

Introduction

The STSPIN32F0A is a system-in-package providing an integrated solution suitable for driving three-phase BLDC motors using different driving modes. The integrated MCU (STM32F031C6 with extended temperature range, suffix 7 version) allows downloading the firmware on the field through the serial interface thanks to the embedded bootloader.

The bootloader is stored in the internal boot ROM (system memory) of the microcontroller. Its main task is to download the application program to the internal Flash memory through the serial peripheral USART.

This application note presents the general concept of the bootloader. It describes the supported peripherals and hardware requirements to be considered when using the bootloader of the STM32 microcontroller embedded in the STSPIN32F0A and the bootloader protocol based on the USART.

Contents

- 1 Related documents 4**
- 2 STSPIN32F0A microcontroller unit 4**
 - Memories and boot mode 4
- 3 General bootloader description 6**
 - 3.1 Bootloader activation 6
 - 3.2 Bootloader identification 6
 - 3.3 Hardware connection requirements 7
 - 3.4 Bootloader memory management 7
- 4 Bootloader configuration and selection 8**
- 5 Bootloader timing 10**
 - 5.1 Bootloader startup timing 10
 - 5.2 USART connection timing 11
- 6 USART bootloader code sequence 12**
 - 6.1 Choosing the baud rate 13
 - 6.2 USART bootloader command set 14
 - Communication safety 15
- 7 Revision history 16**

List of tables

Table 1.	STSPIN32F0A embedded bootloader	6
Table 2.	STSPIN32F0A bootloader parameters	6
Table 3.	STSPIN32F0A configuration in system memory boot mode.	8
Table 4.	USART bootloader minimum timings	11
Table 5.	USART bootloader commands	14
Table 6.	Document revision history	16

List of figures

Figure 1.	USART connection	7
Figure 2.	STSPIN32F0A bootloader selection diagram	9
Figure 3.	Bootloader startup timing description	10
Figure 4.	USART connection timing description	11
Figure 5.	USART bootloader code sequence diagram	12

1 Related documents

Please refer to the following documents available from www.st.com:

- STSPIN32F0A datasheet
- STM32F031C6 datasheet (suffix 7 version)
- Application notes
 - AN2606: STM32 microcontroller system memory boot mode
 - AN3155: USART protocol used in the STM32 bootloader

2 STSPIN32F0A microcontroller unit

The integrated MCU is the STM32F031C6 with following main characteristics

- Core: ARM® 32-bit Cortex®-M0 CPU, frequency up to 48 MHz
- Memories: 4 kB of SRAM, 32 kB of Flash memory
- CRC calculation unit
- Up to 16 fast I/Os
- Advanced control timer dedicated for PWM generation
- Up to 5 general purpose timers
- 12-bit ADC (up to 9 channels)
- Communication interfaces: I²C, USART, SPI
- Serial wire debug (SWD)
- Extended junction temperature range: -40 to 125 °C

Memories and boot mode

The device has the following features

- 4 kB of the embedded SRAM accessed (read/write) at the CPU clock speed with 0 wait states and featuring embedded parity checking with an “exception generation” for fail-critical applications
- The non-volatile memory is divided into two arrays
 - 32 kB of the embedded Flash memory for programs and data
 - Option bytes

The option bytes are used to write-protect the memory (with 4 kB granularity) and/or readout-protect the whole memory with the following options

- Level 0: no readout protection
- Level 1: memory readout protection, the Flash memory cannot be read from or written to if either debug features are connected or the boot in the RAM is selected
- Level 2: chip readout protection, debug features (Cortex-M0 serial wire) and the boot in the RAM selection disabled

At the startup the BOOT0 pin and the boot selector option bit are used to select one of the three boot options

- Boot from the user Flash memory
- Boot from the system memory
- Boot from the embedded SRAM

The boot loader is located in the system memory. It is used to reprogram the Flash memory by using the USART on pins PA14/PA15.

The main Flash memory is aliased in the boot memory space (0x00000000), but still accessible from its original memory space (0x08000000). In other words, the Flash memory contents can be accessed starting from the address 0x00000000 or 0x08000000.

3 General bootloader description

3.1 Bootloader activation

The STSPIN32F0A bootloader is activated by applying the following pattern

- Boot0(pin) = 1 and nBoot1(bit) = 1

The system memory boot mode can be exited by getting out from bootloader activation condition and generating hardware reset or using the Go command to execute the user code.

3.2 Bootloader identification

The STSPIN32F0A bootloader support USART peripherals to download the code to the internal Flash memory. The bootloader identifier (ID) provides information about the supported serial peripherals: ID equal to 0x10 means that it is the version of the device bootloader using one USART only.

Table 1. STSPIN32F0A embedded bootloader

Supported serial peripheral	Bootloader ID		Bootloader protocol version
	ID	Memory location	
USART1	0x10	0x1FFFF7A6	USART (V 3.1)

The bootloader (protocol) version can be retrieved using the bootloader Get Version command.

The bootloader protocol's command set and sequences for each serial peripheral are the same for all STM32 devices. However, some parameters depend on the device and bootloader version

- PID (Product ID)
- Valid RAM addresses (RAM area used during bootloader execution is not accessible) accepted by the bootloader when the Read Memory, Go and Write Memory commands are requested.
- System Memory area

[Table 2](#) shows the values of these parameters for the STSPIN32F0A embedded microcontroller.

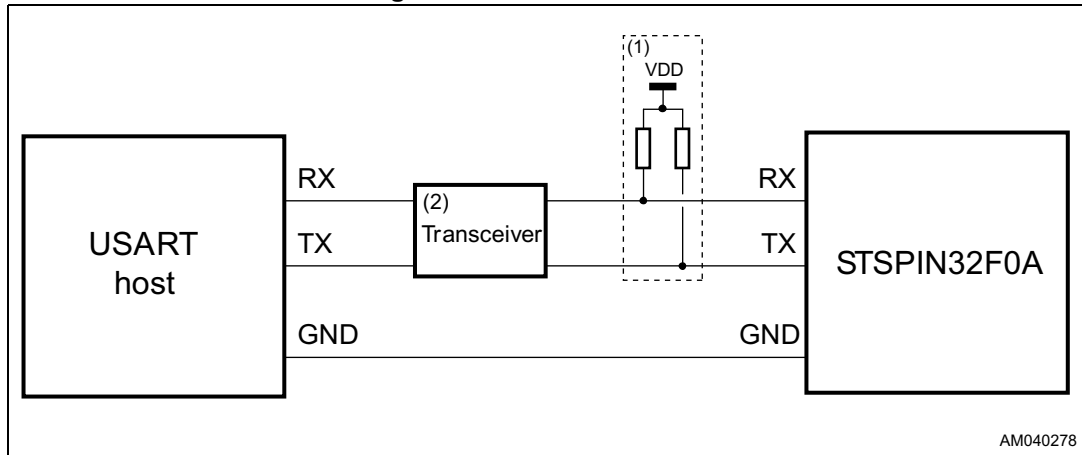
Table 2. STSPIN32F0A bootloader parameters

PID	Bootloader ID	RAM memory	System memory
0x444	0x10	0x20000800 - 0x20000FFF	0x1FFFE000 - 0x1FFFF7FF

3.3 Hardware connection requirements

To use the USART bootloader, the host has to be connected to the PA15 (RX) and PA14 (TX) pins of the USART interface.

Figure 1. USART connection



1. Pull-up resistors should be added, if pull-up resistors are not connected in the host side. VDD = 3.3 V and 100 k Ω resistors typically, this values depend on the application and host.
2. A transceiver could be required to adapt the voltage level between the device and host (e.g. RS232).

3.4 Bootloader memory management

All write operations using bootloader commands must only be word-aligned (the address should be a multiple of 4). The number of data to be written must also be a multiple of 4 (non-aligned half page write addresses are accepted).

4 Bootloader configuration and selection

The STSPIN32F0A bootloader is activated by applying pattern

- Boot0(pin) = 1 and nBoot1(bit) = 1

[Table 3](#) shows the hardware resources used by this bootloader.

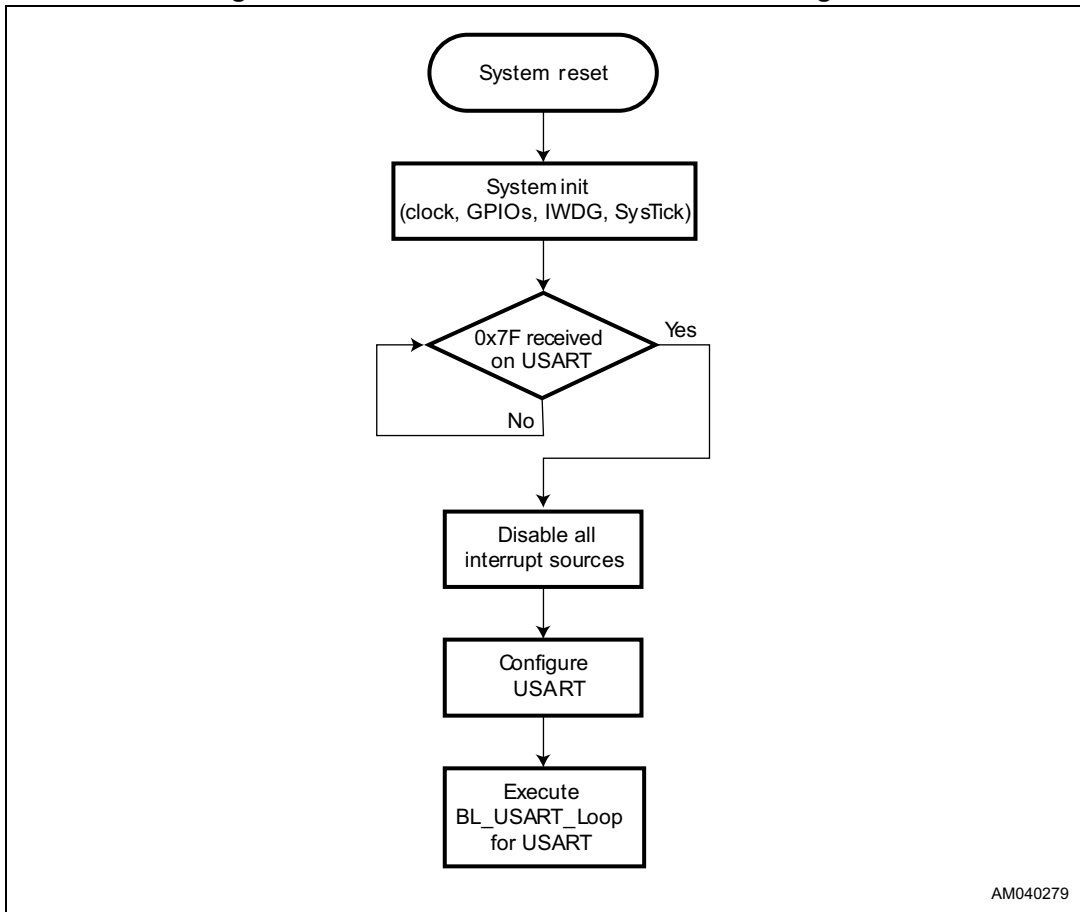
Table 3. STSPIN32F0A configuration in system memory boot mode

Feature/peripheral	State	Comment
RCC	HIS enabled	The system clock frequency is 24 MHz (using PLL clocked by HSI).
1 Flash Wait State		
RAM	-	2 Kbyte starting from the address 0x20000000 are used by the bootloader firmware.
System memory	-	3 Kbyte starting from the address 0x1FFFE000 contains the bootloader firmware.
IWDG	-	The independent watchdog (IWDG) prescaler is configured to its maximum value. It is periodically refreshed to prevent the watchdog reset in case the user previously enabled the hardware IWDG option.
USART1	Enabled	Once initialized, the USART1 configuration is 8 bits, even parity and 1 Stop bit.
USART1_RX pin	Input	PA15 pin: USART1 in the reception mode
USART1_TX pin	Output	PA14 pin: USART1 in the transmission mode
SysTick timer	Enabled	Used to automatically detect the serial baud rate from the host.

The system clock is derived from the embedded internal high-speed RC; no external quartz is required for the bootloader execution.

Note: After the device has booted in the bootloader mode, serial wire debug (SWD) communication is no longer possible until the system is reset. This is because the SWD uses the PA14 pin (SWCLK) which is already used by the bootloader (USART1_TX).

Figure 2. STSPIN32F0A bootloader selection diagram



Note: For this version the “know limitation” is that for the USART interface, two consecutive NACKs instead of 1 NACK are sent when a Read Memory or Write Memory command is sent and the RDP level is active.

5 Bootloader timing

This section presents the typical timings of the bootloader firmware that should be used to ensure correct synchronization between the host and STSPIN32F0A device.

Two types of timings will be described herein

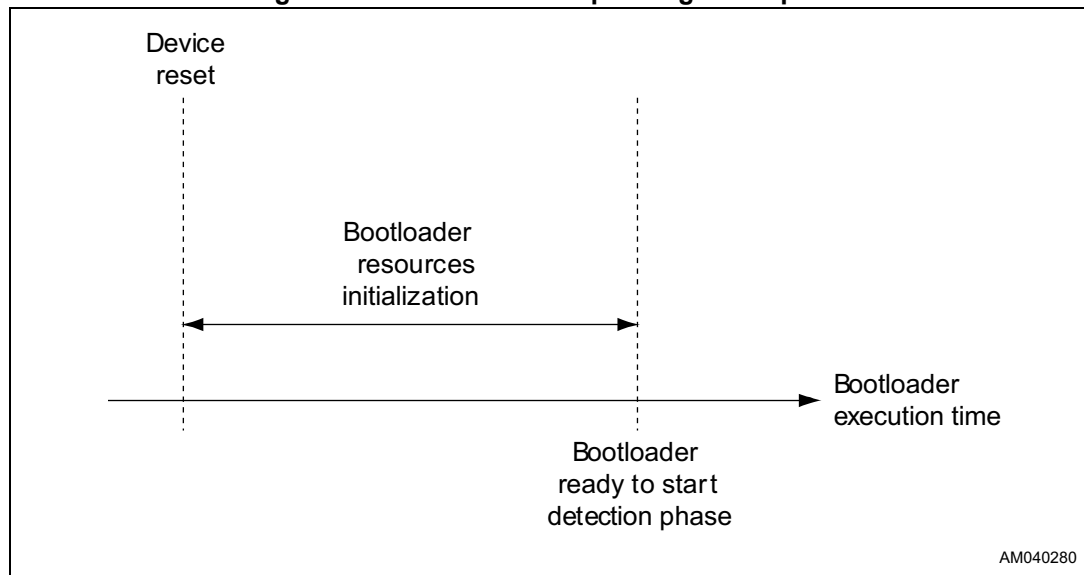
- STSPIN32F0A device bootloader resources initialization duration
- Communication interface selection duration

After these timings, the bootloader is ready to receive and execute host commands.

5.1 Bootloader startup timing

After the bootloader reset, the host should wait until the STSPIN32F0A bootloader is ready to start the detection phase with a specific interface communication. This time corresponds to bootloader startup timing (minimum bootloader startup equal 1.612 ms), during which resources used by the bootloader are initialized.

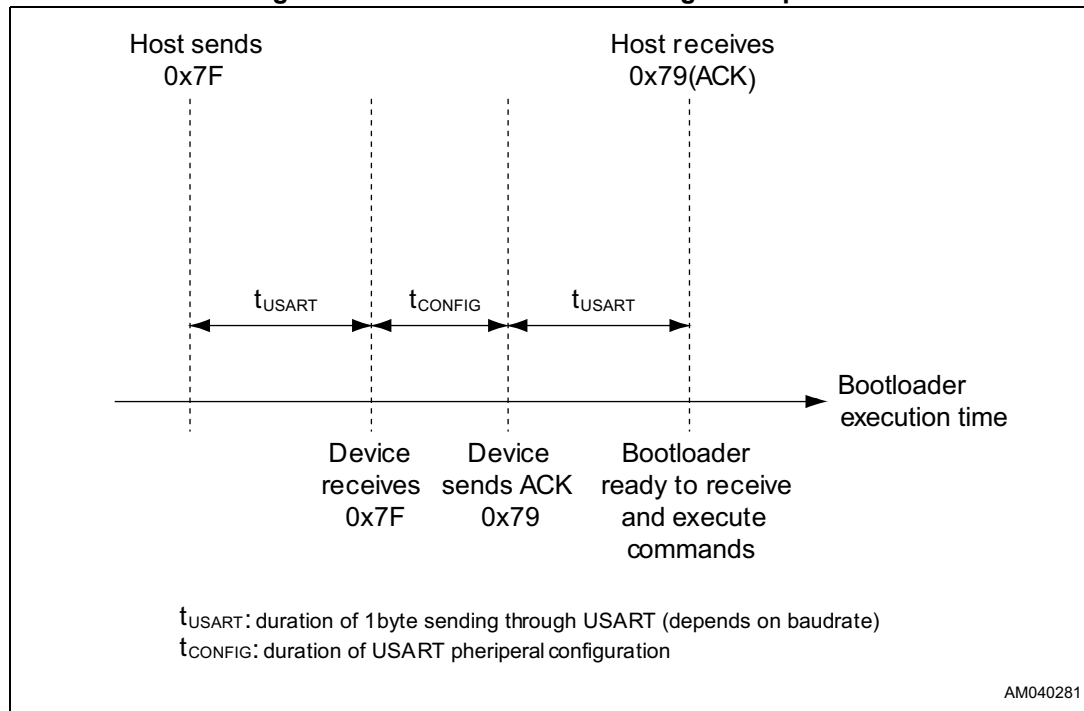
Figure 3. Bootloader startup timing description



5.2 USART connection timing

USART connection timing is the time that the host should wait for between sending the synchronization data (0x7F) and receiving the first acknowledge response (0x79).

Figure 4. USART connection timing description



1. Receiving any other character different from 0x7F (or line glitches) will cause the bootloader to start communication using a wrong baud rate. The bootloader measures the signal length between the rising edge of first 1 bit in 0x7F to the falling edge of the last 1 bit in 0x7F to deduce the baud rate value.
2. The bootloader does not re-align the calculated baud rate to standard baud rate values (i.e. 1200, 9600, 115200, etc.).

Table 4. USART bootloader minimum timings

One USART byte sending t_{USART_min} [ms]	USART configuration t_{CONFIG_min} [ms]	USART connection [ms]
0.078125	0.0064	0.16265

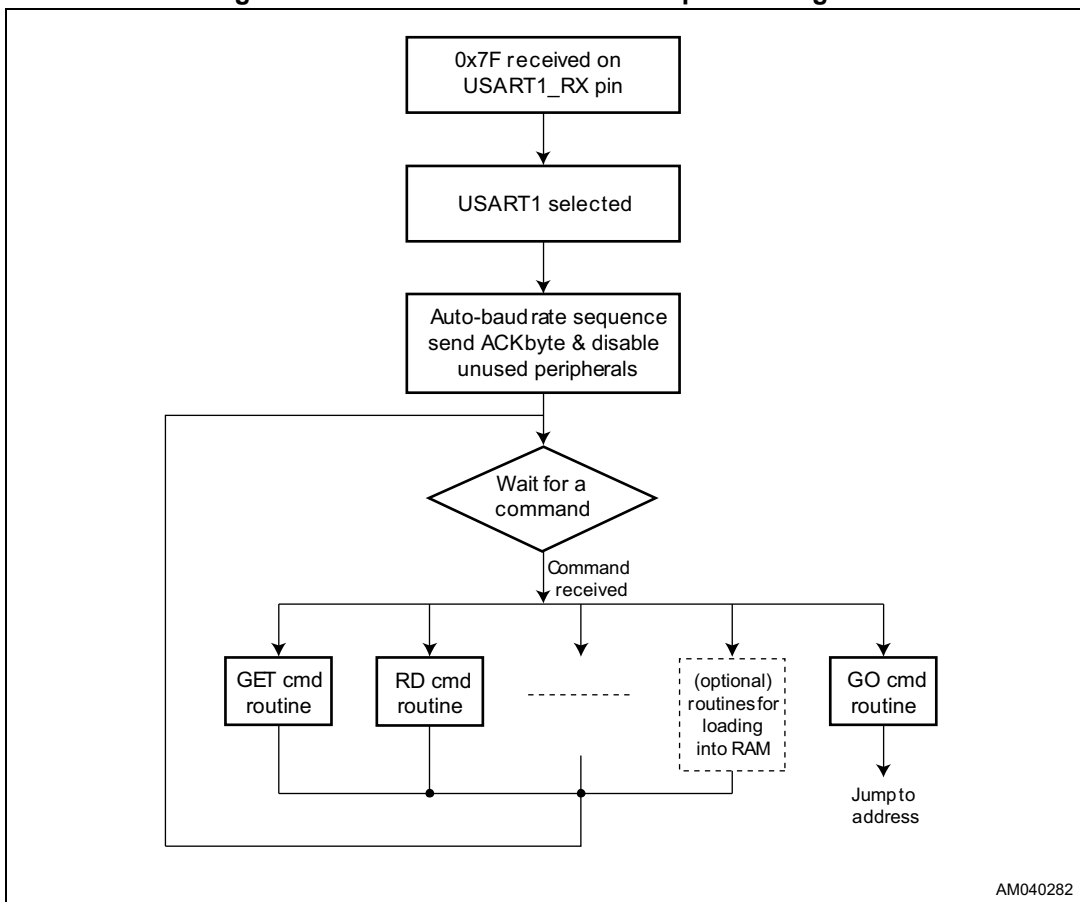
6 USART bootloader code sequence

Once the system memory boot mode is entered and the STSPIN32F0A has been configured, the bootloader code begins to scan the USART1_RX line pin, waiting to receive the 0x7F data frame: one start bit, 0x7F data bits, even the parity bit and one stop bit.

The duration of this data frame is measured using the timer. The count value of the timer is then used to calculate the corresponding baud rate factor with respect to the current system clock.

Next, the code initializes the serial interface accordingly. Using this calculated baud rate, an acknowledge byte (0x79) is returned to the host, which signals that the STSPIN32F0A is ready to receive commands.

Figure 5. USART bootloader code sequence diagram



AM040282

6.1 Choosing the baud rate

The calculation of the serial baud rate for the USART1, from the length of the first byte that is received, is used to operate the bootloader within a wide range of baud rates. However, the upper and lower limits have to be kept, in order to ensure the proper data transfer.

For a correct data transfer from the host to the microcontroller, the maximum deviation between the internal initialized baud rate for the USART1 and the real baud rate of the host should be below 2.5%. The deviation (f_B , in percent) between the host baud rate and the microcontroller baud rate can be calculated using the following formula

Equation 1

$$f_B = \left| \frac{\text{MCU baud rate} - \text{Host baud rate}}{\text{MCU baud rate}} \right| \times 100 \% , \text{ where } f_B \leq 2.5\%$$

This baud rate deviation is a nonlinear function depending on the CPU clock and the baud rate of the host. The maximum of the function (f_B) increases with the host baud rate. This is due to the smaller baud rate prescale factors, and the implied higher quantization error.

The lowest tested baud rate (B_{LOW}) is 1200. Baud rates below B_{LOW} would cause the timer to overflow. In this event, the USART1 would not be correctly initialized.

B_{HIGH} is the highest baud rate for which the deviation still does not exceed the limit. All baud rates between B_{LOW} and B_{HIGH} are below the deviation limit. The highest tested baud rate (B_{HIGH}) is 115200.

6.2 USART bootloader command set

The supported commands are listed in [Table 5](#). For the detailed description of each command, refer to the AN3155 USART protocol used in the STM32 bootloader.

Refer to the STM32 product datasheets and to the “STM32 microcontroller system memory boot mode” application note (AN2606) to know which memory areas are valid for these commands.

Table 5. USART bootloader commands

Command ⁽¹⁾	Code	Description
Get ⁽²⁾	0x00	Gets the version and the allowed commands supported by the current version of the bootloader
Get Version and Read Protection Status ⁽²⁾	0x01	Gets the bootloader version and the Read Protection status of the Flash memory
Get ID ⁽²⁾	0x02	Gets the MCU ID
Read Memory ⁽³⁾	0x11	Reads up to 256 bytes of the memory starting from an address specified by the application
Go ⁽³⁾	0x21	Jumps to the user application code located in the internal Flash memory or in the SRAM
Write Memory ⁽³⁾	0x31	Writes up to 256 bytes to the RAM or Flash memory starting from an address specified by the application
Erase ^{(3), (4)}	0x43	Erases from one to all the Flash memory pages
Extended Erase ^{(3), (4)}	0x44	Erases from one to all the Flash memory pages using the two byte addressing mode
Write Protect	0x63	Enables the write protection for some sectors
Write Unprotect	0x73	Disables the write protection for all Flash memory sectors
Readout Protect	0x82	Enables the read protection
Readout Unprotect ⁽²⁾	0x92	Disables the read protection

1. If a denied command is received or an error occurs during the command execution, the bootloader sends the NACK byte and goes back to command checking.
2. Read protection - when the RDP (read protection) option is active, only this limited subset of commands is available. All other commands are NACKed and have no effect on the device. Once the RDP has been removed, the other commands become active.
3. Refer to the STM32 product datasheets and to the “STM32 microcontroller system memory boot mode” application note (AN2606) to know which memory areas are valid for these commands.
4. Erase (x043) and Extended Erase (0x44) are exclusive. A device may support either the Erase command or the Extended Erase command but not both.

Communication safety

All communications from the programming tool (PC) to the device are verified by

1. Checksum: received blocks of data bytes are XORed. A byte containing the computed XOR of all previous bytes is added to the end of each communication (checksum byte). By XORing all received bytes, data + checksum, the result at the end of the packet must be 0x00
2. For each command the host sends a byte and its complement (XOR = 0x00)
3. UART: parity check active (even parity)

Each packet is either accepted (ACK answer) or discarded (NACK answer)

- ACK = 0x79
- NACK = 0x1F

7 Revision history

Table 6. Document revision history

Date	Revision	Changes
21-Dec-2017	1	Initial release.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST's terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers' products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2017 STMicroelectronics – All rights reserved

