Introduction

The purpose of this document is to describe the STMicroelectronics STM32™ and STM8™ Flash loader demonstrator application that was developed to illustrate the System memory boot loader capabilities.

This document details the prerequested hardware and software environments, as well as the use cases of the demonstrator software.
Contents

1 Getting started ................................................................. 5
  1.1 Package contents ....................................................... 5
    1.1.1 Software contents ............................................. 5
    1.1.2 Hardware contents ............................................ 5
  1.2 System requirements .................................................. 5
  1.3 Flash loader demonstrator installation ......................... 7
    1.3.1 Software installation ....................................... 7
    1.3.2 Hardware installation ..................................... 9

2 User interface description ............................................. 10

3 Command-line usage .................................................... 20

4 Revision history .......................................................... 24
List of tables

Table 1. Document revision history ........................................... 24
List of figures

Figure 1. System Properties dialog box ............................................. 6
Figure 2. Device Manager window ................................................ 7
Figure 3. InstallShield Wizard ....................................................... 8
Figure 4. InstallShield Wizard license agreement ................................. 8
Figure 5. Connection settings page ................................................. 11
Figure 6. Flash status page ........................................................... 12
Figure 7. Device information page - STM32 example .......................... 13
Figure 8. Device information page - STM8 example ........................... 14
Figure 9. Operation choice page for STM32 ...................................... 16
Figure 10. Operation choice page for STM8 ...................................... 17
Figure 11. Option byte edition page ............................................... 18
Figure 12. Operation progress page ............................................... 19
Figure 13. Command-line version ............................................... 20
1 Getting started

1.1 Package contents

The following items are supplied in the Flash loader demonstrator package:

1.1.1 Software contents

1. **STBLLIB.dll**: a dynamic-link library implementing the system memory boot loader protocol and the communication APIs as virtual functions to be loaded dynamically from the **STUARTBLLib.dll** file.
2. **STUARTBLLib.dll**: a dynamic-link library implementing the system memory boot loader protocol and the RS232 COM communication APIs.
3. **Files.dll**: a dynamic-link library implementing the needed file manipulation APIs to load and store binary, hexadecimal and motorola S19 files.
4. STMicroelectronics Flash loader.exe: a wizard application that provides the high-level operations that can be performed by the user.
5. **STMFlashLoader.exe**: a command-line version of the STMicroelectronics Flash loader.exe that provides the same features over several options.
6. The “Map” directory is located in the installation directory. It contains the mapping description files of the supported devices.
7. The “Src” directory is located in the installation directory. It contains the header and Lib files of the two DLLs and the complete source of the command-line version.
8. The “Doc” directory is located in the installation directory, it contains the UM0462 and UM0516 ([Windows API for STMicroelectronics microcontroller boot loaders](http://www.st.com)) user manuals.

1.1.2 Hardware contents

The Flash loader demonstrator is designed to work with all STMicroelectronics devices that support the system memory boot mode UART protocol. For more details, please visit the STMicroelectronics website ([http://www.st.com](http://www.st.com)).

1.2 System requirements

In order to use the Flash loader demonstrator with the Windows operating system, a recent version of Windows, such as Windows 98, Millennium, 2000, XP, Vista or Windows7 must be installed on the PC.

The version of the Windows OS installed on your computer may be determined by right-clicking on the “My Computer” icon on the desktop, then clicking on the “Properties” item in the displayed pop-up menu. The OS type is displayed in the “System Properties” dialog box under the “System” label as shown in *Figure 1*. 
For communication purposes, you need to verify that you have an available COM port (RS232) if the application implements the UART interface.

To check that you have an available interface (COM), right-click on the "My Computer" icon on the desktop and select "Properties" from the pop-up menu. The “System Properties” dialog box appears. Click on the “Hardware” tab, and then on the “Device manager” button to display the system hardware configuration. Available COM ports are grouped under the “Ports (COM & LPT)” node in the hardware tree as shown in Figure 2.
It is interesting to know the capabilities of the COM port. To find out, right click on the Communication Port (COMx) item then click on “Properties” to display the Properties window. Select the “Port Settings” tab, then click on the arrow next to the “Bits per second” combo box to know the baud rates supported by the port.

1.3 Flash loader demonstrator installation

1.3.1 Software installation

If an older version is installed on you computer, remove it by using the “Add or Remove Programs” service in the “Control Panel”.

Run the provided Setup.exe file: the InstallShield Wizard will guide you through the installation of the Flash loader demonstrator application on your computer as shown in Figure 3 and Figure 4 (You must accept the license agreement to install the software).
Once the software has been successfully installed, click on the “Finish” button. The version.txt file containing the new release notes will automatically open in the Microsoft®-native Notepad application. Closing Notepad will launch the Flash loader demonstrator if the check boxes are kept as default in the InstallShield Wizard.
1.3.2 Hardware installation

As the Flash loader demonstrator is able to communicate over the UART interface, the device should be connected to a spare PC COM port in the event of a UART communication.
2 User interface description

The Flash loader demonstrator is designed as a wizard application. It is structured into six steps, the:

1. Connection settings page
2. Flash status page
3. Device information page
4. Operation choice page
5. Option byte edition page
6. Operation progress page

Step 1

Run the Flash loader demonstrator application from the “Programs” menu (connection to the device has not been made yet) then, make sure that the device is connected to your PC and reset it to restart the system memory boot loader code.

This step consists in selecting the UART connection interface and its related settings. Set the connection settings (port name, baud rate and timeout, etc.) as shown in Figure 5. For an optimum configuration for the UART interface, set “Baud Rate” to 115200 bits per second and “Timeout(s)” to 5 seconds.

Ensure that the boot configuration pins are set correctly, then click “Next” to continue. If a connection has been established, the wizard moves to the next step, otherwise a message box is displayed that indicates the error that occurred.

Possible errors messages:

● “Cannot open the COM port”: this message is shown if the selected COM port is not found or if it is already being used by another process.
● “Unrecognized device”: this message is shown if the received value is different from 0x79. Resetting the device may solve the problem.
● “No response from the target”: this message is shown when there is no response from the target. It indicates that the System memory boot loader is not functional. Verify the boot configuration and check that the used microcontroller contains the boot loader code.

Note: The Timeout argument is the period of time after which a read request from the serial port is aborted if no data is received. The recommended value is 5 seconds, but it depends on the used environment, like the hardware performance.
Figure 5. Connection settings page

Note: “Echo” combo-box is present in version 2.1.0 of the Flash loader demonstrator for the support of some STM8 devices that use LIN echo back emulation through the UART protocol. If these devices are not used, this option should be kept disabled.
Step 2

In the second step the connection has been established and communication has started. It consists in displaying the Flash memory status. This status can be read-protected, in which case the “Next” button is disabled until the read protection is removed by clicking on the “Remove protection” button.

**Note:** Clicking on the “Remove protection” button will not only read-unprotect the Flash memory, it will also erase all its pages.

**Figure 6. Flash status page**
**Step 3**

In this step the Wizard displays the available device information such as the target ID, the firmware version, the supported device, the memory map and the memory protection status.

Select the target name in the target combobox as shown in *Figure 7* and *Figure 8*, then click on “Next” to continue.

---

**Figure 7. Device information page - STM32 example**
Figure 8. Device information page - STM8 example
Step 4

At this step, select the requested operation—Erase, Download, Upload or Disable/Enable Flash protection or Edit option bytes—and set the related parameters:

1. **Erase**
   a) Choose “All” to erase the whole memory
   b) Choose “Selection” to customize the Erase operation. Click the “...” button to display the memory mapping dialog window. Then check the pages to be erased and click “Ok”.

2. **Download**
   - Click the related browse button to open a binary, hexadecimal or S19 Motorola file.
     - If the loaded file is a binary file, the download address is the start address of the first page and the “@” field is still editable to accept changes.
     - If the loaded file is a hexadecimal or an S19 motorola file, the download address is the start address of the first record in the file, and the “@” field is read-only.
   - Check the “Verify” check box to launch the verification process when the Download operation is finished.
   - Check “Jump to the user program” to launch the downloaded program.
   - Check “Optimize” to filter FFs packets (256 bytes).
   - Check “Apply option bytes”, then browse the option byte file created by the “Edit option bytes” operation. The values in the selected file will be applied to the device after download.

3. **Upload**
   - Click the related browse button to select which binary, hexadecimal or S19 motorola file will store the uploaded data.

4. **Disable/Enable Flash protection**
   - Select the choices from the two drop-down menus to make up the desired command (Enable Read protection, Disable Read protection, Enable Write protection, Disable Write protection). All protection commands will be applied to all the Flash memory pages except for the Enable Write Protection, which can be customized. This is done by clicking the “...” button to select the pages to be write-protected.

5. **Edit option bytes**
   - If you need to set the option bytes, check the option then click “Next” to move to the option byte edition page (Step 5 Figure 11).

---

**Warning:** Erase and Download operations can be performed only if the write protection is disabled.
Figure 9. Operation choice page for STM32
Step 5

Note: This step applies to STM32 devices only. There is no step 5 for STM8 devices.

The last Wizard page depends on the operation selected in Step 4.

a) Case of an “Edit option bytes” operation:

The Option byte edition page is displayed. It contains the current option byte values loaded from the device: RDP, USER, Data0, Data1, WRP0, WRP1, WRP2 and WRP3. For more details, please refer to the Option byte loader section in the “STM32F10xxx Flash programming manual” (PM0042 available from www.st.com).

This step gives the possibility of applying the edited option byte values, loading them from the device and saving them to a file.
b) Case of any other operation:

The operation page is shown. It gives the size of the data to be downloaded or uploaded, the percent completed and the duration of the operation as illustrated in Figure 12.

- If the operation is successful, the progress bar is green colored. If an error occurs, the bar turns red and the error is displayed.
- To stop the operation click the “Cancel” button.
- If the “Jump to the user program” check box was checked in the previous step (Step 4), and the user program was successfully downloaded, communication with the system memory boot loader is lost. Consequently, the “Back” button is redirected to the “Connection settings page” (Step 1) to avoid the launch of a new operation.

If the “Jump to the user program” check box was not checked in Step 4, the “Back” button is still active and you can return to Step 4 and select a new operation.
Figure 12. Operation progress page

- Target: STM32_Connectivityline_256K
- Map file: STM32_Connectivityline_256K_STmap
- Operation: DOWNLOAD
- File name: C:\MyFile.v19
- File size: 256.00 KB (362144 bytes)
- Status: 209.88 KB (214922 bytes) of 256.00 KB (362144 bytes)
- Time: 00:00:31

Downloading data 82%
3 Command-line usage

The command-line version (STMFlashLoader.exe) provides the same functionality as the GUI. It supports several options in order to run a sequence of operations.

The following paragraphs describe the available command-line options as shown in Figure 13.

**Figure 13. Command-line version**

```
STMFlashLoader.exe [options] [Arguments] [options] [Arguments]...
-?    (Show this help)
-c:   (Establish connection to the COM port)
-pn   : e.g. 1, 2, ... default 1
-hr   : 115200, 57600, ... default 57600
-db   : value in (0, 6, 7, 8) default 6
-pr   : value in (NONE, ODD, EVEN), default EVEN
-sb   : value in (1, 2, 3, 4), default 1
-ec   : value ON or OFF, default is OFF
-to   : <uc> e.g. 1000, 2000, 3000, ... default 5000
-device_name: e.g. STM32_low-density.bin, [See the Map directory]
-e:    (erase flash pages)
--all all pages: erase all pages
--page page group pages group codes: erase specified group pages
--u file_name: full path name of the file
--d file_name: full path name of the file
--a address(hex): start @ in hex; ignored if it is not a binary file
--f file_name: full path name (.bin, .hex or .s19 file)
--v: verify after download
--n: output: overwrite FFs data
--r address(hex): address in hexadecimal
--w address(hex): address in hexadecimal
--d: disable read protection
--w: activate or deactivate protections
--e: enable write protection for sector codes (e.g 1,2, etc.)
--p: enable write protection
--dp: disable read protection
--wp: enable write protection, all arguments following this one will fail
--o file_name: get option bytes from the device and write it in the specified file
--set file_name: load option bytes from the specified file and write it to the device
--set-wals --OPR hex_val: set the specified option bytes: OPR in: User, HWP, Data1, Data2, WRP1, WRP2, WRP3
-Rtu  (Get RTU line to Hi, Lo)
-W   (Set W to Hi, Lo)
The -c option allows you to select the COM port that the command uses to communicate with the target MCU. By default, the command uses COM1. To select different COM port and connection settings, use the -c option in the form:

- \(-c \--pn\) port_number (e.g. 1, 2,..., default 1)
- \(-c \--br\) baud_rate (e.g. 115200, 57600..., default 57600)
- \(-c \--db\) data_bits (value in \{5,6,7,8\}..., default 8)
- \(-c \--pr\) parity (value in \{NONE,ODD,EVEN\}..., default EVEN)
- \(-c \--sb\) stop_bits (value in \{1,1.5,2\}..., default 1)
- \(-c \--ec\) echo (value ON or OFF..., default is OFF)
- \(-c \--to\) time_out ((ms) e.g 1000, 2000, 3000..., default 5000)

The -c option supports multiple arguments. This means that you can set more than one argument in the same command:

- -c --pn 1 --br 115200 --to 7000

-\i device_name\n
Defines the MCU target to be used.

For example: STM8_32K, STM32_Med-density_128K, STM32_High-density_512K, STM32_Low-density_16K etc. The device name is the name of the map file located in the Map directory.

-\e\n
Erase command.

According to the given arguments, the command can be used to erase a specific page of memory or, to erase the entire Flash memory. This operation can take a second or more to complete, depending on the memory size involved.

- \(-e \--all\) erase all pages
- \(-e \--sec\) number_of_pages group pages_group_codes
  - \(-e \--sec\) 3 0 1 2 erases 3 groups of pages coded 0, 1 and 2

-\u\n
Uploads the Flash memory contents to the specified file (bin, hex or s19 file; the file type is recognized by its extension), to specify a file use the -u option in the form:

- \(-u \--fn\) file_name (full path name)

-\d\n
Downloads the contents of the specified file into the MCU Flash memory at the specified address. To specify the file to be downloaded and the download address, use the -d option in the form:

- \(-d \--a\) address(hex) \--fn file_name (full path name (bin, hex or s19 file); the file type is recognized by its extension).

The address is mandatory in the case of binary files and ignored in the case of hex and s19 files.

To verify the downloaded data, add the --v argument.

To optimize and remove FF packets, use the --o argument.
-o
Gets or sets option bytes.
Use --get to read option bytes from the device and store values to a file.
   --get --fn file_name (full path name)
Use --set to write option bytes to the device. The option bytes can be read from a file or
given as values.
   --set --fn file_name (full path name)
   --set --vals --OPB hex_value (OPB in (User, RDP, Data0, Data1, WRP0, WRP1, WRP2, WRP3).
The -o option can accept multiple arguments as shown below:
   -o --get get_file_name --set set_file_name
   -o --get get_file_name --set --vals --User 01 --RDP 5A --Data0 DE --Data1 EA

---
Warning: When setting option bytes, if RDP is not equal to A5h, the
read protection is activated, and all subsequent operations
will fail.
---
-p
Activates or deactivates the protection. It is used as shown below:

- p --erp (Activate read protection)
- p --drp (Deactivate read protection)
- p --ewp number_of_pages_group pages_group_codes (activates the write protection on the given page group codes)
- p --dwp (disables the write protection)

Warning: The --erp argument activates the read protection. All subsequent operations will fail. To avoid this kind of problem, use -p --erp as the last argument.

-r
Performs a jump to the specified address. It is used as follows:
- r --a address(hex)

-Rts
Sets the COM RTS pin to either high or low level. It is used as follows:
- Rts -- Hi

-Dtr
Sets the COM DTR pin to either high or low level. It is used as follows:
- Dtr -- Lo

Note: Note that all command-line options found are executed IN ORDER. Thus, with a careful arrangement of the command-line options, you can perform a complicated sequence of operations using custom batch files.
4 Revision history

Table 1. Document revision history

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>05-Jun-2008</td>
<td>2</td>
<td>Flash loader demonstrator version upgraded to V1.1. Small text changes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Section 1.1.1: Software contents updated. Section 1.2: System requirements modified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Welcome step removed, Flash status page and Option byte edition page added.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Section 3: Command-line usage added. Step 2 on page 12 added. Step 5 on page 17 modified.</td>
</tr>
<tr>
<td>17-Jun-2008</td>
<td>3</td>
<td>Software revision updated in Section 1.3.1: Software installation on page 7.</td>
</tr>
<tr>
<td>31-Oct-2008</td>
<td>4</td>
<td>Flash loader demonstrator version upgraded to V1.2. It can also be used with STM8 devices. Figure 5 to Figure 12 updated accordingly.</td>
</tr>
<tr>
<td>04-Mar-2009</td>
<td>5</td>
<td>Flash loader demonstrator version upgraded to V1.3. Figure 5 to Figure 12 updated accordingly.</td>
</tr>
<tr>
<td>02-Jul-2009</td>
<td>6</td>
<td>Flash loader demonstrator version upgraded to V2.0. This version applies not only to the STM32™ family but also to the STM8™ family. Section 1.1.1: Software contents updated (STUARTBLLib.dll and STCANBLLib.dll added). Section 1.2: System requirements modified. Figure 2: Device Manager window changed. Section 1.3.1: Software installation and Section 1.3.2: Hardware installation updated. Step 1 and Figure 5: Connection settings page updated. Small text changes.</td>
</tr>
<tr>
<td>12-Nov-2009</td>
<td>7</td>
<td>Flash loader demonstrator version upgraded to V2.1.0. Figure 1 to Figure 13 updated accordingly. Flash loader demonstrator is upgraded to support only UART protocol for STM8™ and STM32™ families.</td>
</tr>
</tbody>
</table>