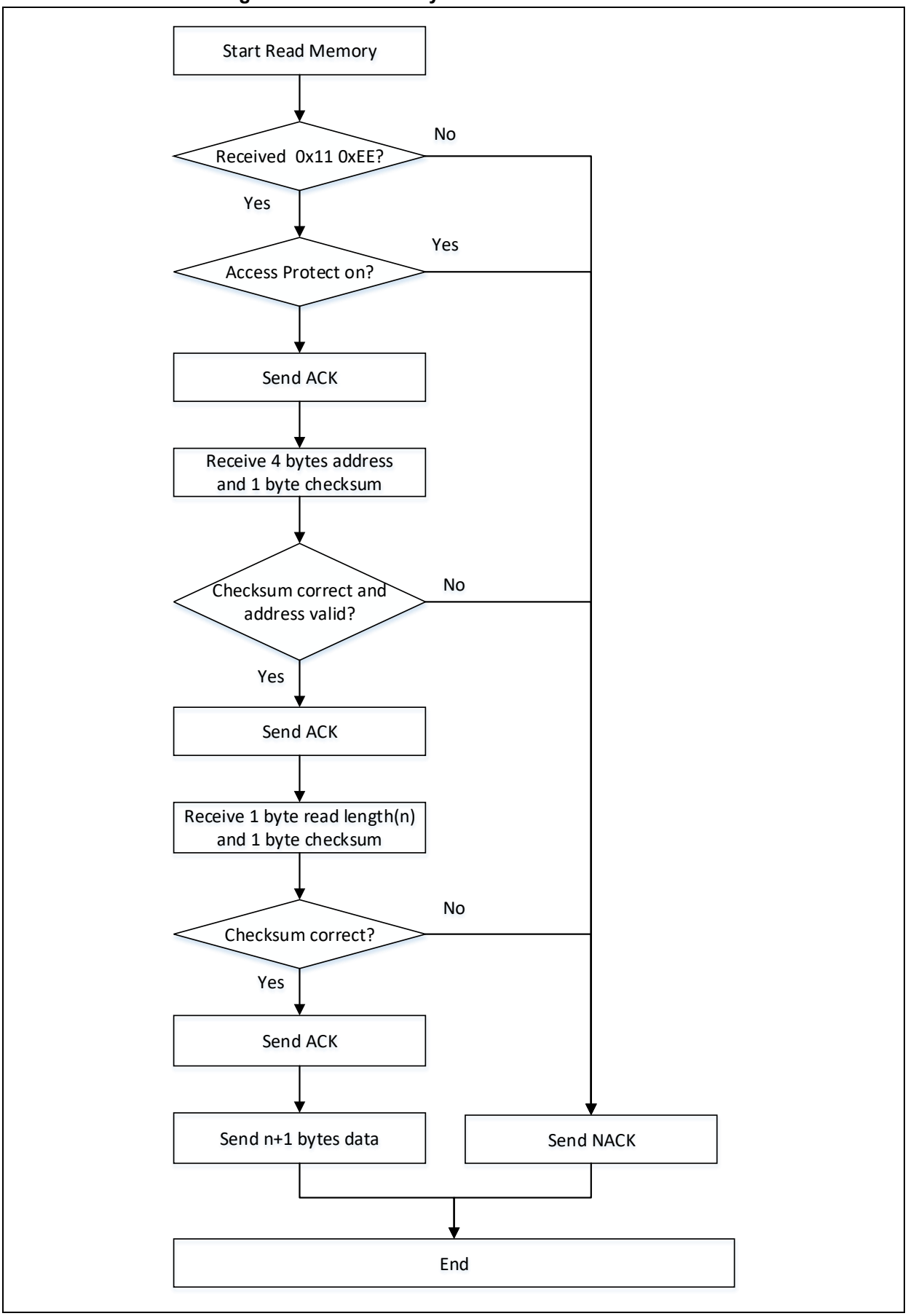


Figure 11 Read Memory flow chart on device side



4.5.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x11	
2		0xEE	
	1	ACK/NACK	When a NACK is received, it indicates that access protection is enabled, this command stops.
3		*	Address MSB
4		*	
5		*	
6		*	Address LSB
7		*	Checksum: XOR（address byte3~byte6）
	2	ACK/NACK	When NACK is received, this command stops.
8		*	Read data length – 1 (n)
9		*	Checksum: 0xFF XOR byte8
	3	ACK/NACK	When NACK is received, this command stops.
	4	*	Destination data
	…	…	Destination data
	4+n+1	*	Destination data

4.6 Jump

The Jump command is used to jump to a given address. It can jump to main memory and SRAM. The address to jump must be valid. The range of valid addresses depends on the microcontrollers.

After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits to receive 4-byte address and its checksum. If both address and checksum are valid, the device gives out an ACK to host, before jumping to the address to run.

This command cannot be used when access protection is enabled.

4.6.1 Jump flow chart

Figure 12 Jump flow chart on host side

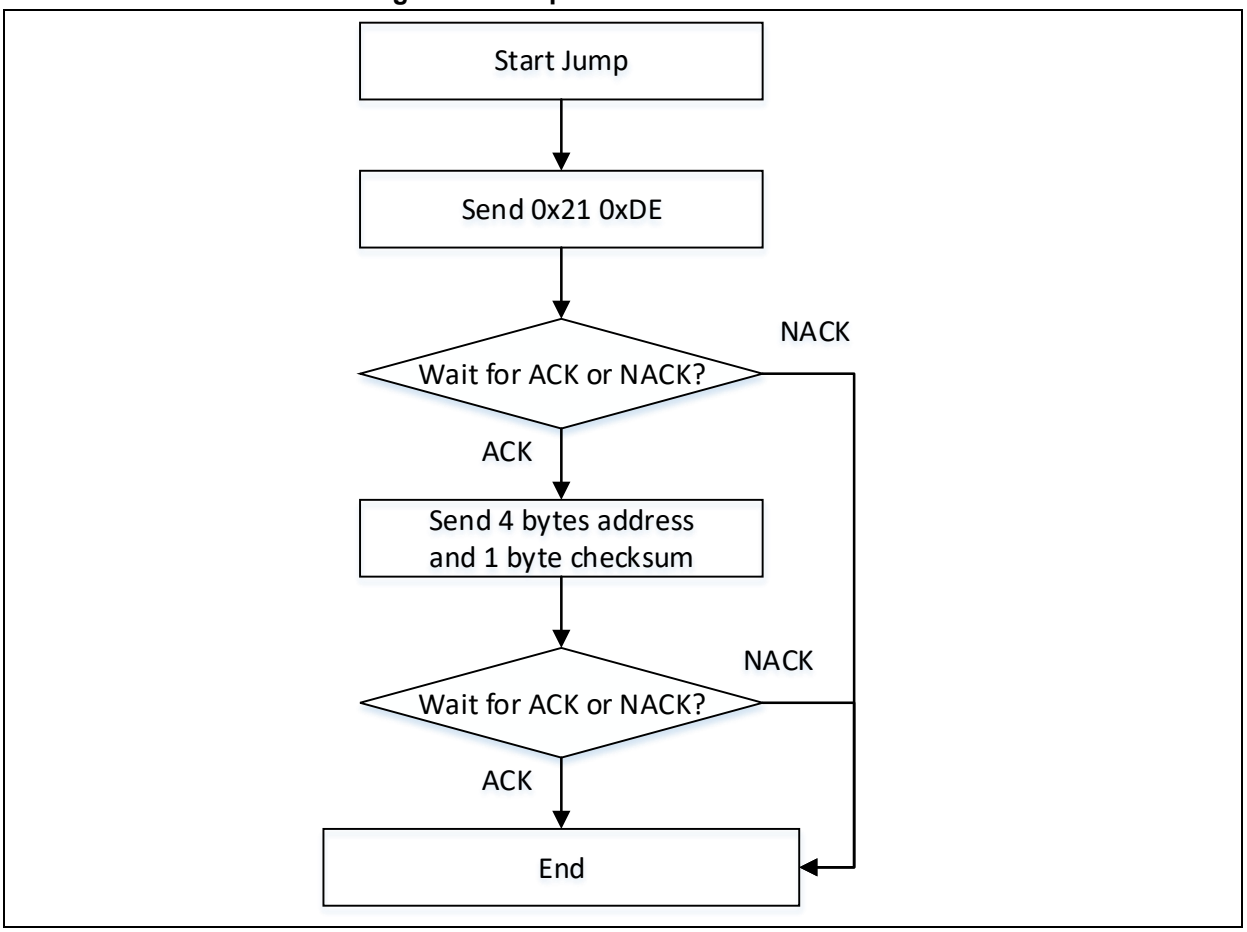
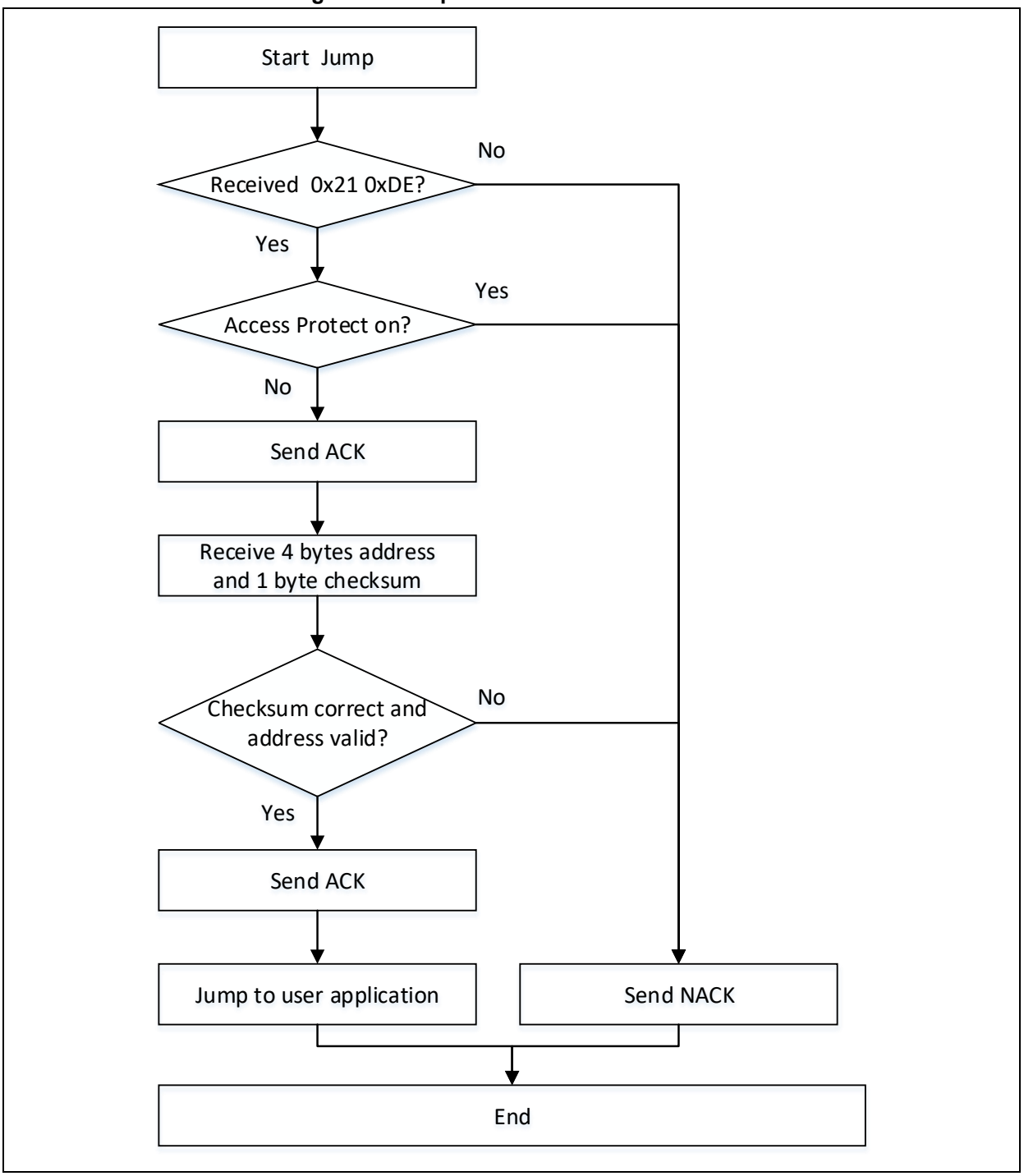


Figure 13 Jump flow chart on device side



4.6.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x21	Jump
2		0xDE	Jump
	1	ACK/NACK	The reception of NACK indicates that access protection is enabled, this command stops.
3		*	Address MSB
4		*	
5		*	

Transmit	Receive	Data	Description
6		*	Address LSB
7		*	Checksum：XOR（address byte3~byte6）
	2	ACK/NACK	

4.7 Write Memory

The Write memory command is used to write data to main memory, SRAM, user system data. The data in the corresponding valid address must be erased before write main memory. The range of valid addresses depends on the microcontrollers.

User system data area has two different write operations, depending on its size:

- When user system data area is less than 256 bytes:
  The host sends a single write command to write all user system data, and the device, after receiving this command, erases user system data automatically, and writes data, before performing a system reset.
- When user system data area is greater than 256 bytes
  The host must send several write commands to write all user system data
  After receiving this command, if the address is a start address of user system data area, the device erases and writes it; if it is not a start address of the user system data area, no erase is performed, the device writes data to the corresponding address directly instead. A system reset is also performed accordingly at the end of write operation.
  Note: For AT32F435xx/AT32F437xx, no system reset is performed after write access to user system data area, so a reset command is needed from host.

After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits for 4-byte write address and its checksum. If both address and its checksum are correct, the devices sends an ACK to host, and waits for 1-byte write length (it is the to-be-written data size decremented by 1, for example, if write 10-byte data, this length is 9) and its checksum. For a correct checksum, the device writes data to the corresponding address, and sends an ACK at the end of write operation.

This command cannot be used when access protection is enabled.

4.7.1 Write memory flow chart

Figure 14 Write Memory flow chart on host side

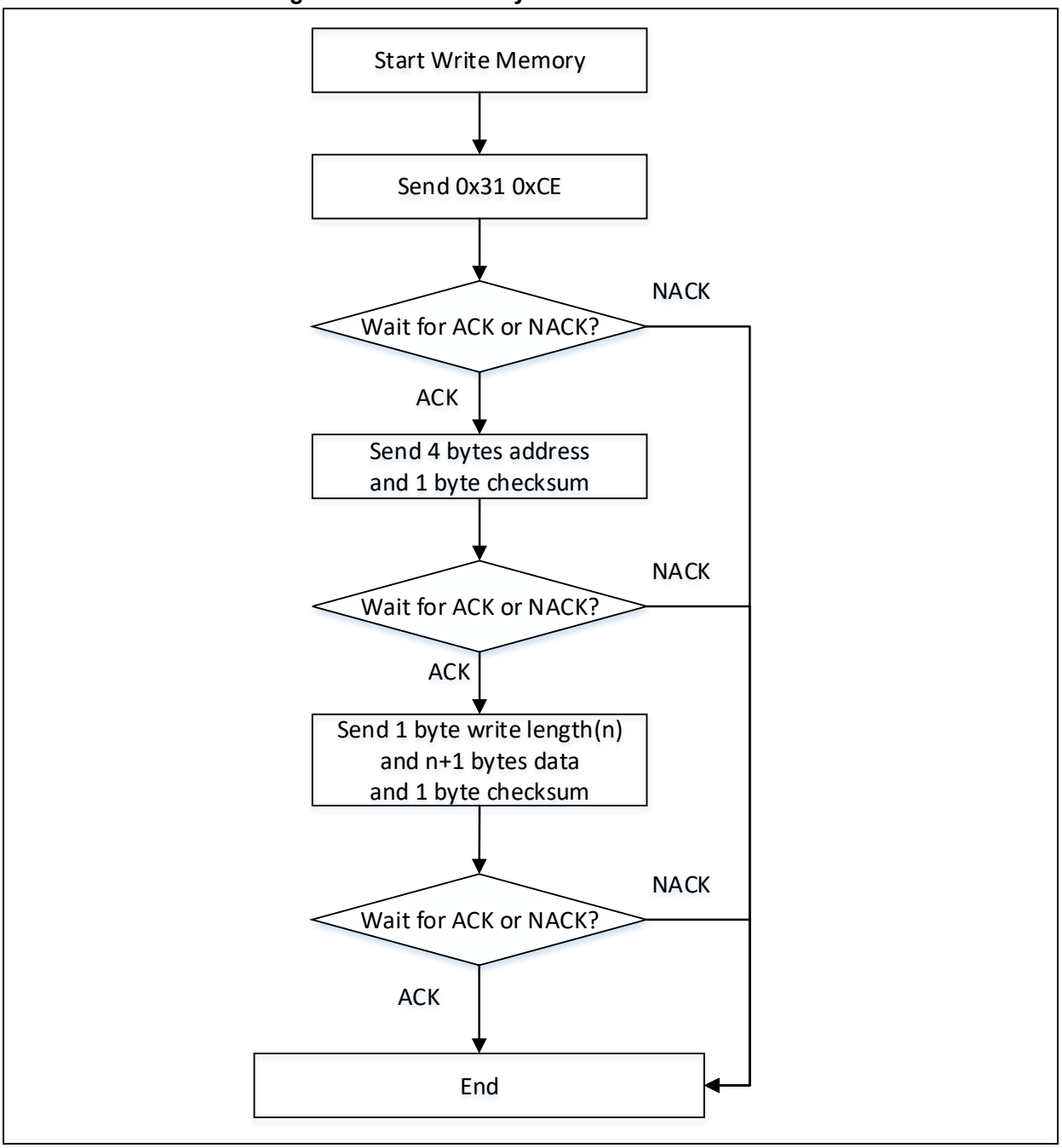
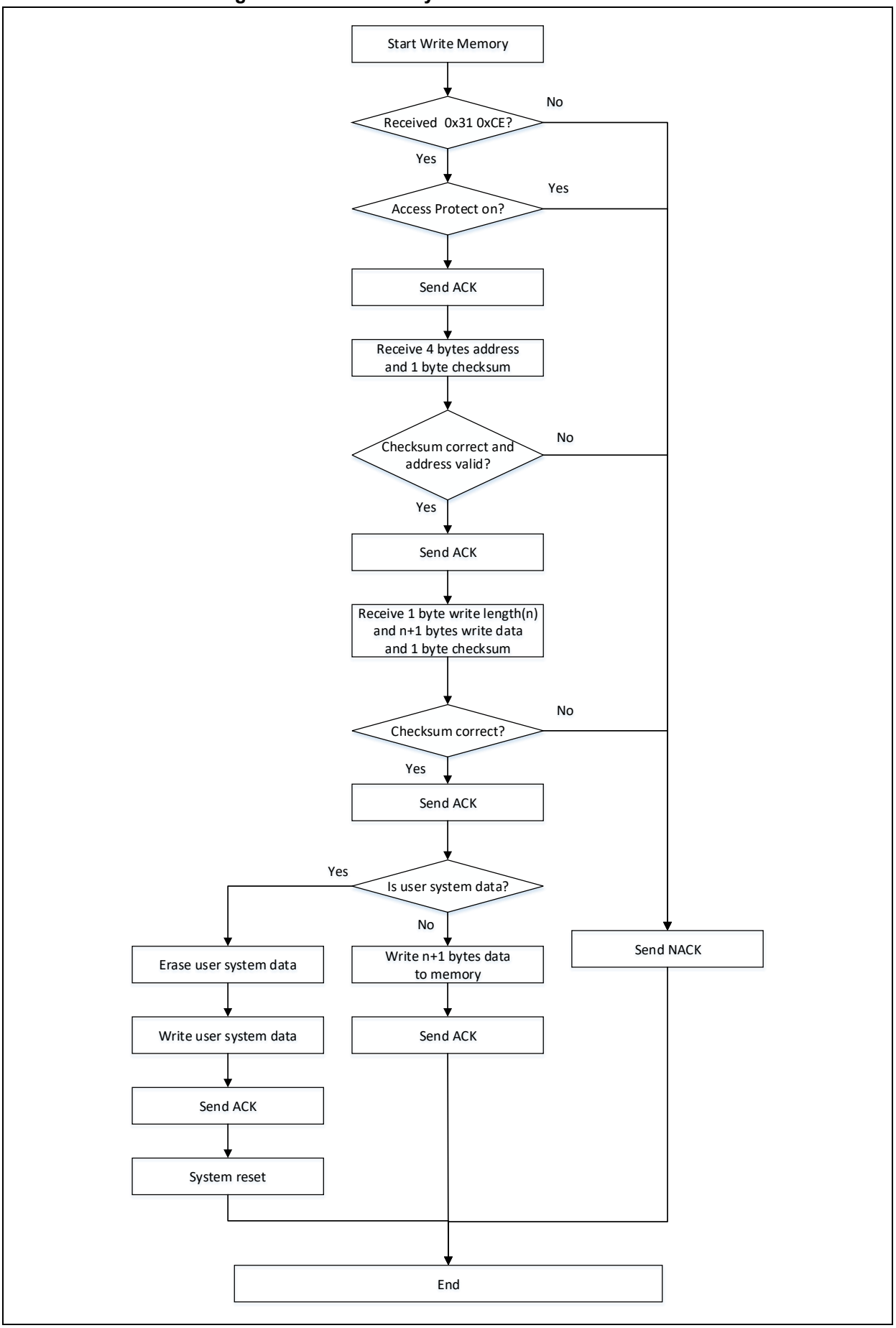


Figure 15 Write Memory flow chart on device side



4.7.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x31	Write Memory
2		0xCE	Write Memory
	1	ACK/NACK	When NACK is received, it indicates that access protection is enabled, this command stops.
3		*	Address MSB
4		*	
5		*	
6		*	Address LSB
7		*	Checksum: XOR（address byte3~byte6）
	2	ACK/NACK	When a NACK is received, this command stops.
8		*	Write length n - 1
9		*	Write data
…		…	Write data
9+n		*	Write data
			Checksum: XOR byte8 - byte9+n
	3	ACK/NACK	End of command

4.8 Erase

The Erase command is used to erase main memory. 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. Besides, for AT32F435xx/AT32F437xx, block erase is also supported.

After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits to receive 2-byte data (this data defines erase type)

- For sector erase, it indicates the number of sector to erase. The device waits to receive the index of sectors (n) to erase, and then erases the sector, and sends an ACK to host at the end of erase operation.
- For mass erase or Bank erase, the device performs erase operation directly, and then sends an ACK to host at the end of erase operation.
- For block erase, the device waits to receive 4-byte block start address and its checksum. If the checksum is correct, the device performs block erase and sends an ACK to host at the end of erase operation. Both AT32F435xx and AT32F437xx support block erase, of 64 KB each block.

This command cannot be used when access protection is enabled.

Table 2 Erase type summary

Type	Index	Location
Sector erase	0x00 0x00	Sector0
	0x00 0x01	Sector1
	…	…
	0x80 00	Bank3 Sector0
	0x80 01	Bank3 Sector1
	…	…
	0x8F 0xFF	Bank3 Sector4096
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

Note: Refer to the particular MCU for sector size.

4.8.1 Erase flow chart

Figure 16 Erase flow chart on host side

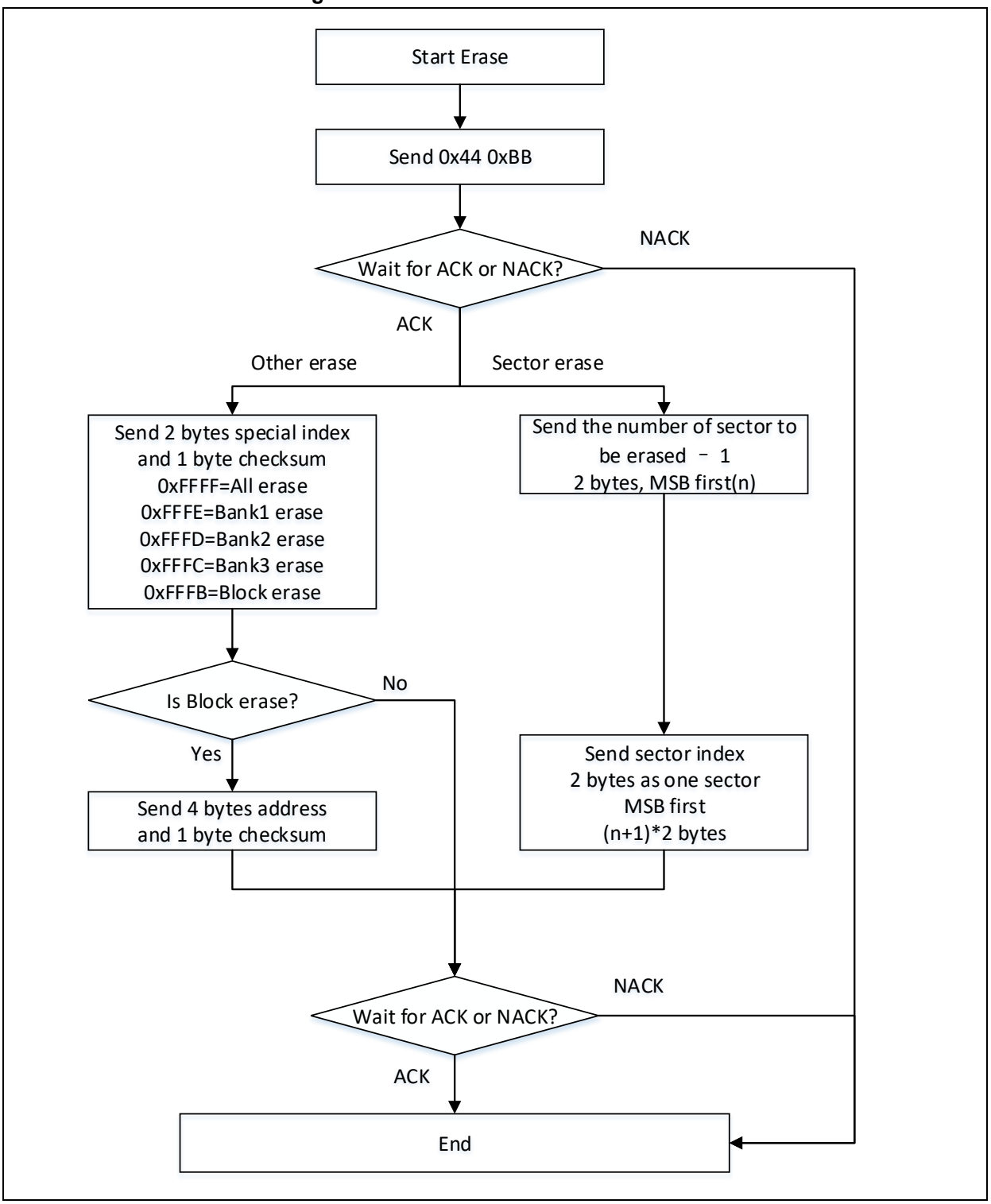
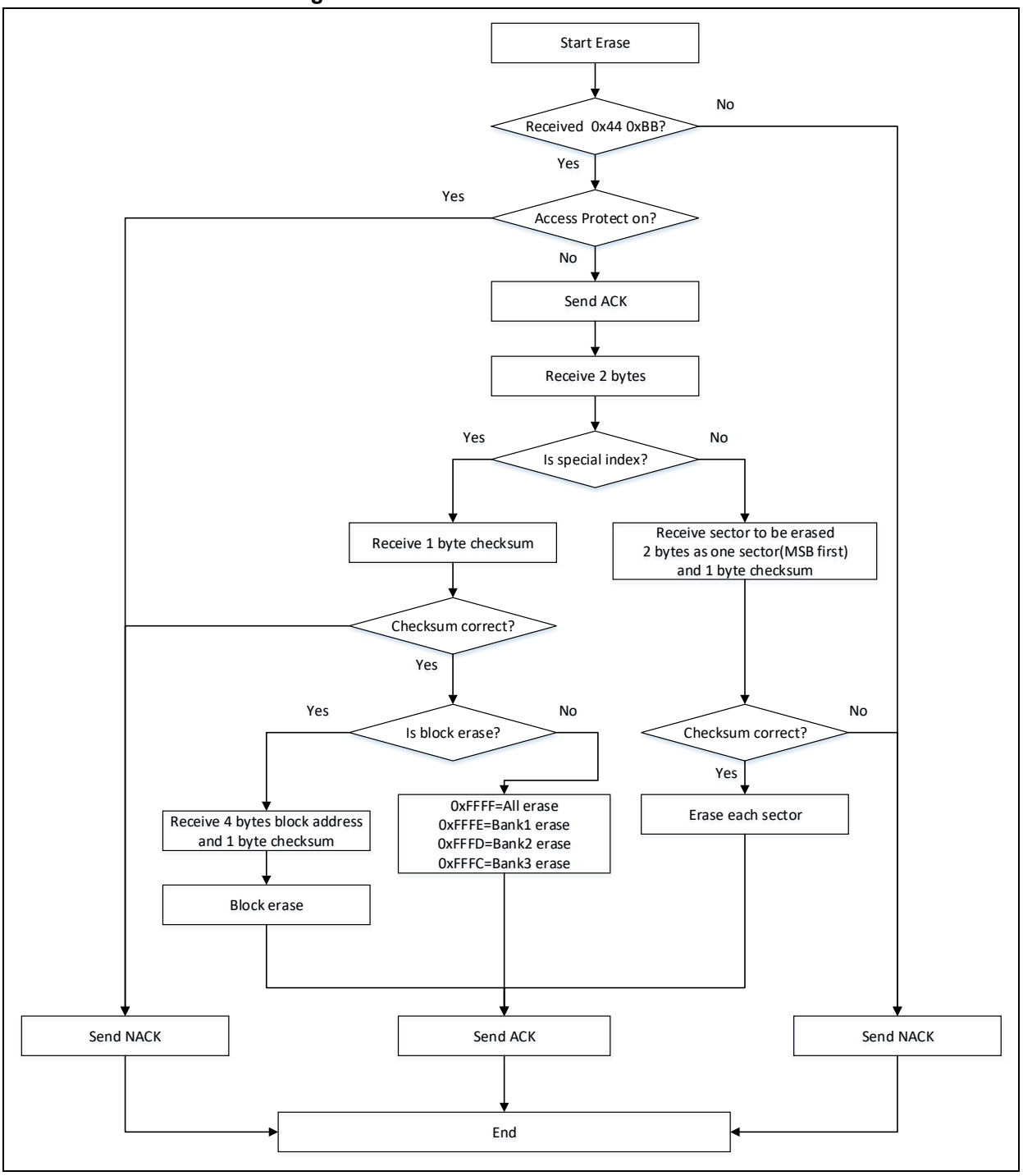


Figure 17 Erase flow chart on device side



4.8.2 Data transfer process on host side

Sector erase:

Transmit	Receive	Data	Description
1		0x44	Erase
2		0xBB	Erase
	1	ACK/NACK	When a NACK is received, this command stops.
3		*	Number of sector erase – 1 (n) MSB
4		*	Number of sector erase – 1 (n) LSB

Transmit	Receive	Data	Description
5		*	First Sector index (MSB)
6		*	First Sector index (LSB)
…		*	X Sector index (MSB)
…		*	X Sector index (LSB)
7+2(n+1)		*	Checksum: XOR (byte 3~6+2(n+1))
	2	ACK	

Block erase:

Transmit	Receive	Data	Description
1		0x44	Erase
2		0xBB	Erase
	1	ACK/NACK	When a NACK is received, this command stops.
3		0xFF	Block erase
4		0xFB	Block erase
5		*	Checksum: XOR (3~4 bytes)
6		*	Block address (MSB)
7		*	Block address
8		*	Block address
9		*	Block address (LSB)
10		*	Checksum: XOR (6~9 bytes)
	2	ACK	

Bank or mass erase

Transmit	Receive	Data	Description
1		0x44	Erase
2		0xBB	Erase
	1	ACK/NACK	When a NACK is received, this command stops.
3		0xFF	Mass or Bank erase index (MSB)
4		*	Mass erase (0xFF) or Bank erase (0xFE,, 0xFD, 0xFC) index (LSB)
5		*	Checksum: XOR (3~4 bytes)
	2	ACK	

4.9 Erase and Program Protect

Erase and Program protect command is used to protect a given sector against erase and programming.

After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits to receive 1-byte length -1 (n), n+1 byte sector index (refer to user system data area for more information on erase protection bit), and 1-byte checksum. For a correct checksum, the device erases user system area (preserving other data than erase protection byte in the user system area),

configures erase and program protection, and then sends an ACK to host before performing a system reset.

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.

This command cannot be used when access protection is enabled.

4.9.1 Erase and program protect flow chart

Figure 18 Erase and Program Protect flow chart on host side

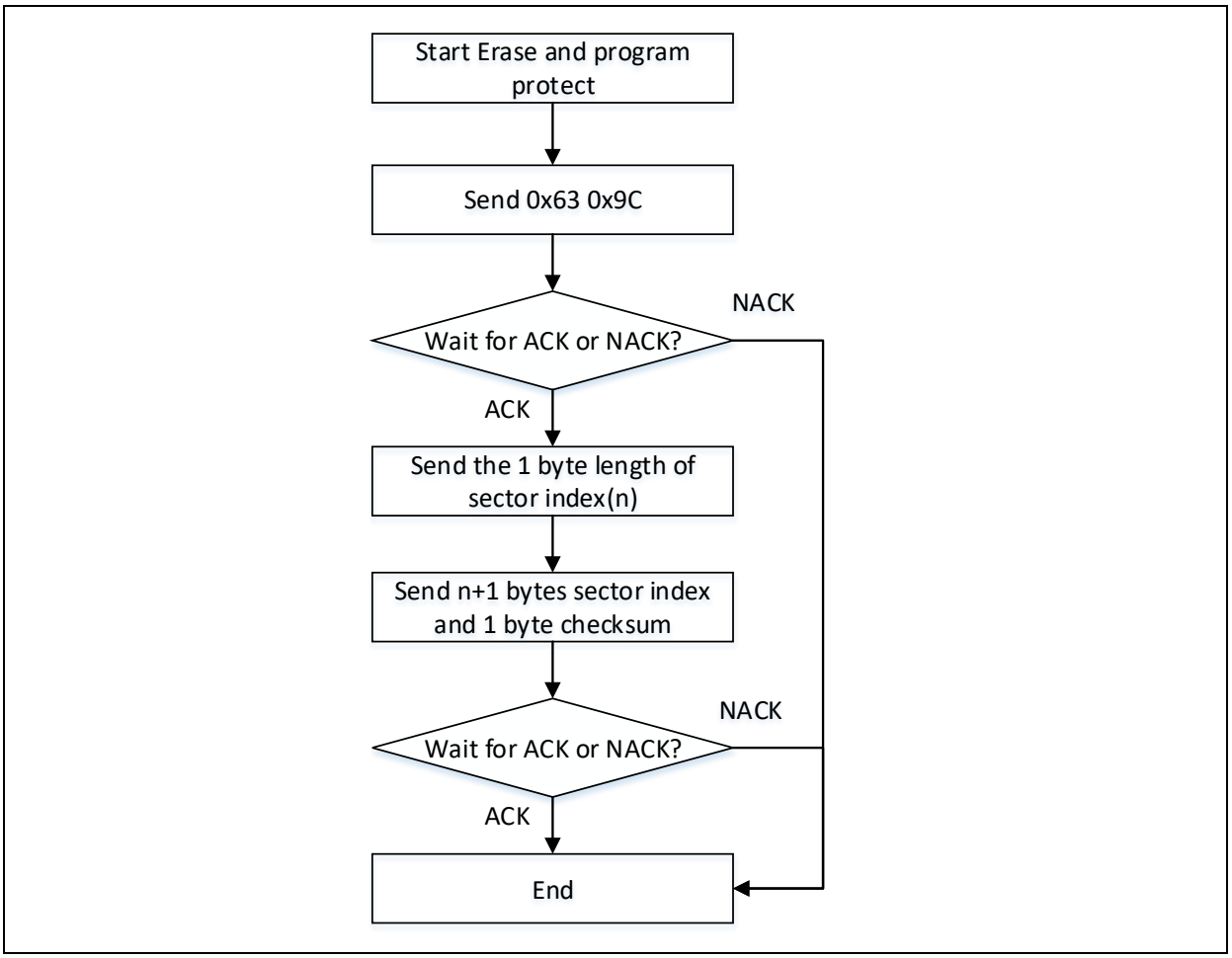
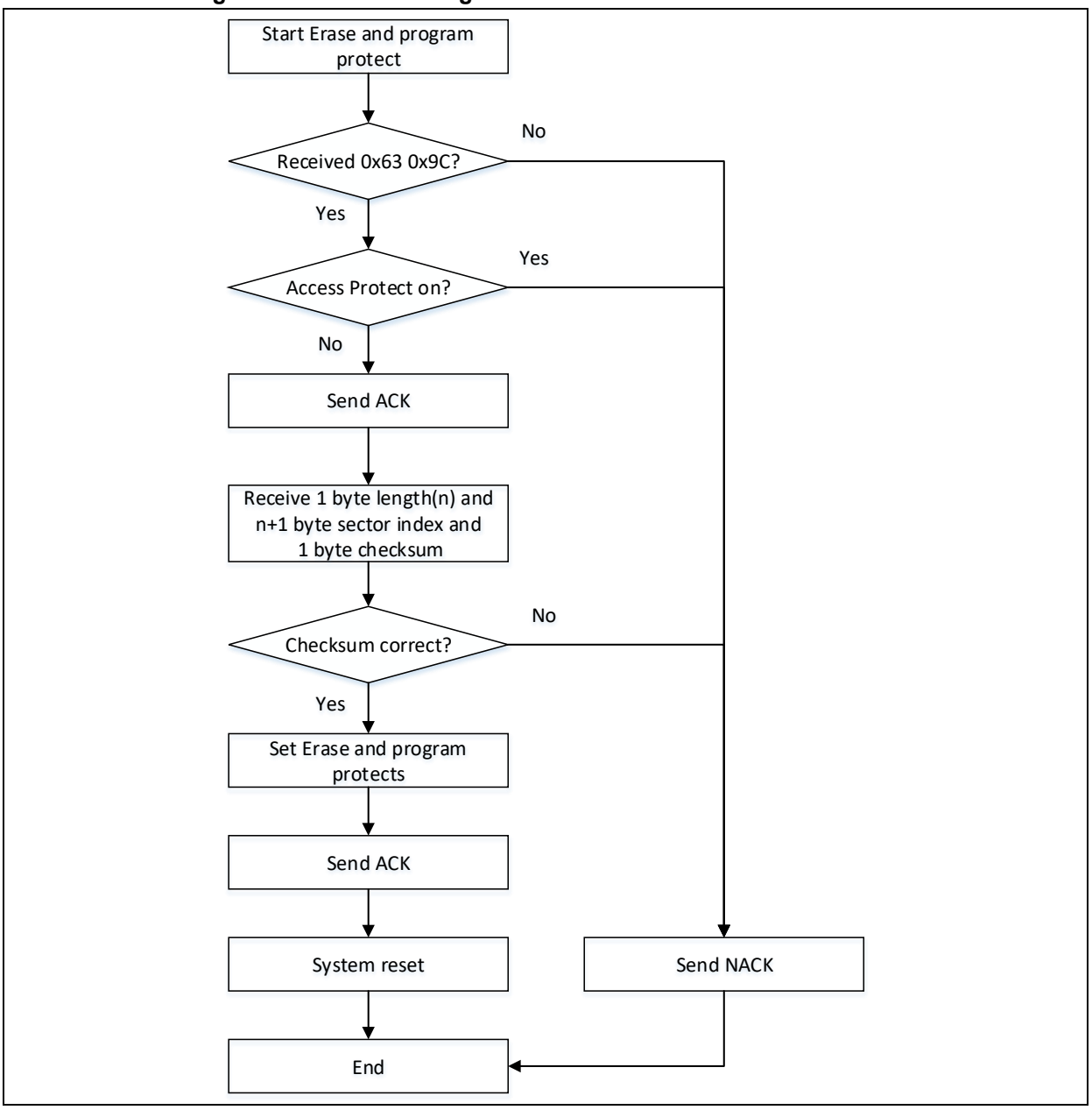


Figure 19 Erase and Program Protect flow chart on device side



4.9.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x63	Erase and program protect
2		0x9C	Erase and program protect
	1	ACK/NACK	When a NACK is received, this command stops
3		*	Protection length byte -1 (n)
…		…	Protection setting index
5+n+1		*	Checksum: XOR (byte 3~4+n+1)
	2	ACK	

4.10 Erase and Program Unprotect

Erase and program unprotect command is used to unlock memory protection.

After receiving this command, if access protection is disabled, the device sends an ACK to host, unlock erase and program protection of all sectors, and sends an ACK to host once more, before a system reset.

This command cannot be used when access protection is enabled.

4.10.1 Erase and Program Unprotect flow chart

Figure 20 Erase and Program Unprotect flow chart on host side

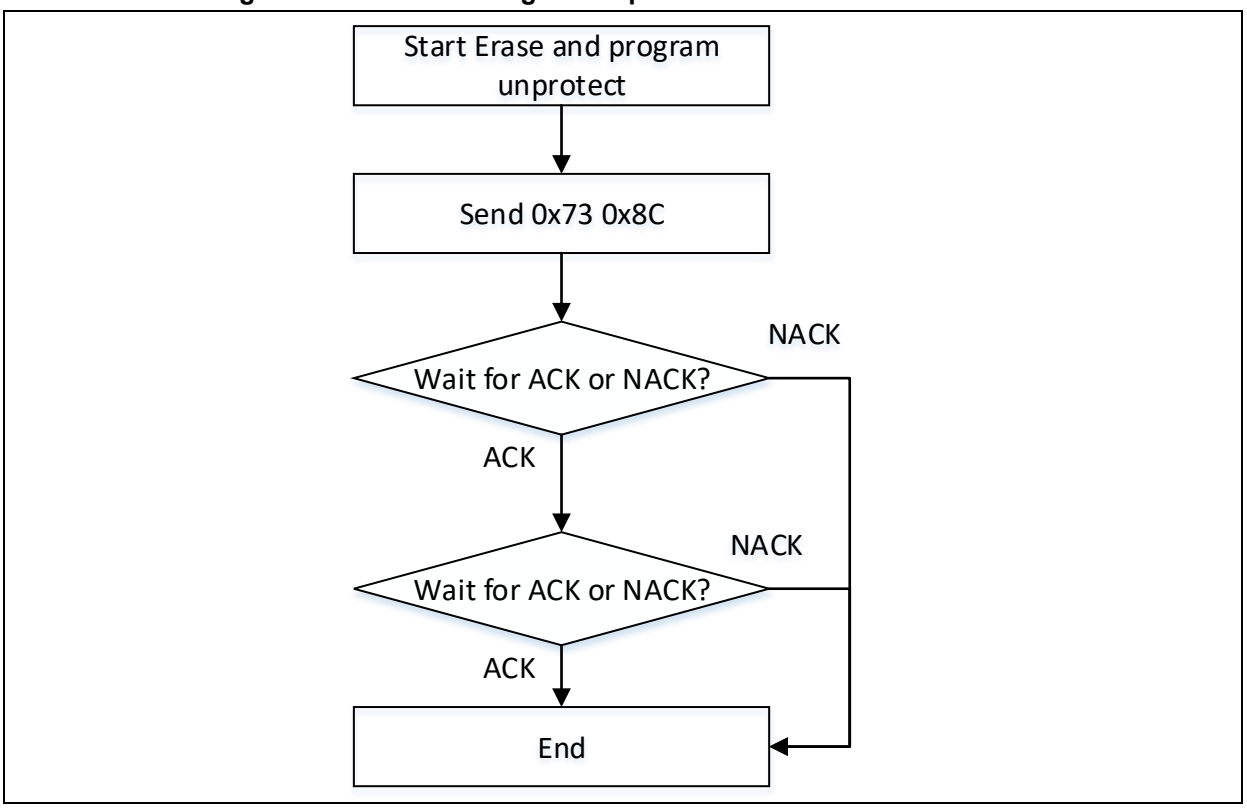
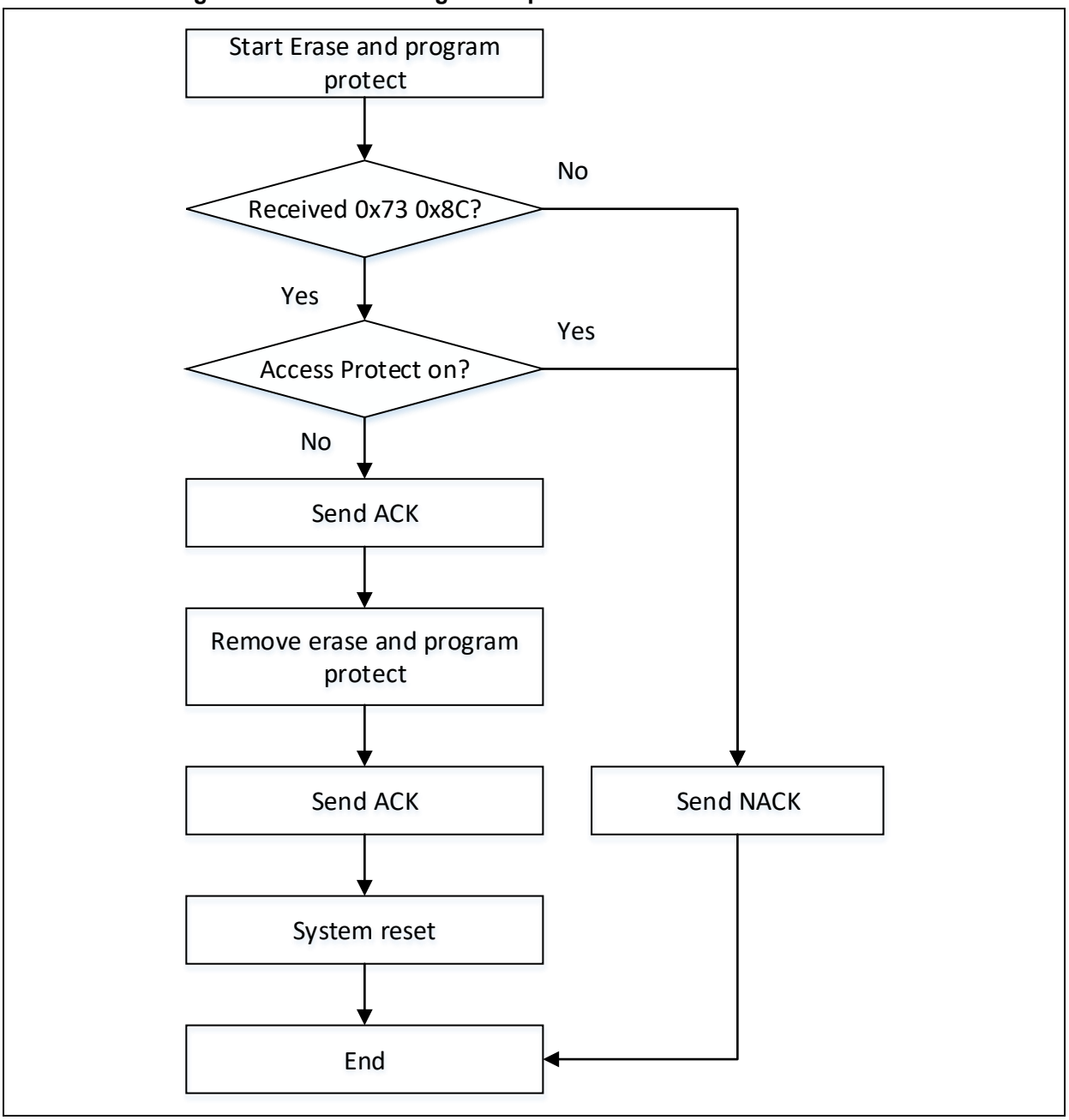


Figure 21 Erase and Program Unprotect flow chart on device side



4.10.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x73	Erase and program unprotect
2		0x8C	Erase and program unprotect
	1	ACK/NACK	When a NACK is received, this command stops
	2	ACK	

4.11 Access Protect

Access Protect command is used to enable memory access protection. Once enabled, read memory is prohibited.

After receiving this command, if access protection is disabled, the device sends an ACK to host, enables access protection, and sends an ACK to host at the completion of access protection, before performing a system reset.

This command cannot be used when access protection is enabled.

4.11.1 Access Protect flow chart

Figure 22 Access Protect flow chart on host side

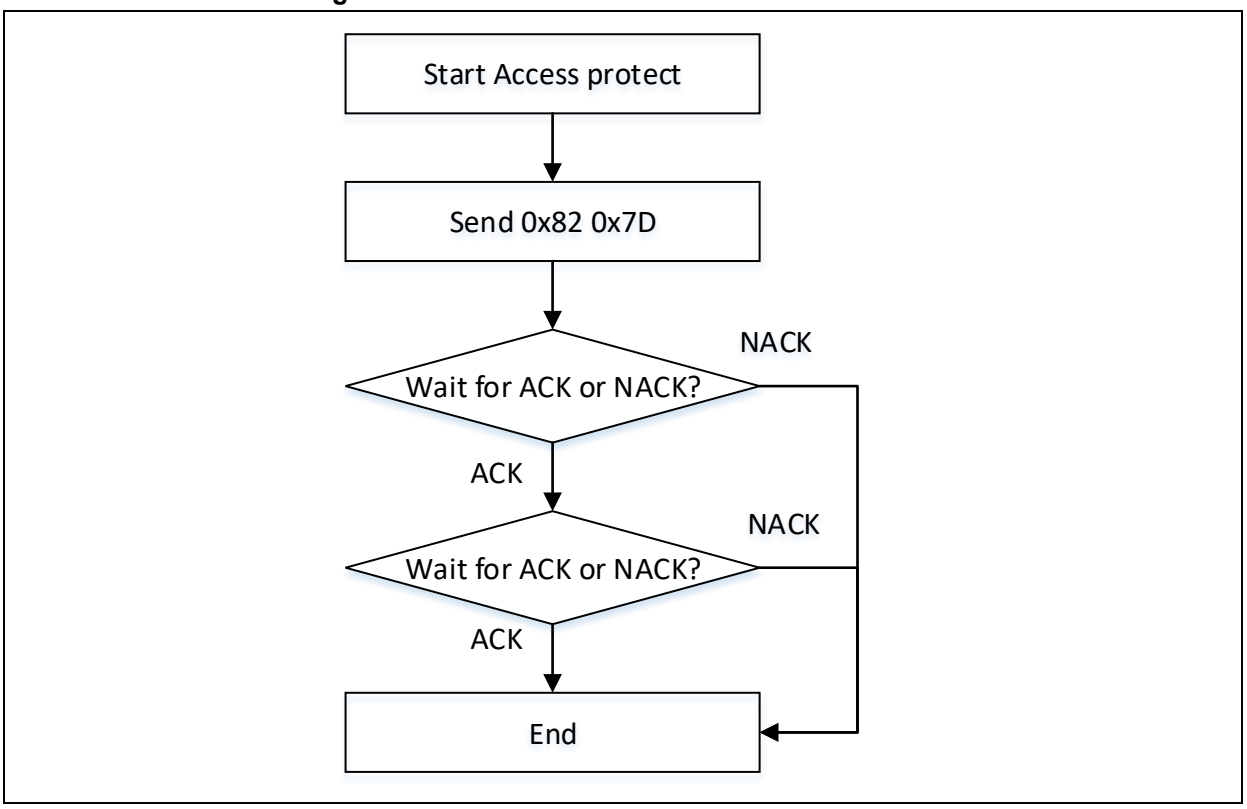
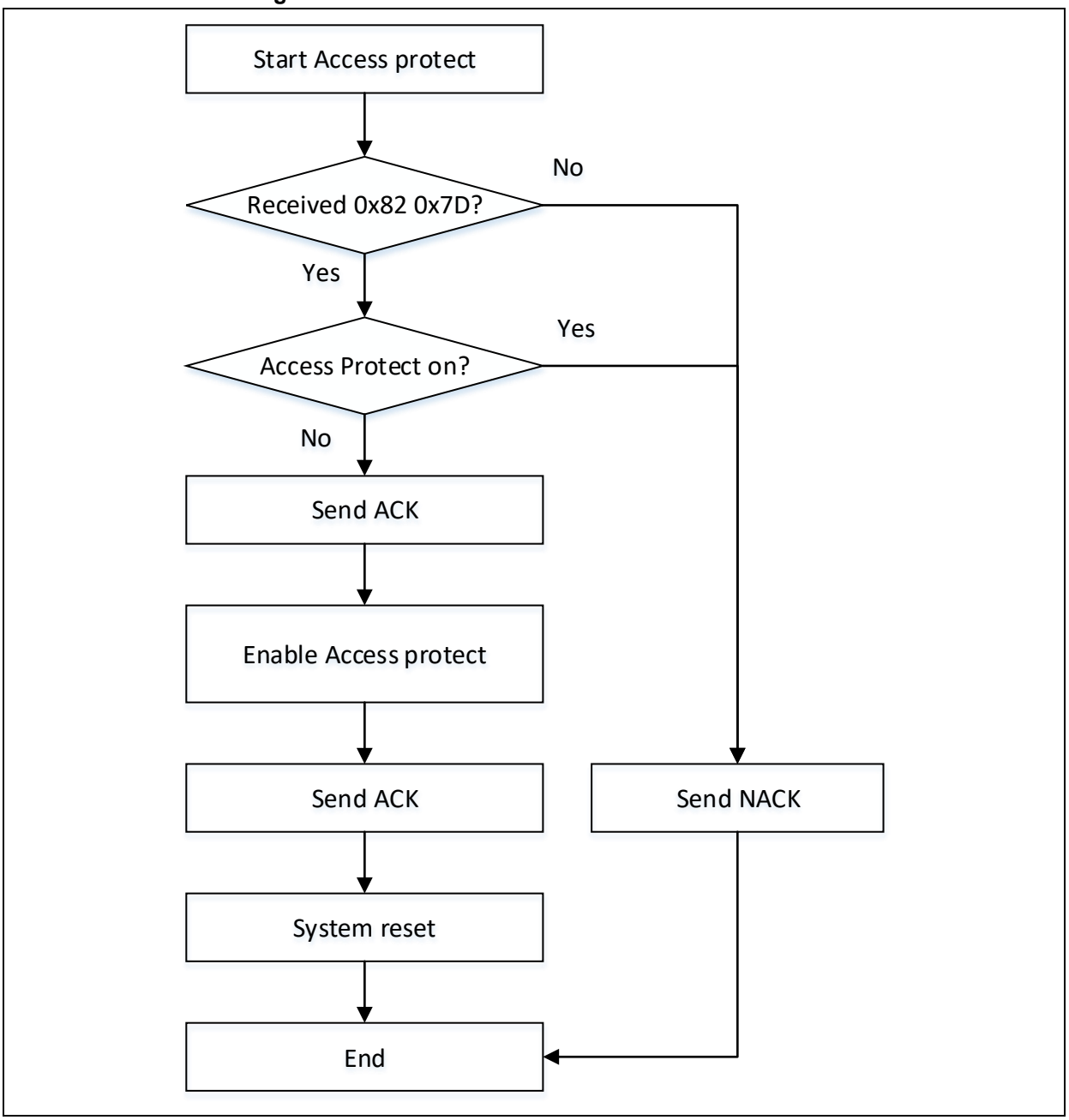


Figure 23 Access Protect flow chart on device side



4.11.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x82	Access protect
2		0x7D	Access protect
	1	ACK/NACK	When a NACK is received, this command stops.
	2	ACK	

4.12 Access Unprotect

Access unprotect is used to unlock memory access protection. Once enabled, the device erases all memory data automatically.

After receiving this command, the device sends an ACK to host, and unlocks access protection, then sends an ACK to host, before performing a system reset.

4.12.1 Access unprotect flow chart

Figure 24 Access Unprotect flow chart on host side

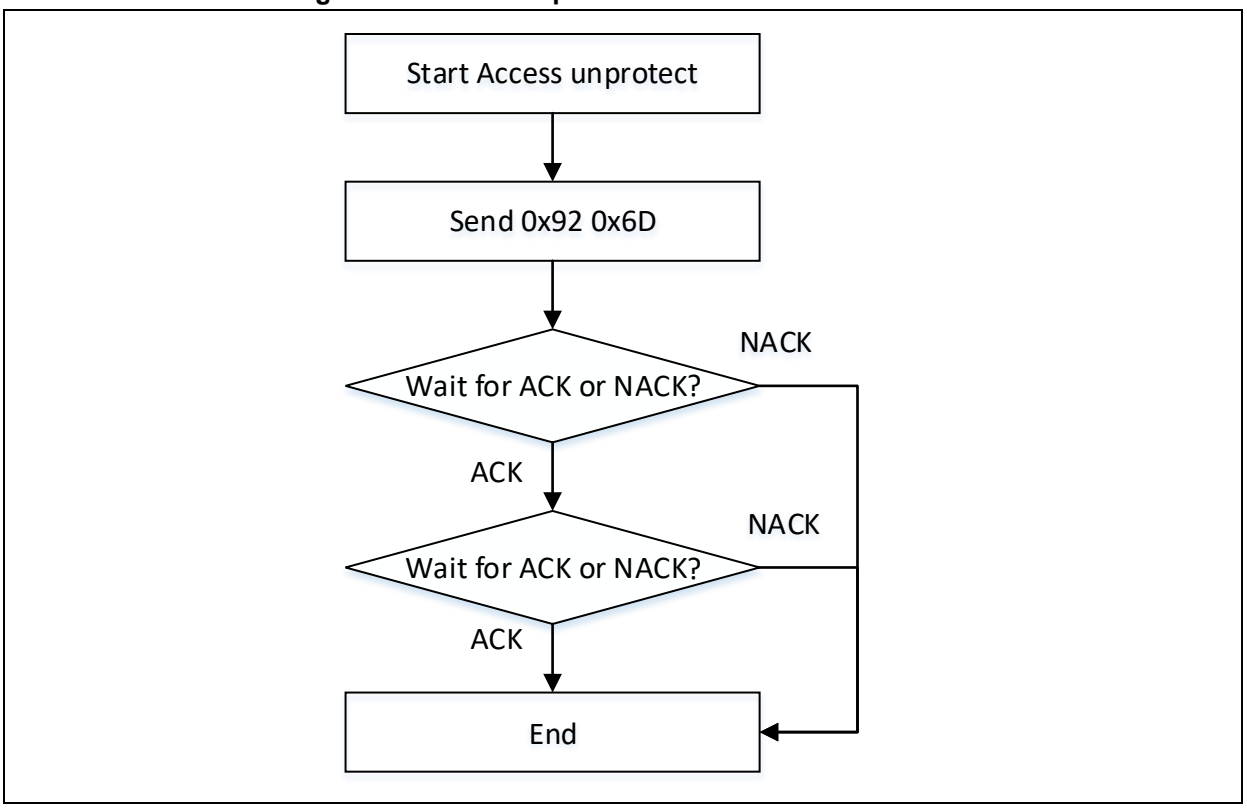
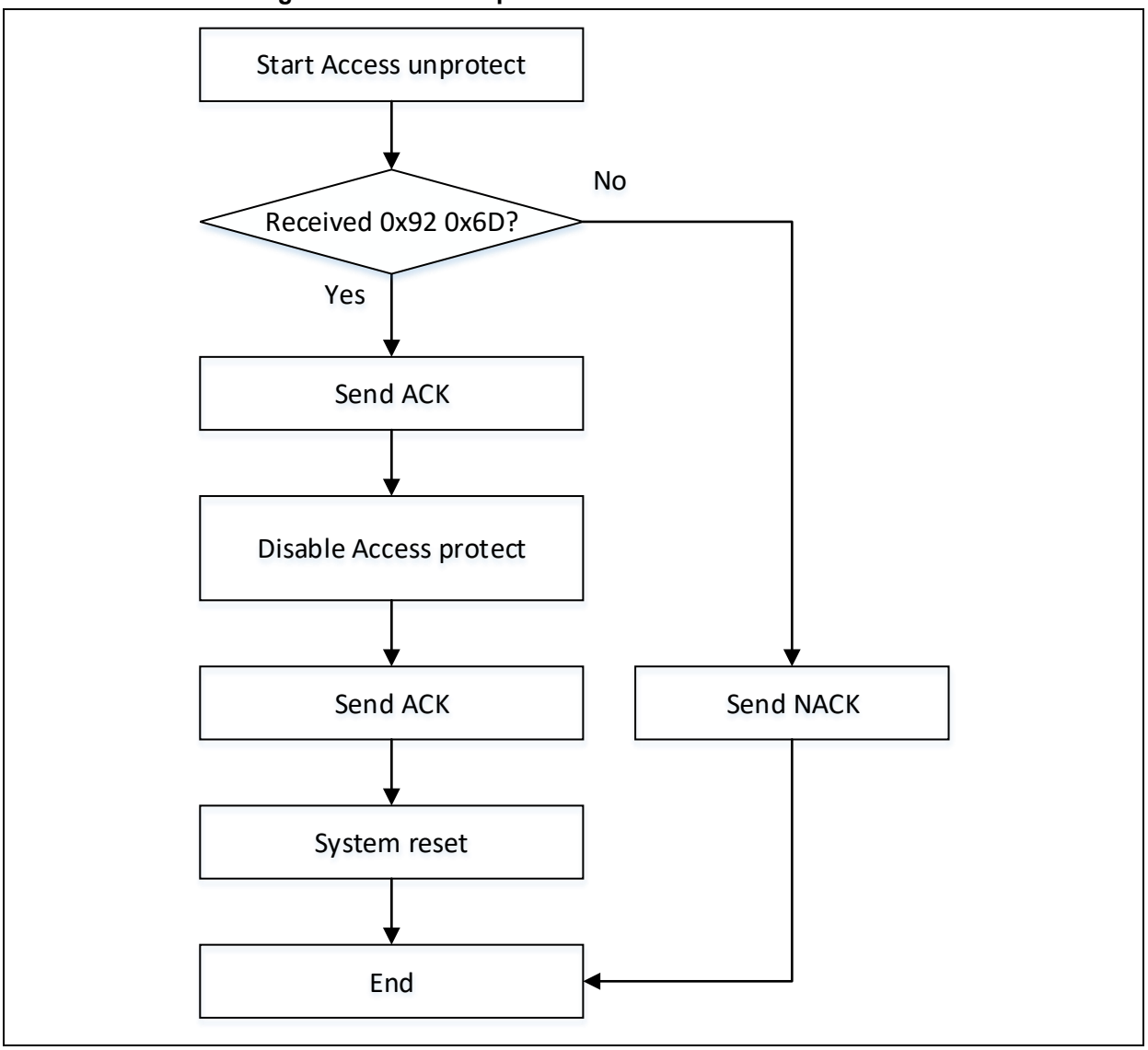


Figure 25 Access Unprotect flow chart on device side



4.12.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0x92	Access unprotect
2		0x6D	Access unprotect
	1	ACK/NACK	When a NACK is received, this command stops
	2	ACK	

4.13 Firmware CRC

The Firmware CRC command is used to check memory data integrity. The host can set memory area for CRC check at sector level, but the memory address must be section aligned.

After receiving this command, the device sends an ACK to host, and waits to receive 4-byte start address and 1-byte checksum. If both address and checksum are valid, the device sends an ACK to host, and then waits to receive the number of sector to calculate decremented by 1 (2 bytes), and 1-byte checksum. For a correct checksum, the device sends an ACK to host, and starts CRC, and then returns 4-byte CRC to host.

The Firmware CRC adopts MPEG-2 CRC algorithm.

4.13.1 Firmware CRC flow chart

Figure 26 Firmware CRC flow chart on host side

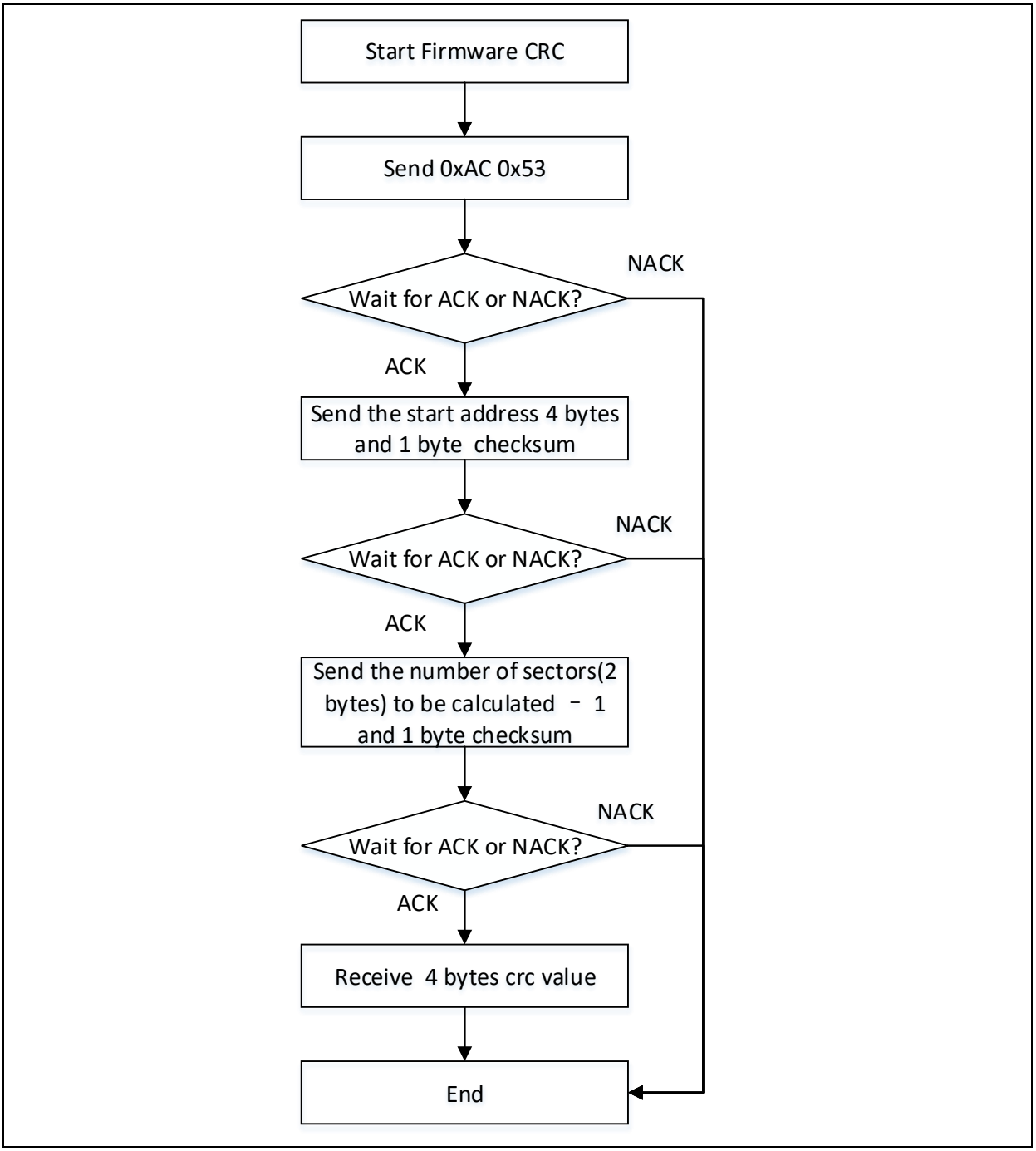
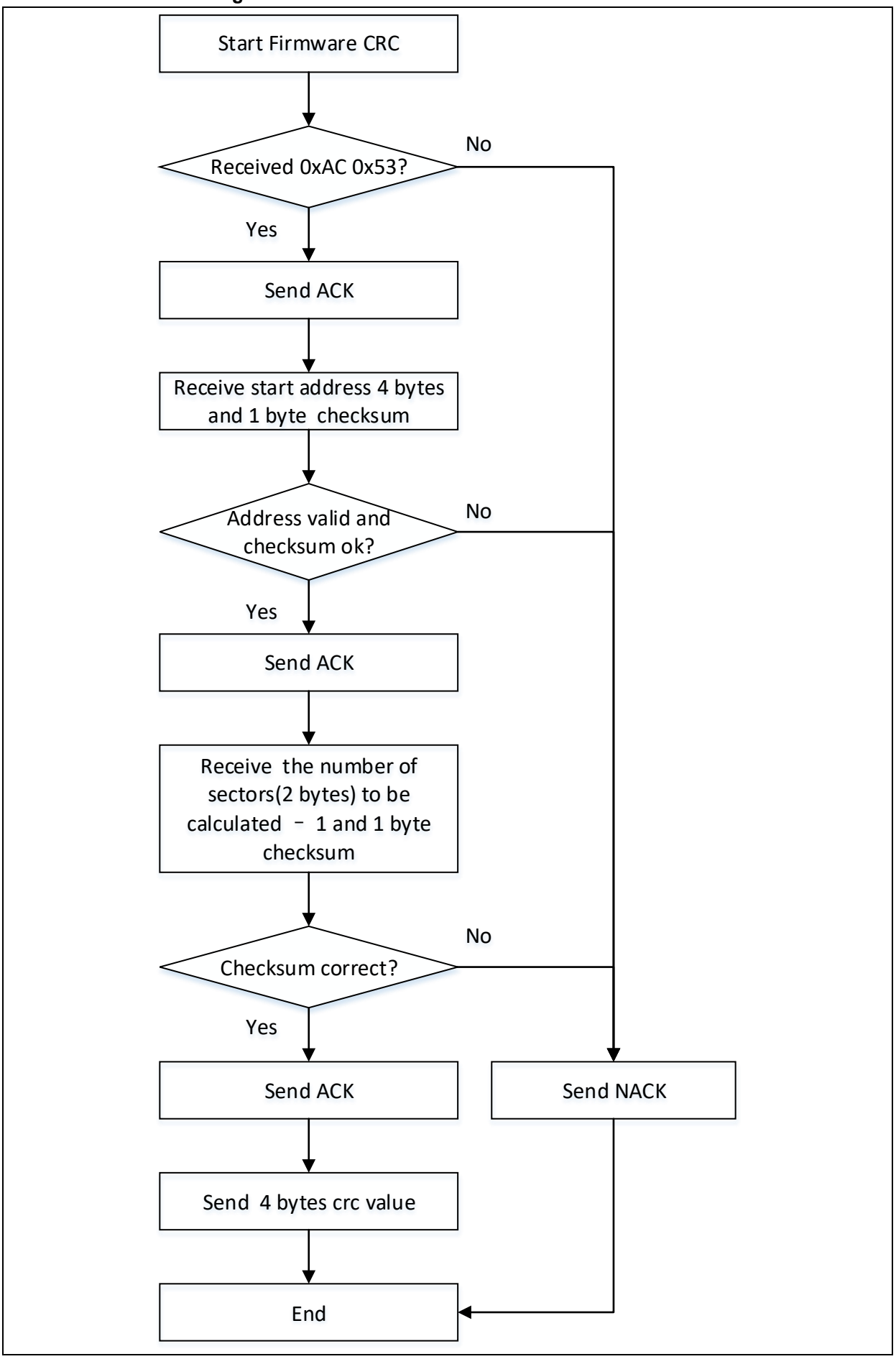


Figure 27 Firmware CRC flow chart on device side



4.13.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xAC	Firmware CRC
2		0x53	Firmware CRC
	1	ACK/NACK	When a NACK is received, this command stops.
3		*	Address (MSB)
4		*	Address
5		*	Address
6		*	Address (LSB)
7		*	Checksum XOR (byte 3~6)
	2	ACK/NACK	When a NACK is received, this command stops.
8		*	Number of sectors – 1 MSB (n)
9		*	Number of sectors – 1 LSB (n)
10		*	Checksum XOR （byte 8~9）
	3	ACK/NACK	When a NACK is received, this command stops.
	4	*	CRC value (MSB)
	5	*	CRC value
	6	*	CRC value
	7	*	CRC value (LSB)

4.14 Enable SPIM

Enable SPIM command is used to expand and enable Bank3. To use this command, SPIM Flash type, Flash size and Flash encryption range must be programmed. Refer to the particular reference manual for more information on SPIM.

Flash type list:

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

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.

After receiving this command, the device sends an ACK to host, and waits to receive 1-byte Flash type, 4-byte Flash size, 4-byte FLASH_FDA, and 1-byte checksum. If both parameters and checksum are valid, the devices sends an ACK to host.

4.14.1 Enable SPIM floc chart

Figure 28 Enable SPIM flow chart on host side

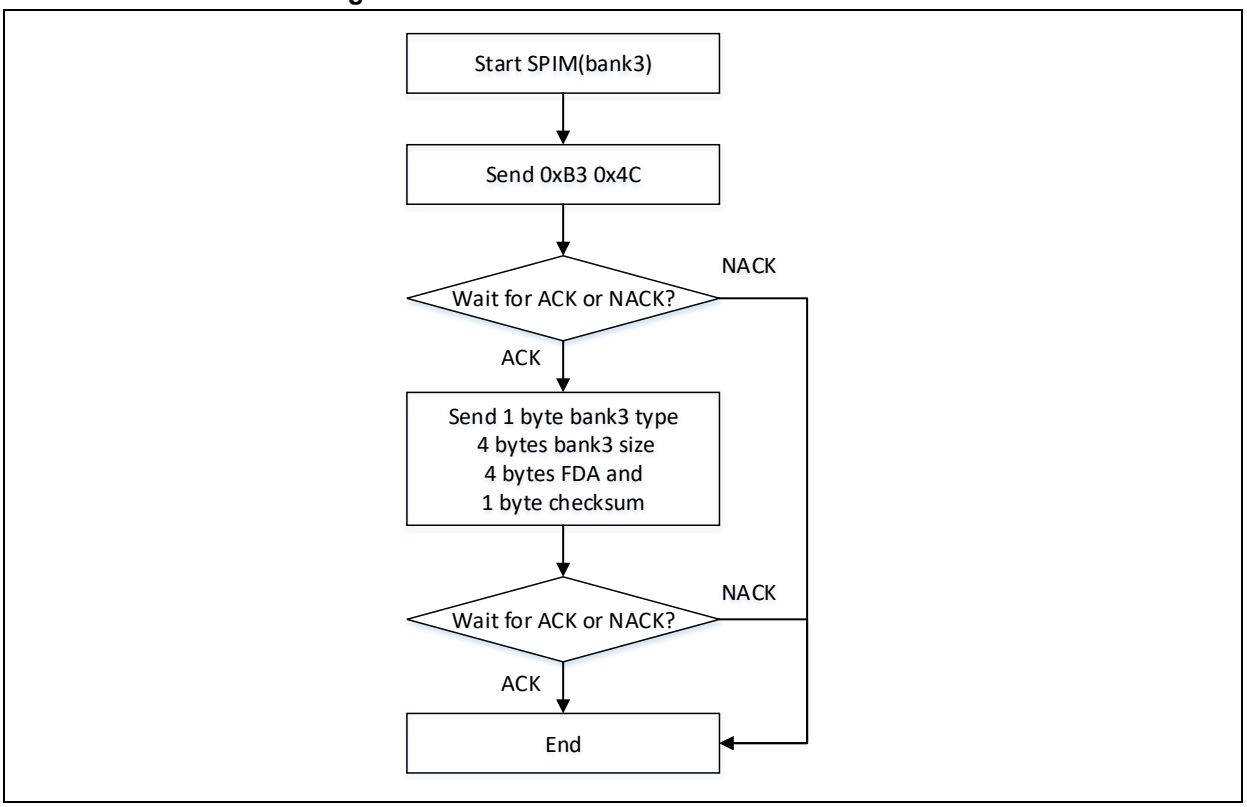
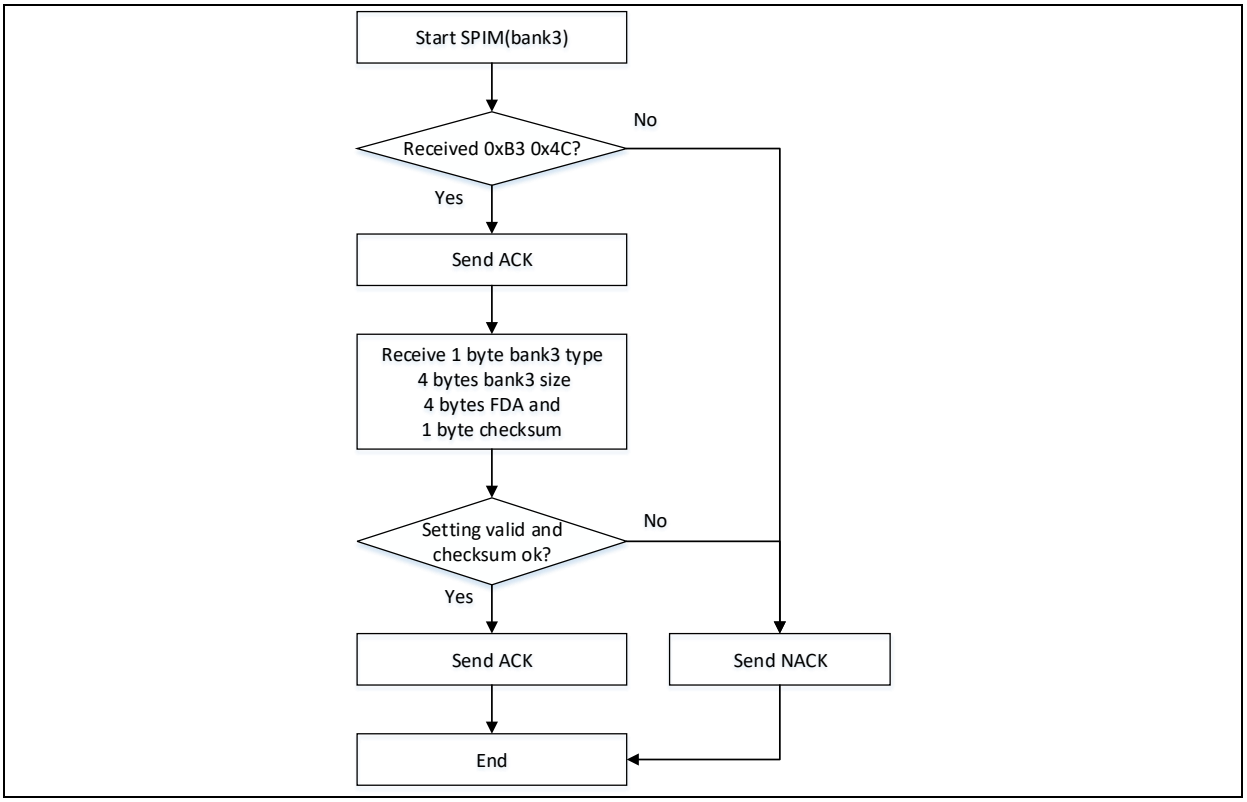


Figure 29 Enable SPIM flow chart on device side



4.14.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xB3	Enable SPIM
2		0x4C	Enable SPIM
	1	ACK/NACK	When a NACK is received, this command stops.
3		*	Flash type
4		*	Flash size (MSB)
5		*	Flash size
6		*	Flash size
7		*	Flash size (LSB)
8		*	Flash FDA (MSB)
9		*	Flash FDA
10		*	Flash FDA
11		*	Flash FDA (LSB)
12		*	Checksum XOR (byte 3~11)
	2	ACK	

4.15 Enable sLib

Enable sLib command is used to enable sLib. Refer to sLib user guideline for more information on sLib.

After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits to receive 4-byte sLib password, 2-byte sLib start sector, 2-byte sLib data/instruction start sector, 2-byte sLib end sector, and 1-byte checksum. if the checksum is valid, the device starts sLib settings, and returns 1-byte setting status later, and then sends an ACK to host. Besides, the sLib setting parameters do not take effect until a system reset.

This command cannot be used when access protection is enabled.

sLib status:

- 1: Enable sLib
- 0: sLib configuration is successful

4.15.1 Enable sLib flow chart

Figure 30 Enable sLib flow chart on host side

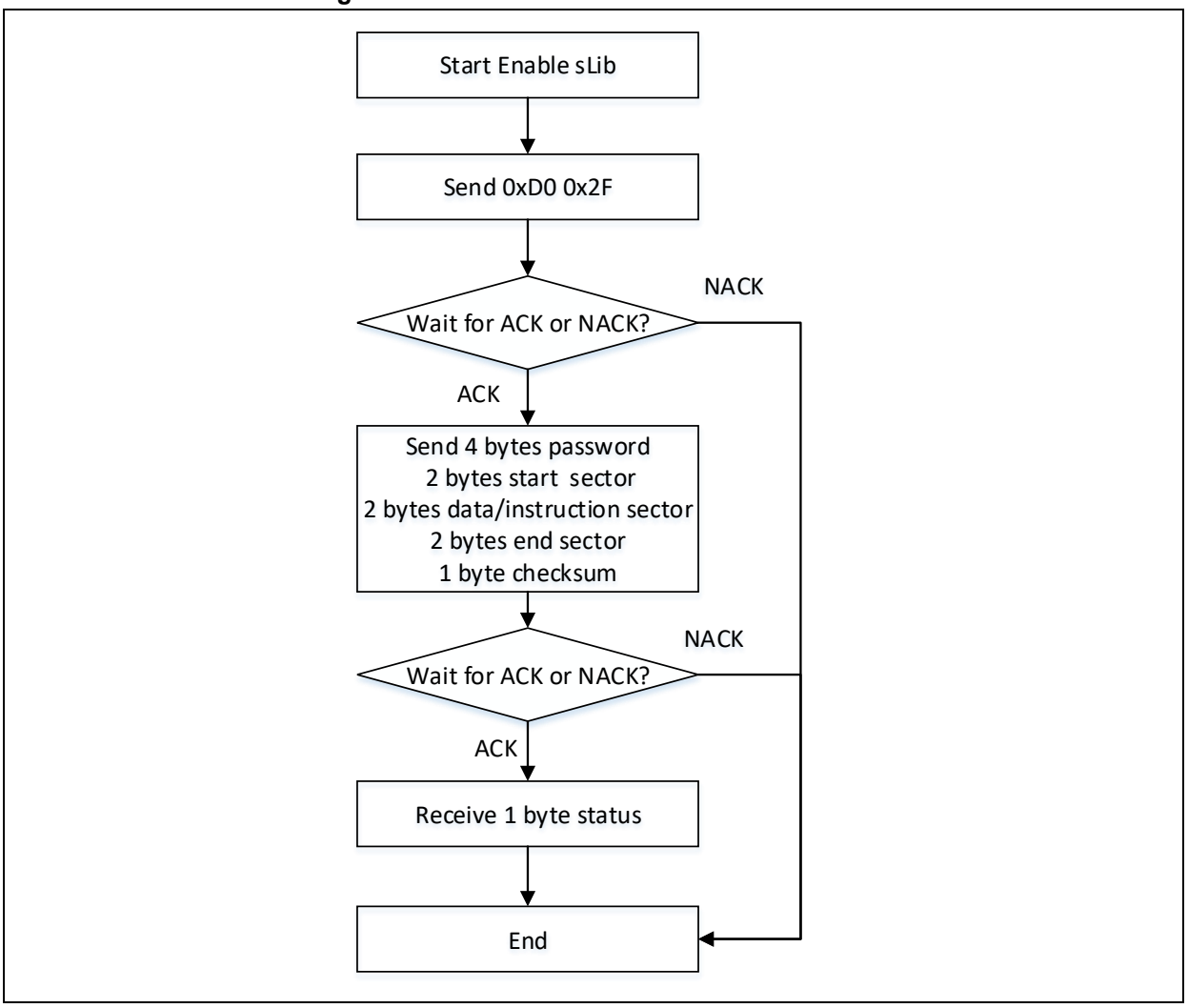
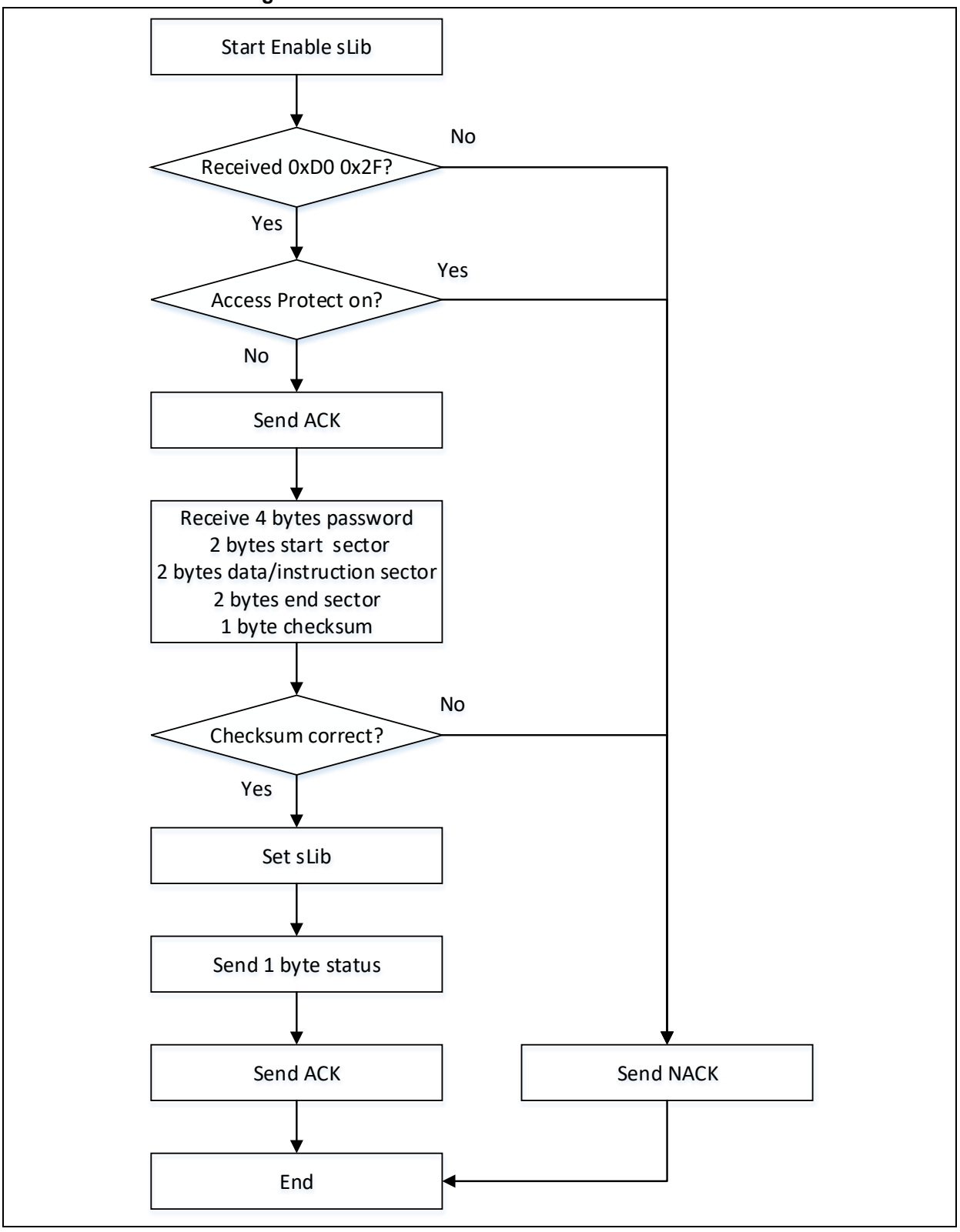


Figure 31 Enable sLib flow chart on device side



4.15.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xD0	Enable sLib
2		0x2F	Enable sLib
	1	ACK/NACK	When a NACK is received, this command stops.
3		*	Password (MSB)
4		*	Password
5		*	Password
6		*	Password (LSB)
7		*	sLib start sector (MSB)
8		*	sLib start sector (LSB)
9		*	sLib data/instruction start sector (MSB)
10		*	sLib data/instruction start sector (LSB)
11		*	sLib end sector (MSB)
12		*	sLib end sector (LSB)
13		*	Checksum XOR byte (3~12)
	2	0/1	sLib setting status
	3	ACK	

4.16 Disable sLib

Disable sLib command is used to disable sLib, in other words, erasing all memory data.
After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits to receive 4-byte password and 1-byte checksum. If the checksum is valid, the device disables the sLib, returns 1-byte sLib disable status and sends an ACK to host.
sLib disable status:

- 1: sLib password wrong
- 0: sLib disabled

This command cannot be used when access protection is enabled.

4.16.1 Disable sLib flow chart

Figure 32 Disable sLib flow chart on host side

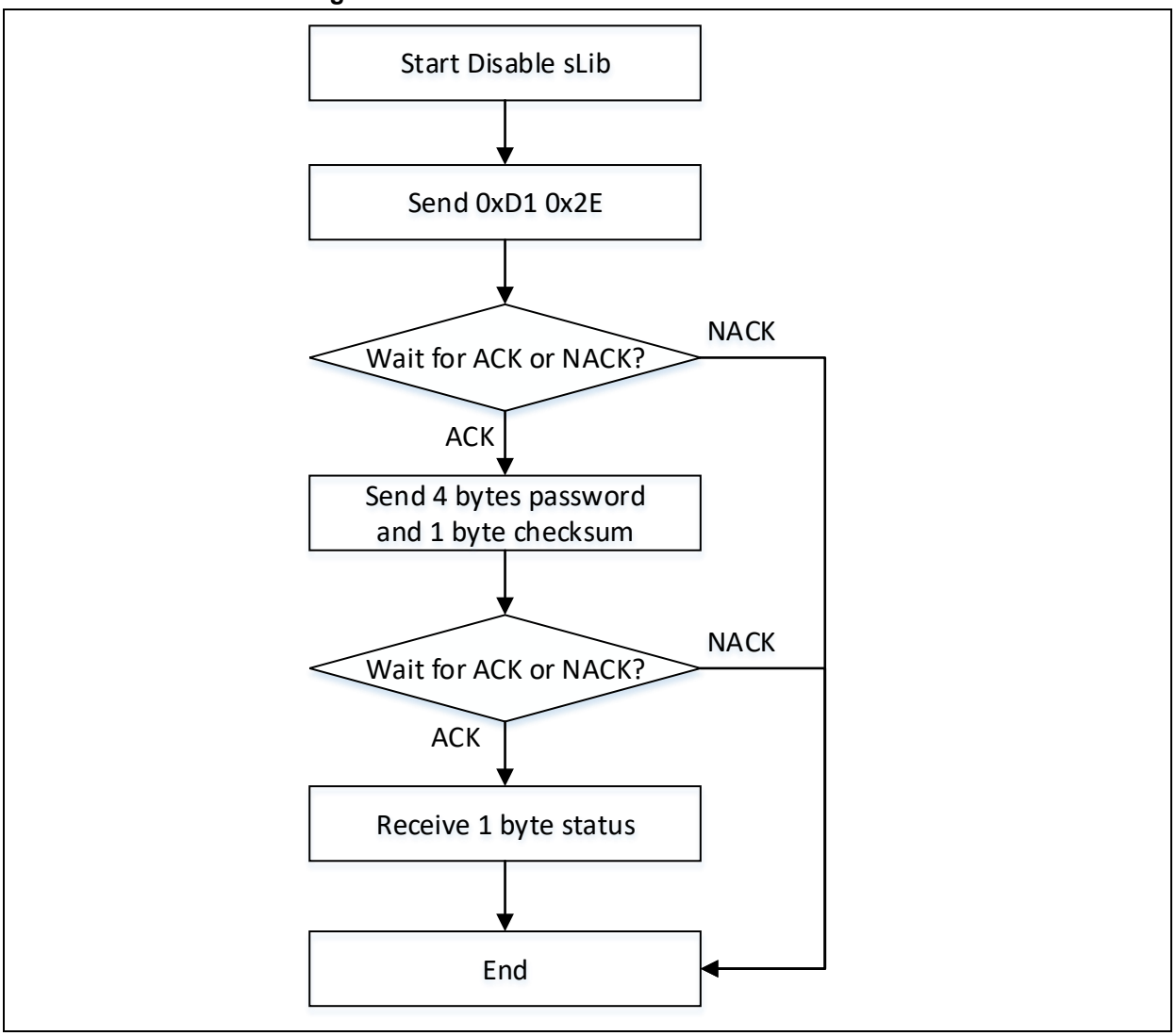
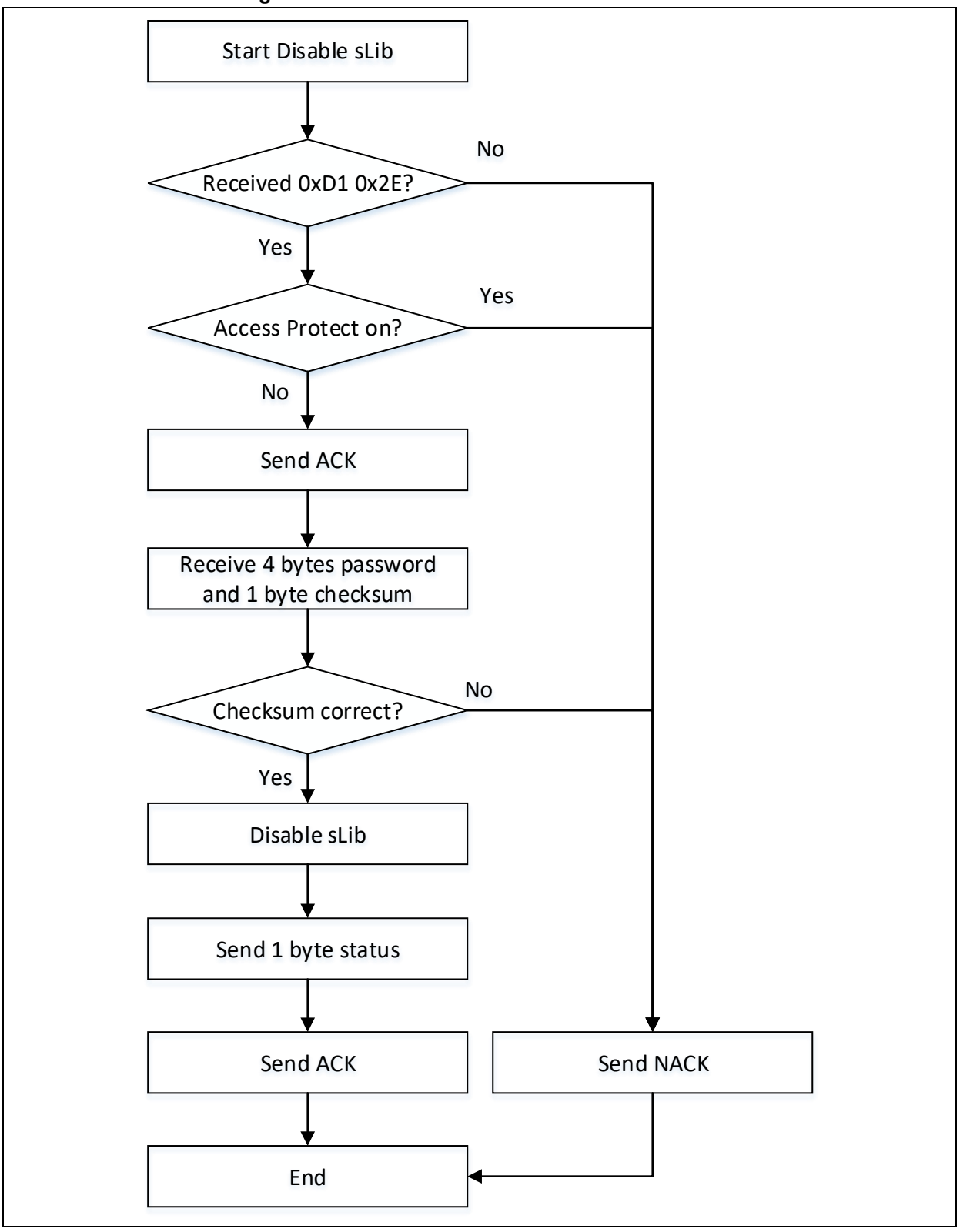


Figure 33 Disable sLib flow chart on device side



4.16.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xD1	Disable sLib
2		0x2E	Disable sLib
	1	ACK/NACK	When a NACK is received, this command stops.
3		*	Password (MSB)
4		*	Password
5		*	Password
6		*	Password (LSB)
	2	0/1	sLib disable status
	3	ACK	

4.17 Get sLib status

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 16-byte data)
After receiving this command, the device sends an ACK to host, and returns 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 12-byte data
After receiving this command, the device sends an ACK to host, and returns 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.*

4.17.1 Get sLib status flow chart

Figure 34 Get sLib status flow chart on host side

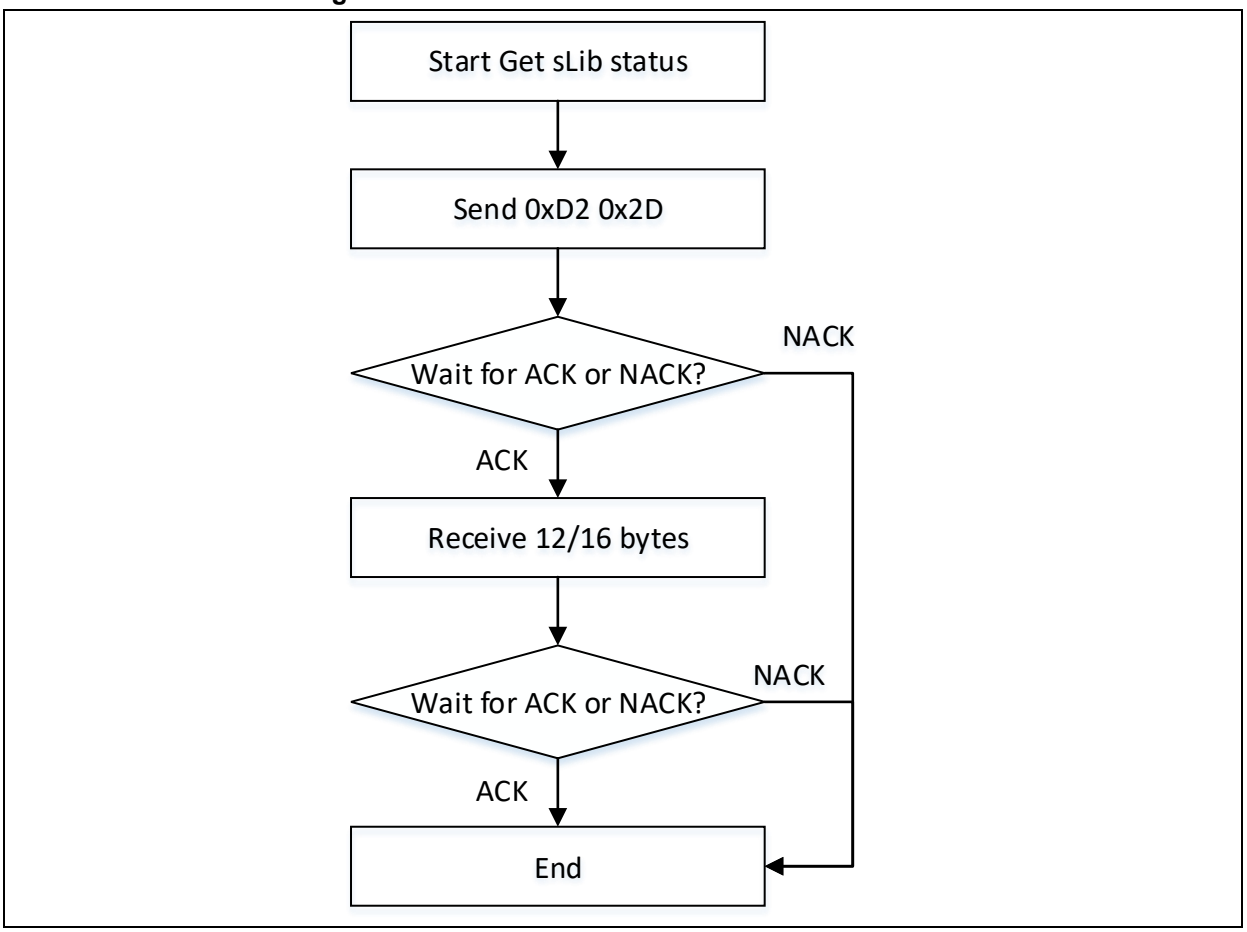
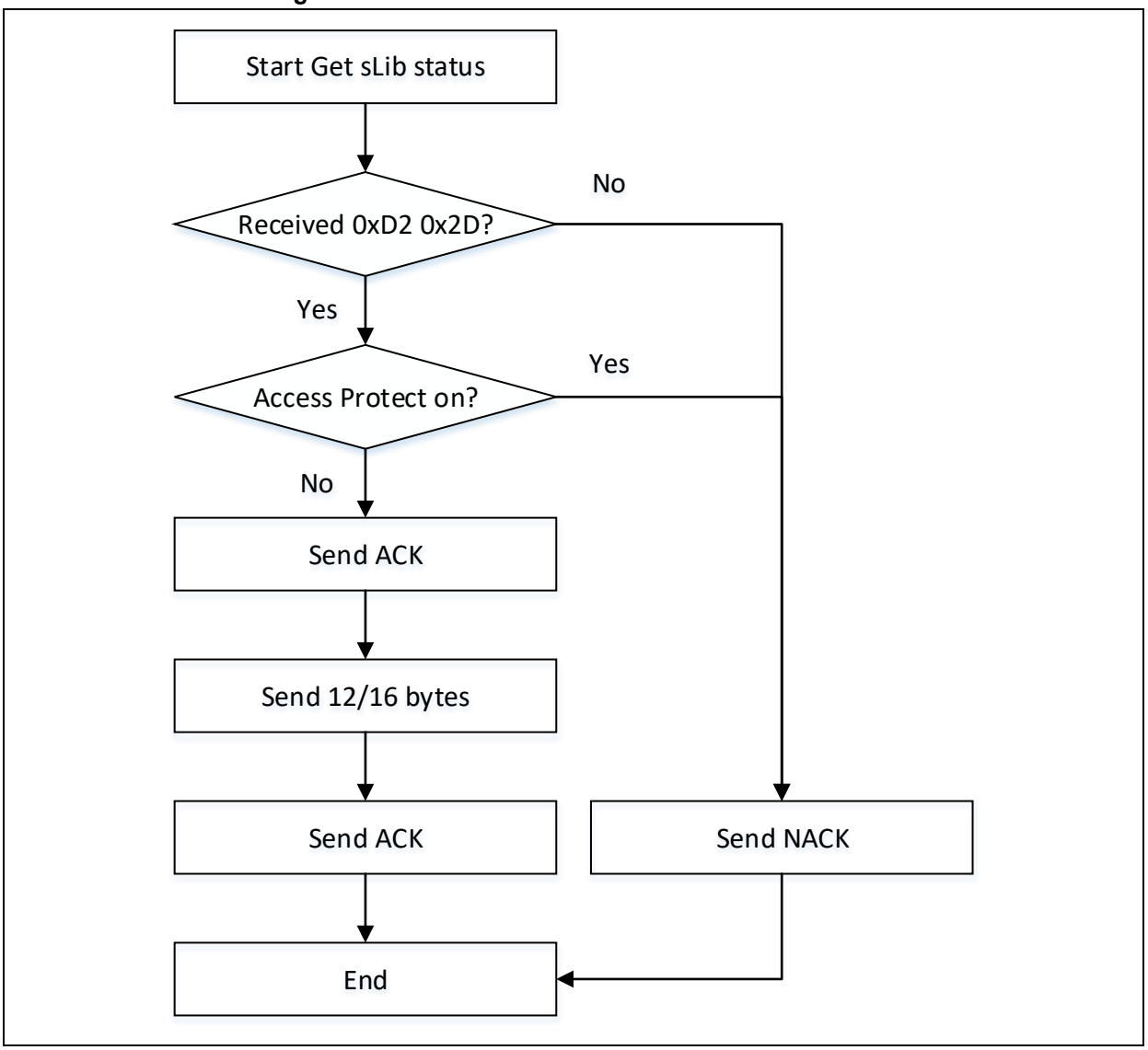


Figure 35 Get sLib status flow chart on device side



4.17.2 Data transfer process on host side

AT32F435xx/AT32F437xx transfer process:

Transmit	Receive	Data	Description
1		0xD2	Get sLib status
2		0x2D	Get sLib status
	1	ACK/NACK	When a NACK is received, this command stops.
	2	*	SLIB_STS0 (MSB)
	3	*	SLIB_STS0
	4	*	SLIB_STS0
	5	*	SLIB_STS0 (LSB)
	6	*	SLIB_STS1 (MSB)
	7	*	SLIB_STS1
	8	*	SLIB_STS1
	9	*	SLIB_STS1 (LSB)
	10	*	SLIB_STS2 (MSB)
	11	*	SLIB_STS2
	12	*	SLIB_STS2
	13	*	SLIB_STS2 (LSB)
	14		SLIB_MISC_STS (MSB)
	15		SLIB_MISC_STS
	16		SLIB_MISC_STS
	17		SLIB_MISC_STS (LSB)
	18	ACK	

Other products:

Transmit	Receive	Data	Description
1		0xD2	Get sLib status
2		0x2D	Get sLib status
	1	ACK/NACK	When a NACK is received, this command stops.
	2	*	SLIB_STS0 (MSB)
	3	*	SLIB_STS0
	4	*	SLIB_STS0
	5	*	SLIB_STS0 (LSB)
	6	*	SLIB_STS1 (MSB)
	7	*	SLIB_STS1
	8	*	SLIB_STS1
	9	*	SLIB_STS1 (LSB)
	10	*	SLIB_MISC_STS (MSB)
	11	*	SLIB_MISC_STS
	12	*	SLIB_MISC_STS
	13	*	SLIB_MISC_STS (LSB)
	14	ACK	

4.18 SPIM Remap

SPIM Remp command is used to configure external IO multiplexed function for SPIM (bank3). After receiving this command, the device sends an ACK to host, and waits to receive 1-byte Remap flag and its checksum. If the checksum is valid, the device remains Remap status, and sends an ACK to host.

Remap Flag：

- 0: SCK/PB1 CS/PA8 IO0/PA11 IO1/PA12 IO2/PB7 IO3/PB6
- 1: SCK/PB1 CS/PA8 IO0/PB10 IO1/PB11 IO2/PB7 IO3/PB6

4.18.1 SPIM remap flow chart

Figure 36 SPIM Remap flow chart on host side

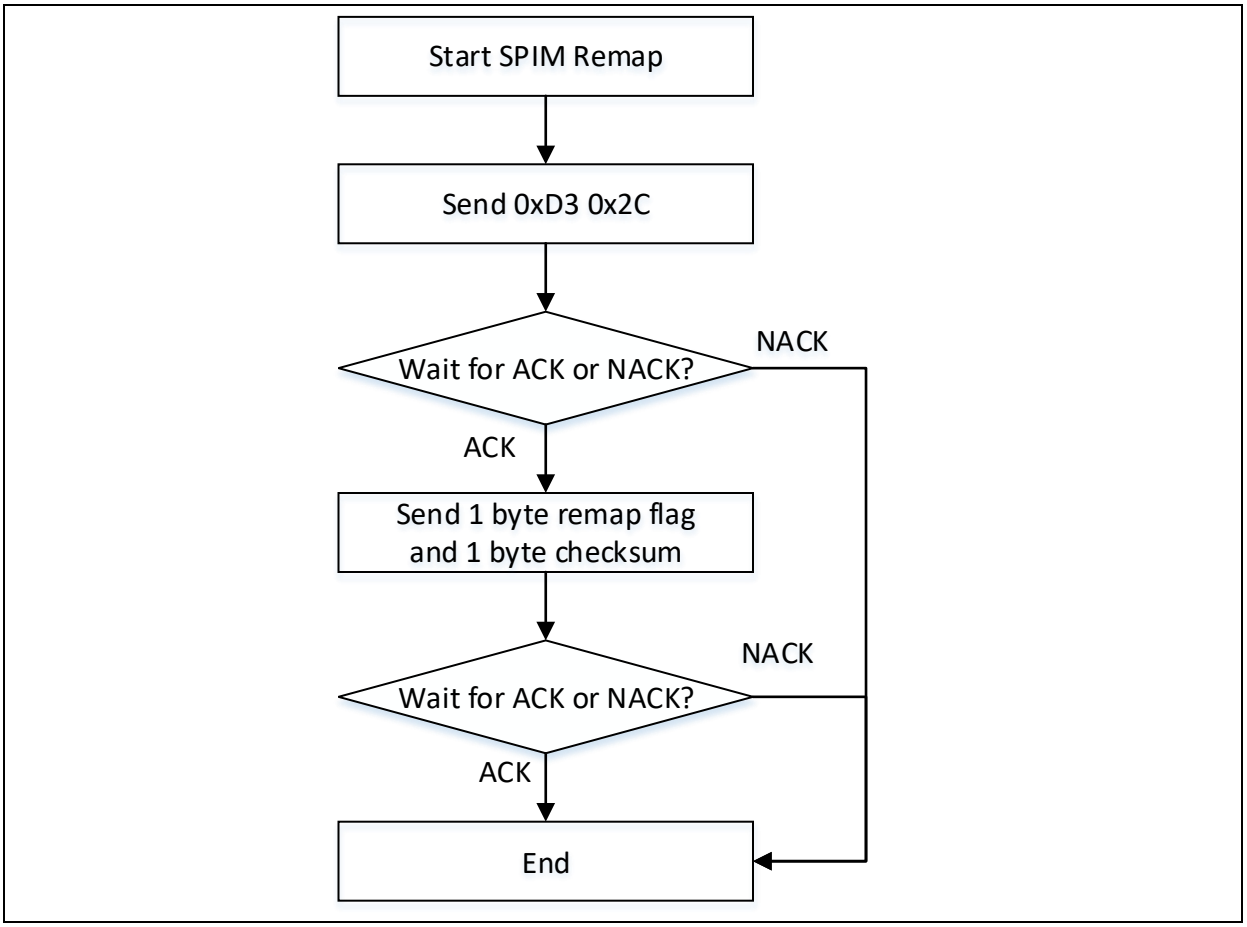
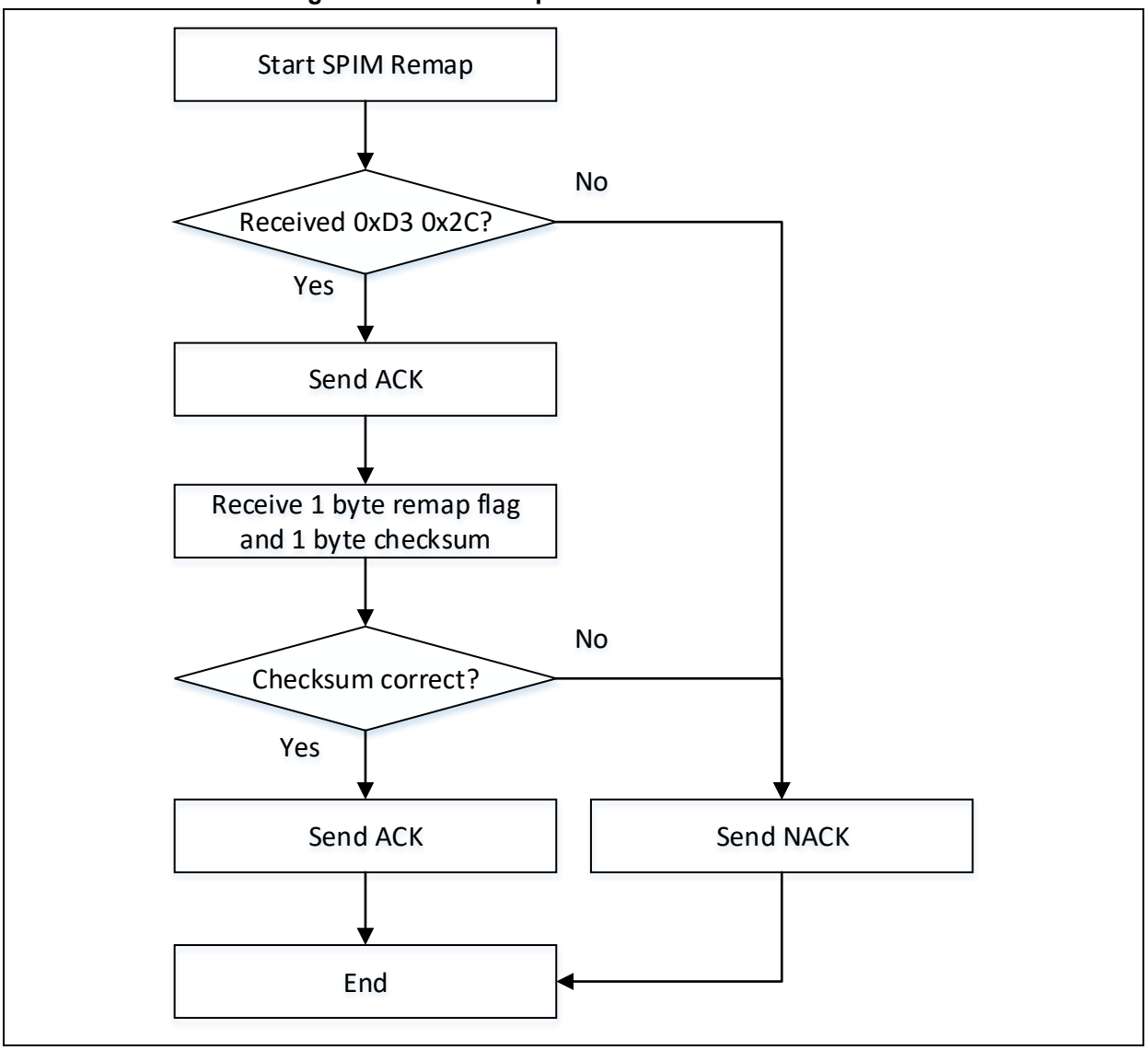


Figure 37 SPIM Remap flow chart on device side



4.18.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xD3	SPIM Remap
2		0x2C	SPIM Remap
	1	ACK/NACK	When a NACK is received, this command stops.
3		0/1	Remap Flag
4		*	Checksum XOR byte3
	2	ACK	

4.19 Reset Device

Reset Device is used for a system reset by device.

After receiving this command, the device sends ACK to host twice, before performing a system reset.

4.19.1 Reset device flow chart

Figure 38 Reset Device flow chart on host side

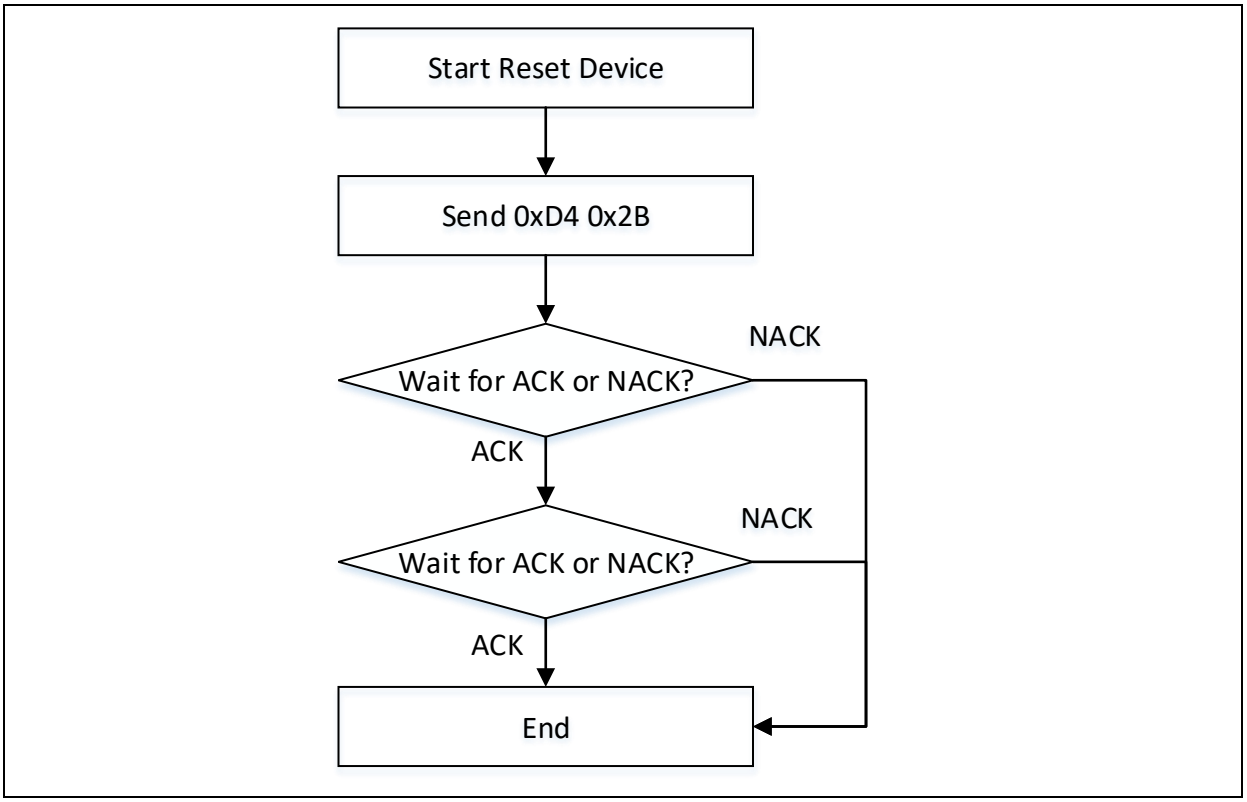
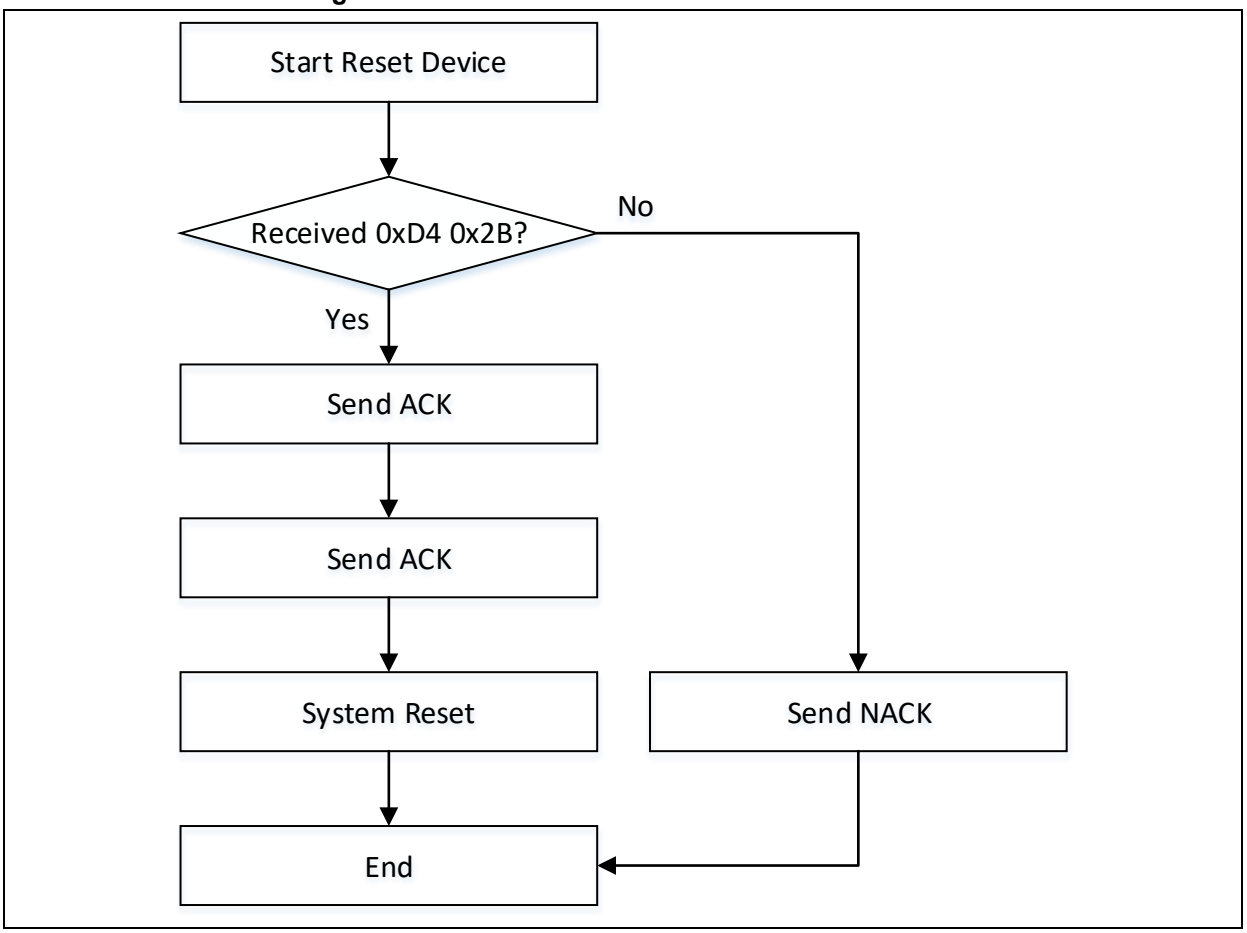


Figure 39 Reset Device flow chart on device side



4.19.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xD4	Reset Device
2		0x2B	Reset Device
	1	ACK/NACK	When a NACK is received, this command stops.
	2	ACK	

4.20 Advanced Access Protect

Advanced Access Protect is used to enable high-level access protection. Refer to the particular reference manual for more information on high-level access protection.

After receiving this command, if access protection is disabled, the device sends an ACK to host, and waits to receive 1-byte flag (it can be any value) and 1-byte checksum and configures advanced access protection, and sends an ACK to host, before performing a system reset.

This command cannot be used when access protection is enabled.

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

4.20.1 Advanced access protection flow chart

Figure 40 Advanced Access Protect flow chart on host side

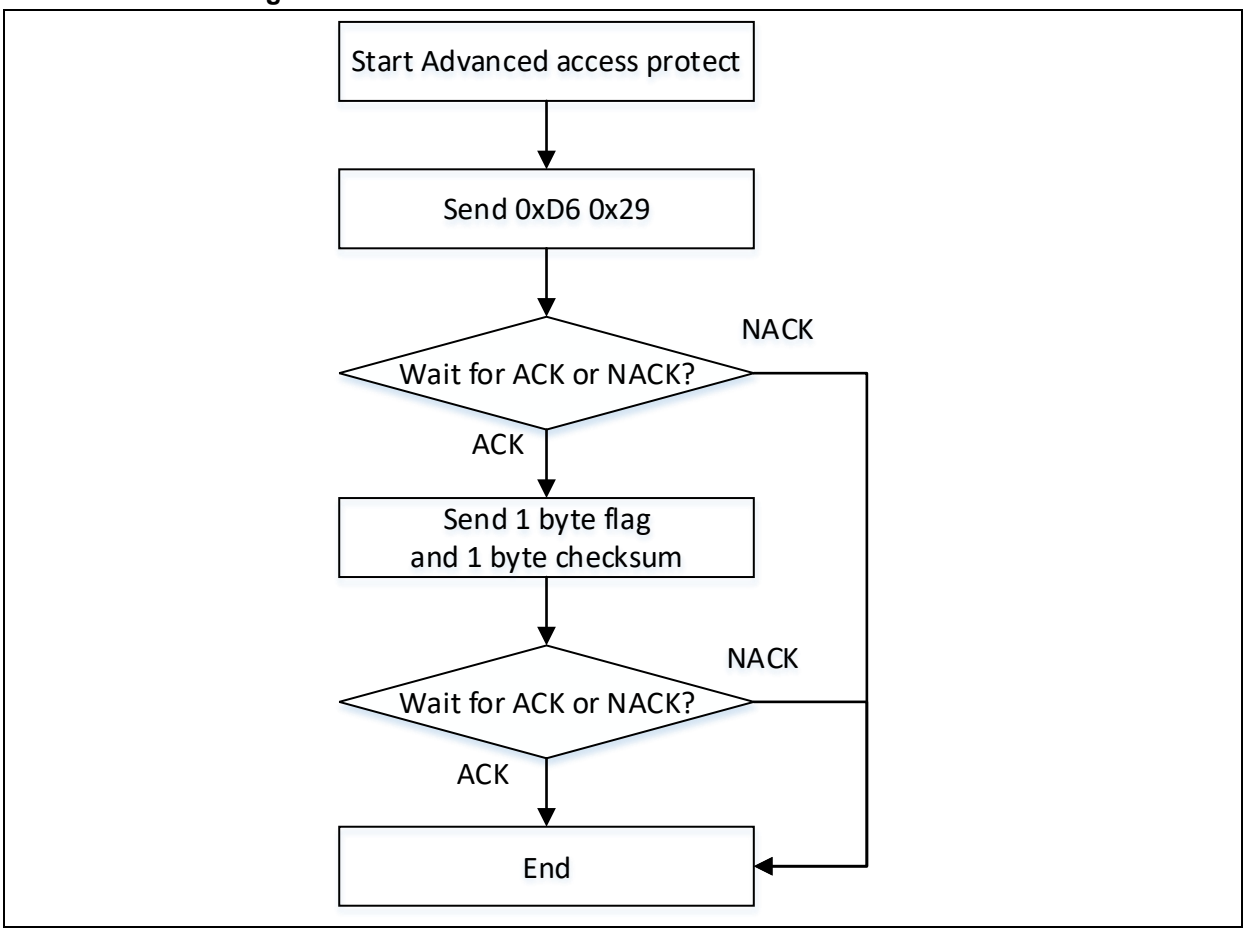
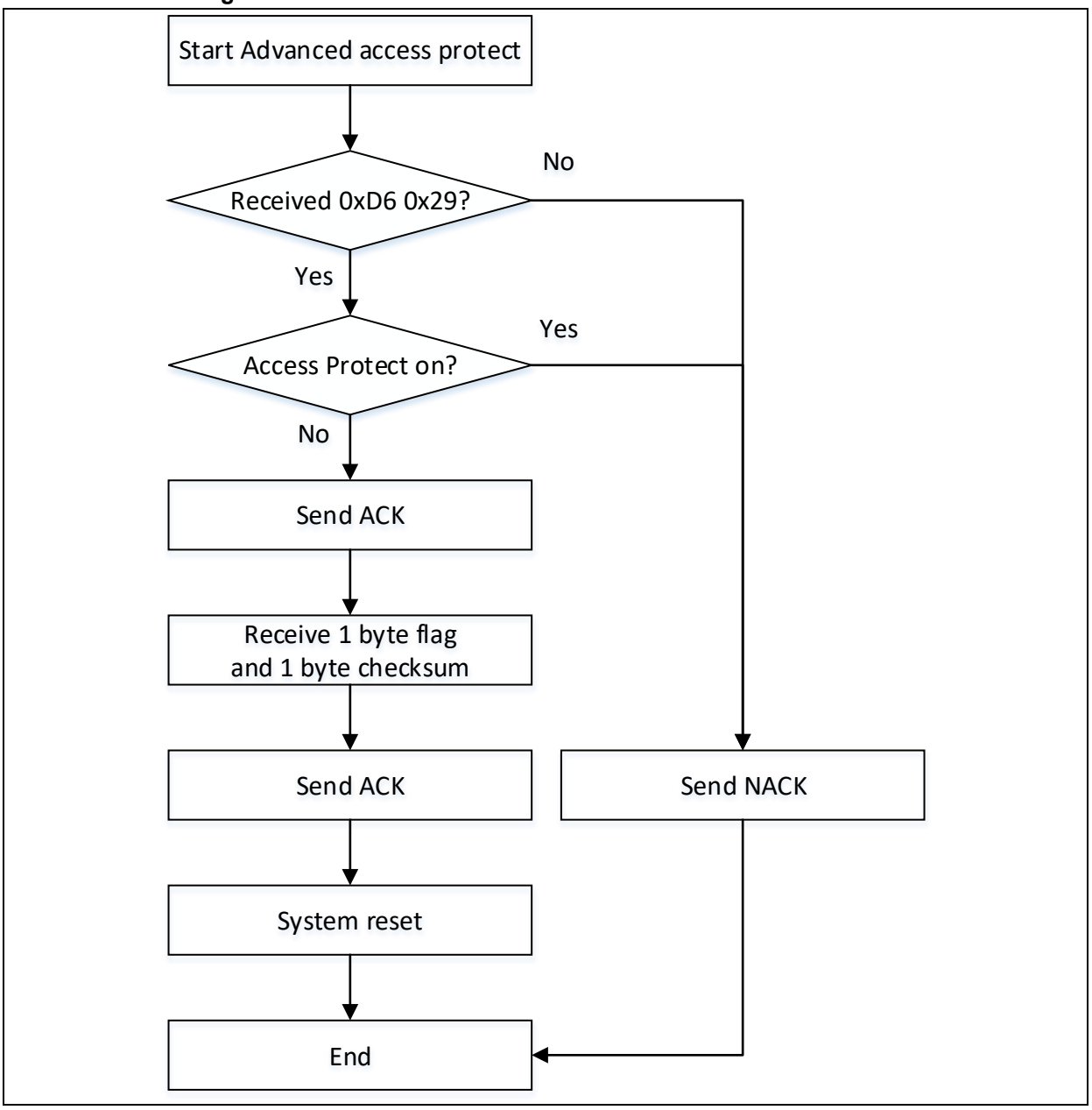


Figure 41 Advanced Access Protect flow chart on device side



4.20.2 Data transfer process on host side

Transmit	Receive	Data	Description
1		0xD3	Advanced access protect
2		0x2C	Advanced access protect
	1	ACK/NACK	When a NACK is received, this command stops.
3		*	flag
4		*	Checksum XOR byte3
	2	ACK	

5 Revision history

Table 3 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.