



Introduction

The ST Handy USB Memory Single Programmer kit is a small, light, low-power, high-performance memory programming tool. It is a joint development of STMicroelectronics and Leap. It is used to program efficiently an extended range of memories, including EPROMs, parallel and serial EEPROMs, and parallel and serial Flash memories. It supports devices operating from 1.8V, 2.0V, 2.5V, 3.0V, 3.3V and 5.0V supply voltages and packaged in DIP (up to 48 pins), PLCC32, SOP16, TSOP48 and TSSOP8.

It features an USB interface to connect to a PC or notebook.

The ST Handy USB Memory single Programmer kit is delivered with an easy-to-use operating software that:

- Quickly and easily selects the device Part Numbers by cross selection.
- Automatically detects a Flash memory IC brand and Part Number.
- Performs programmer self-test by checking the hardware status.

This User Manual is the reference document for installing, configuring and using the Handy USB Memory single Programmer kit. Before using the programming equipment, please read carefully the recommendations contained in this document.

The ST Handy USB Memory single Programmer kit will be referred to as ST Memory Programmer kit throughout the document.

Contents

1	ST Memory Programmer Kit Features	5
1.1	Description of the kit components	5
1.2	Description of the front panel	6
1.3	System requirements	6
1.4	ST Memory Programmer technical characteristics	7
2	Getting started	8
2.1	Installing the programming software	8
2.2	Installing the hardware	8
2.3	Installing the USB software driver	9
2.3.1	Installing the USB driver on Windows 98 and ME	9
2.3.2	Installing the USB driver on Windows XP	10
2.3.3	Installing the USB driver on Windows 2000	12
2.3.4	Troubleshooting	13
2.4	Recommendations and potential problems	14
2.5	Installing and using socket adapters	14
3	How to use the programming software	15
3.1	Description of the Main Dialog Box	15
3.2	Source section	16
3.2.1	Device read-only box	16
3.2.2	DeviceSum read-only box	16
3.2.3	Type submenu	17
3.2.4	Load submenu	18
3.2.5	Read and Confirm commands	19
3.2.6	Edit submenu	19
3.2.7	Save submenu	23
3.3	Process section	24
3.4	Parameters menus	25
3.4.1	Common Setup submenu	25
3.4.2	Extra Parameter submenu	26
3.4.3	Device Option submenu	27
3.5	Options menus	28

3.5.1	System Option submenu	28
3.5.2	Self test submenu	29
3.6	Project menu	30
3.7	About command	31
3.8	Exit command	31
Appendix A	Hotkeys	32
Appendix B	Error Messages	33
B.1	Missing Device Error	33
B.2	Device Insertion Error	33
B.3	Device Type Error	34
4	Revision history	35

List of figures

Figure 1.	Description of the ST Memory Programmer Kit	5
Figure 2.	ST Memory Programmer Front Panel.	6
Figure 3.	Device Manager Window	10
Figure 4.	Hardware Update wizard: Selecting Advanced Installation (Step a).	11
Figure 5.	Hardware Update wizard: Installing the driver from the CD-ROM (Step b)	11
Figure 6.	Found New Hardware wizard	12
Figure 7.	Hardware Update wizard: Installing the driver (Step a)	12
Figure 8.	Hardware Update wizard: Installing the driver (Step b)	13
Figure 9.	Error Message: ST Memory Programmer Not Connected	13
Figure 10.	Installing Socket Adapters	14
Figure 11.	Main Dialog Box	15
Figure 12.	Source section	16
Figure 13.	Type submenu	17
Figure 14.	Load submenu	18
Figure 15.	Load Pop-up Windows	18
Figure 16.	Edit Submenu/8 Bits Window	21
Figure 17.	Get IC Checksum Submenu	22
Figure 18.	Block Process Submenu	22
Figure 19.	Destination Range Error Message	22
Figure 20.	Fill Data Submenu	22
Figure 21.	Buffer Information Tab	23
Figure 22.	Save submenu	23
Figure 23.	Process section	24
Figure 24.	Common Setup Menu	26
Figure 25.	Extra Parameters submenu	26
Figure 26.	Device Option submenu	27
Figure 27.	System Option submenu	28
Figure 28.	Self Test submenu	29
Figure 29.	Project menu.	30
Figure 30.	Save Project submenu	31
Figure 31.	About Window.	31
Figure 32.	Exit window.	31
Figure 33.	Hotkey Menu.	32
Figure 34.	Error message: Missing Device	33
Figure 35.	Error message: Device Insertion Problem	33
Figure 36.	Error message: Device Type To Be Checked	34

1 ST Memory Programmer Kit Features

This section gives a general description of the ST Memory Programmer kit.

1.1 Description of the kit components

The ST Handy USB Memory Single Programmer kit is composed of the ST Memory Programmer plus a full set of accessories (see [Figure 1](#)):

- a 12V/2A power switching converter
- a USB cable
- a CD-ROM containing the programming software, the USB driver software and the User Manual
- Standard Adapters for TSOP48, PLCC32, TSSOP8 and SOP16 (150mils) packages

Please check that the programmer kit includes all the components listed above. If something is missing, please contact:

LEAP ELECTRONIC LTD.,CO.

6th Fl-4, No. 4, Lane 609, Chunghsin Rd., Sec. 5
Sanchung, Taipei Hsien, Taiwan, ROC
TEL: 886-2-29991860
FAX: 886-2-29998815
URL: <http://www.leap.com.tw>
E-mail: market@leap.com.tw

Figure 1. Description of the ST Memory Programmer Kit



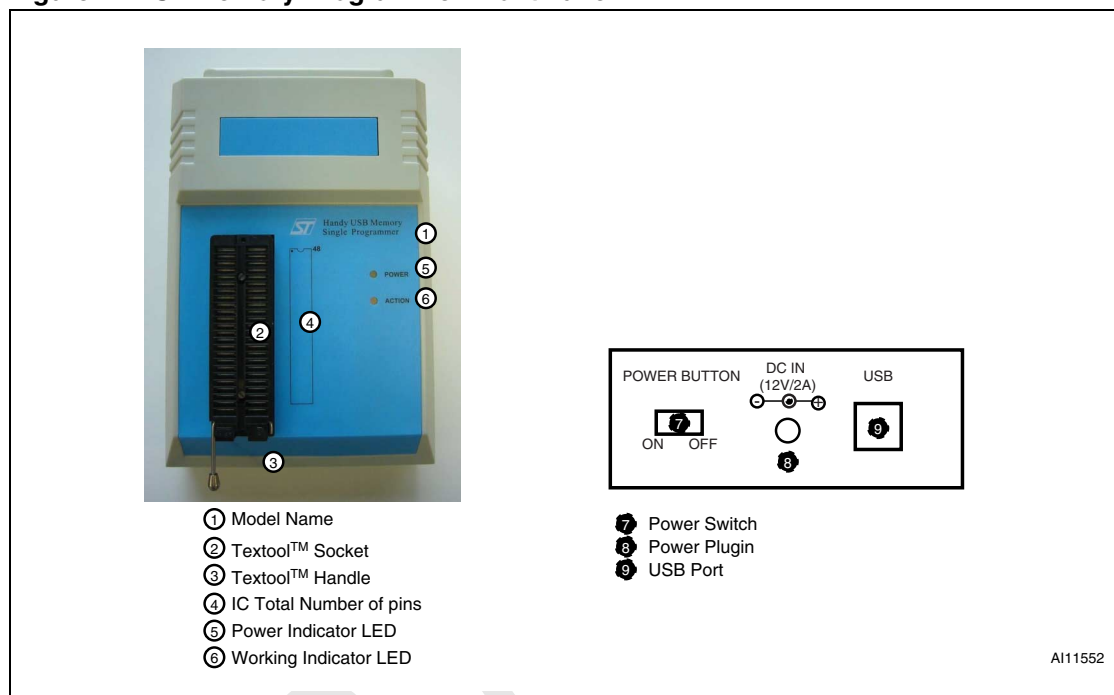
1.2 Description of the front panel

The ST Memory Programmer features a 48-pin DIP Textool™ socket to program devices in DIP packages with up to 48 pins, without the need for adapters, and PLCC32, SOP16 (150mils), TSOP48 and TSSOP8 packages using additional adapters.

A pin number selector sets the device total number of pins.

Note: Other adapters are available on request.

Figure 2. ST Memory Programmer Front Panel



1.3 System requirements

- Host Computer:
 - PC
 - Operating System: Windows 98SE/ME/2000(SP4)/XP(SP2)
 - Processor: Pentium® Pro or above
 - Memory: 128MBytes RAM or more
 - Hard Disk: 60 MBytes or more
 - Communication Interface: USB version 1.1
- Data File Format:
 - Binary
 - Hexadecimal
 - Autodetection of the data format

1.4 ST Memory Programmer technical characteristics

- ST Memory Programmer Physical Dimension
 - Dimensions: 16cm × 11cm × 4.5cm
 - Weight: 0.5kg
 - Operating Temperature: 5 to 45°C
 - Maximum Operating Humidity: 90% (non-condensing)
- ST Memory Programmer Electrical Requirements
 - Wide Voltage Range: 90 to 260V (AC), auto-switching
 - Frequency: 47 to 63Hz
 - Power consumption: 24 Watt maximum
- Device Socket:
 - 48-pin DIP ZIF (Zero Insertion Force) socket
- Standard Adapters (provided by default):
 - PLCC32, SOP16 (150mils), TSOP48, TSSOP8
- Optional Adaptors (provided upon request):
 - PLCC, SOP, TSOP, SSOP, TSSOP, VSOP, PSOP, WSOP, QFP, SDIP, BGA, TBGA, TFBGA
- Devices Supported:
 - EPROMs, EEPROMs, Flash memories, serial EEPROMs, serial Flash memories
 - AC/DC Characteristics
 - Input/Output Voltage: 3.3V and 5.0V
 - Supply Voltage (V_{CC}): 1.8V, 2.0V, 2.5V, 3.0V, 3.3V and 5.0V
 - Operating Current: 500mA (max.)
 - Program Voltage (V_{PP}): 1.0 to 20.0V (max.)
 - Program Current: 500mA (max.)
 - Clock Frequency: 32.0MHz (max.)

2 Getting started

To install the ST Memory Programmer:

1. Install the programming software
2. Install the ST Memory Programmer hardware:
 - Connect the ST Memory Programmer to the Host Computer USB interface.
 - Switch on the ST Memory Programmer.
3. Install the USB software driver.

The programming software must be installed prior to connecting the ST Memory Programmer hardware to the Host Computer and switching it on. The USB driver can be installed prior or after the programming software.

After completing the hardware and software installation, the ST Memory Programmer is ready to be used.

Socket adapters may be necessary. Refer to [Section 2.5](#) for how to install them.

2.1 Installing the programming software

To install the programming software:

1. Insert the CD-ROM provided with your ST Memory Programmer kit.
2. If your computer supports the *AutoRun* wizard, the programming software installation starts when you insert the CD-ROM. Follow the wizard recommendations to complete the installation.
3. If your computer does not support *AutoRun*, run the installation program, *Setup.EXE*, located in the CD-ROM root.

2.2 Installing the hardware

To install the hardware of the ST Memory Programmer:

1. Take the ST Memory Programmer out of the box.
2. Check if the power switch is turned off. If not, turn it off.
3. Use the USB cable to connect the ST Memory Programmer to the Host Computer. Check that the USB cable is correctly connected.
4. Connect the power switching converter to the ST Memory Programmer and plug in the power switching converter without turning on the power. Check that the power cable is correctly connected.
5. Turn on the power. The green LED on the power switching converter and the power LED on the ST Memory Programmer should light up.

2.3 Installing the USB software driver

Once the ST Memory Programmer is connected to the Host Computer, and the power is turned on, the USB driver is ready to be installed.

The driver is located:

- On the CD-ROM: in the subdirectory *CD-R:\Driver*
- On the Host Computer hard disk, if the programming software has already been installed: in the subdirectory *C:\Program Files\STM Programmer \Driver\USBWrite.inf* subdirectory

The USB driver installation depends on the Host Computer Operating System.

- For Windows 98 and Windows ME, refer to [Section 2.3.1](#).
- For Windows 2000 and Windows XP, refer to [Section 2.3.2](#).

The USB driver must be installed only once, when first connecting the programmer to the Host Computer.

2.3.1 Installing the USB driver on Windows 98 and ME

When the Host Computer power is turned on, a message is displayed on the screen to indicate that it has detected a new USB Plug-and-Play device and is installing it automatically. A message is displayed to indicate that the USB driver installation is complete.

If no message is displayed, the driver must be installed manually from the CD-ROM by following the below procedure:

1. In the Control Panel, open the *Add New Hardware* wizard.
2. Select the *Add New Hardware* command in the wizard. The wizard shows a list of drivers:
 - a) If the appropriate driver is shown in the list, select it and install it.
 - b) If the driver is not shown in the list: install it from the Host Computer or from the CD-ROM.
3. If the wizard cannot find the driver at this location, go to the Control Panel, open the *System menu* and select *Device Manager* to list all the hardware devices installed on the computer. Search for 'Handy USB Memory Single Programmer'.
 - a) If there is a question mark or exclamation mark in front of 'Handy USB Memory Single Programmer', the computer cannot find the driver.
 - Double click the driver name.
 - Select the *Reinstall Driver* option in the *General* menu and install the driver from the CD-ROM.
 - b) If there is no mark in front of 'Handy USB Memory Single Programmer', it means that the programmer has been successfully installed and that you can start using it.

2.3.2 Installing the USB driver on Windows XP

To install the USB driver on Windows XP:

1. In the Control Panel, right click on *My Computer* icon and select *Manage* to open the *Computer Management* window.
2. Select *Device Manager*. The list of hardware devices connected to the host computer is displayed. A question mark indicates that a new USB plug-and-play hardware (*USB IC Writer*) has been detected and the corresponding driver must be installed (see [Figure 3](#)).
3. Right click on *USB IC Writer* and select the *Hardware Update* wizard to install the USB driver from the CD-ROM (see [Figure 4](#)):
 - a) Select the Advanced Installation.
 - b) Insert the CD-ROM. In the *CD-R:\Driver* subdirectory, open the *USBWriter.SYS* file (see [Figure 5](#)) and install the driver.

Note: 1 The ST Memory Programmer is compatible with all Windows XP versions. However, it is preferable to use Service Pack 2 or above.

- 2 Using the Add New Hardware wizard and the Auto search is not recommended to install the USB driver.

Figure 3. Device Manager Window

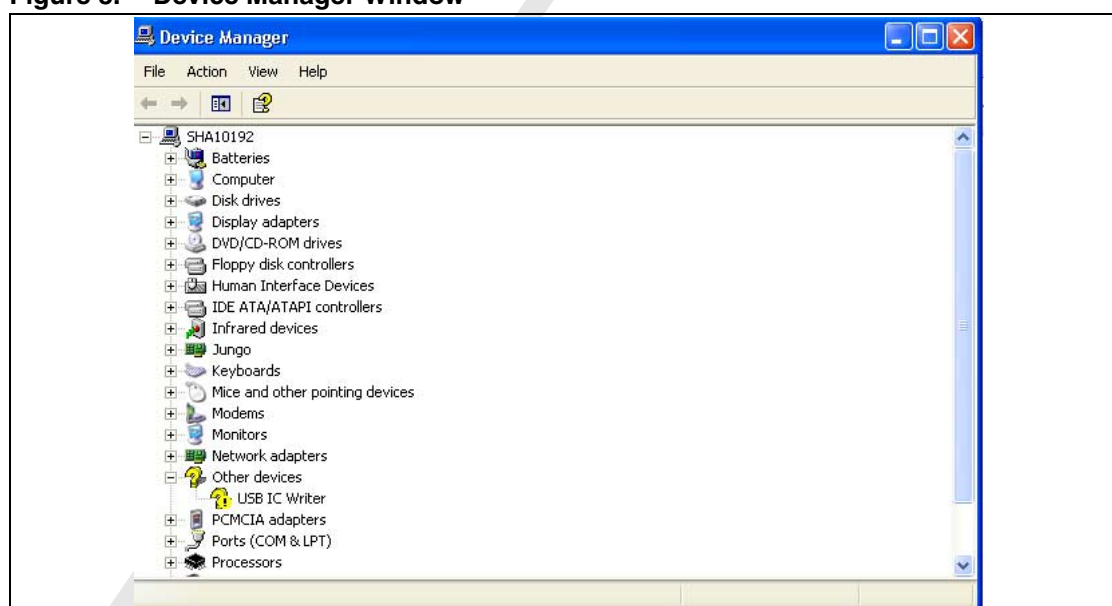
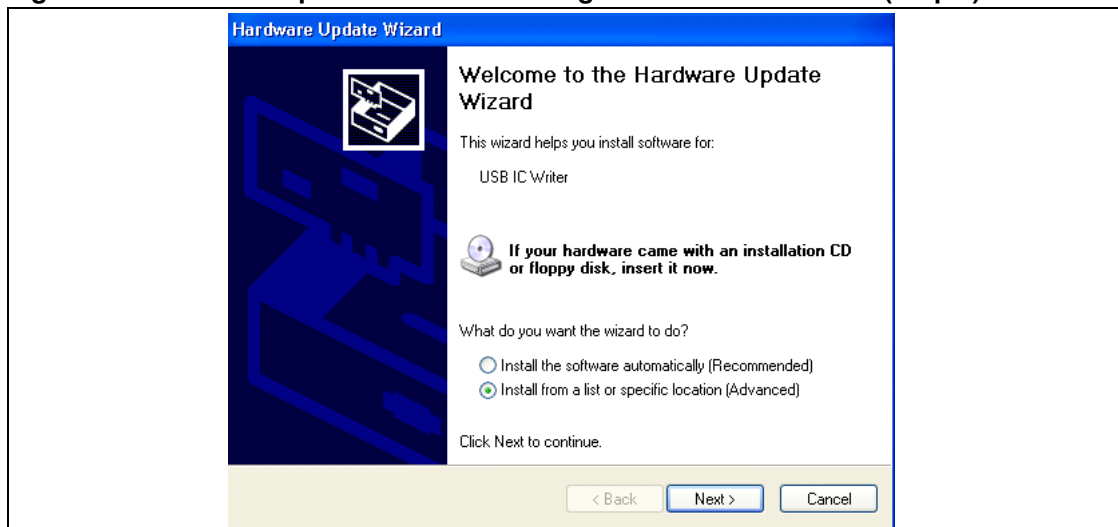
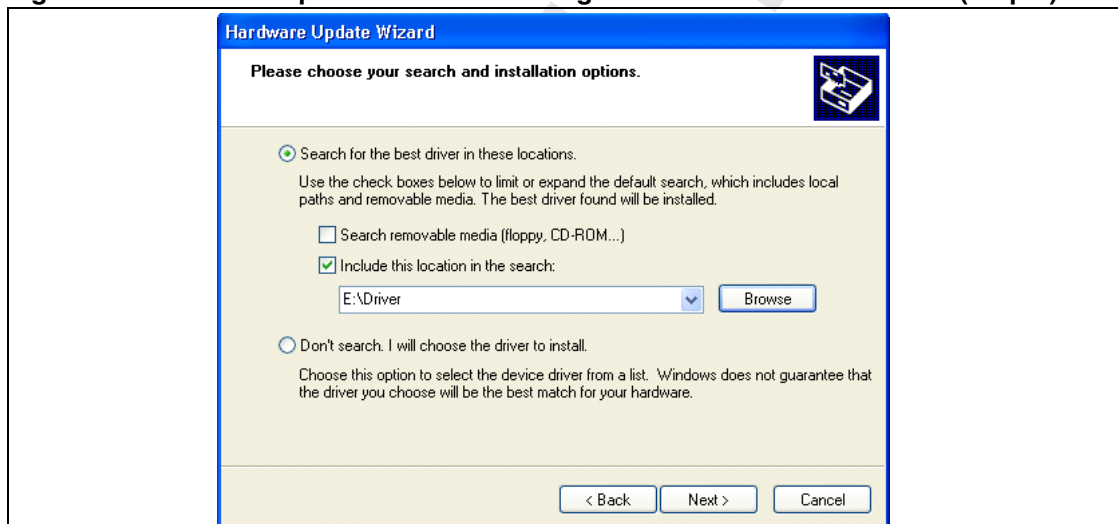


Figure 4. Hardware Update wizard: Selecting Advanced Installation (Step a)**Figure 5. Hardware Update wizard: Installing the driver from the CD-ROM (Step b)**

2.3.3 Installing the USB driver on Windows 2000

When the Host Computer power is turned on, a message is automatically displayed on the screen to indicate that it has detected a new USB plug-and-play hardware and that it is ready to install it.

To install the USB driver:

1. In the Control Panel, open the *Found New Hardware* wizard (see [Figure 6](#)).
2. Install the driver from the CD-ROM:
 - a) Select the *Specify a location* checkbox (see [Figure 7](#)).
 - b) Insert the CD-ROM. In the *CD-R:\Driver* subdirectory, select the *USBWriter.Inf* file and install the driver (see [Figure 8](#)).
 - c) In the *CD-R:\Driver* subdirectory, select the *USBWriter.Sys* file and re-do the installation using this file. (PLEASE CHECK)

Figure 6. Found New Hardware wizard

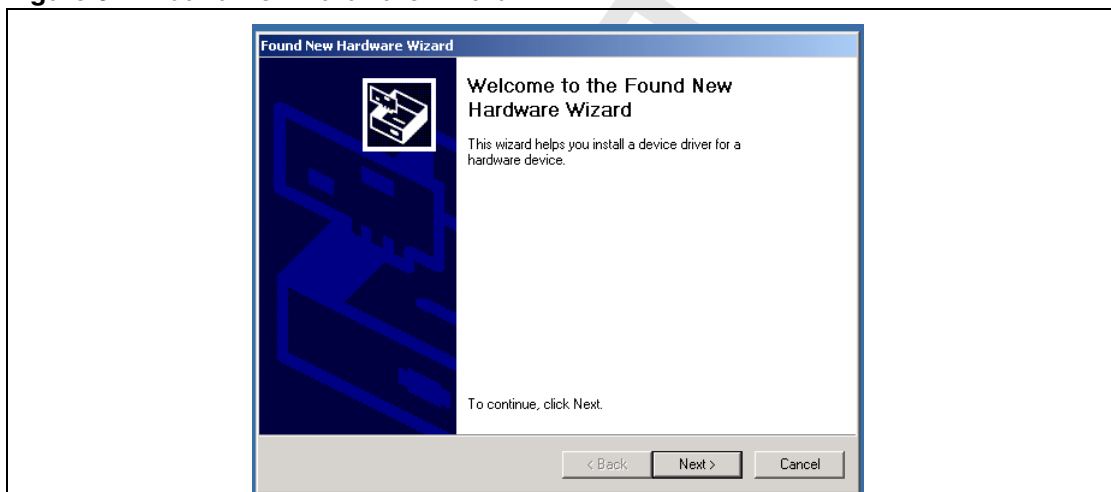


Figure 7. Hardware Update wizard: Installing the driver (Step a)

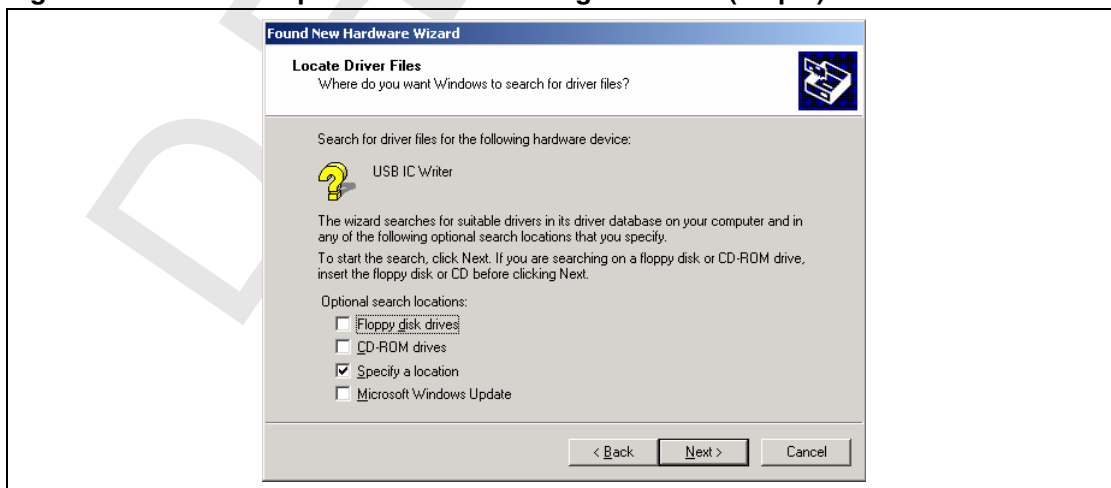
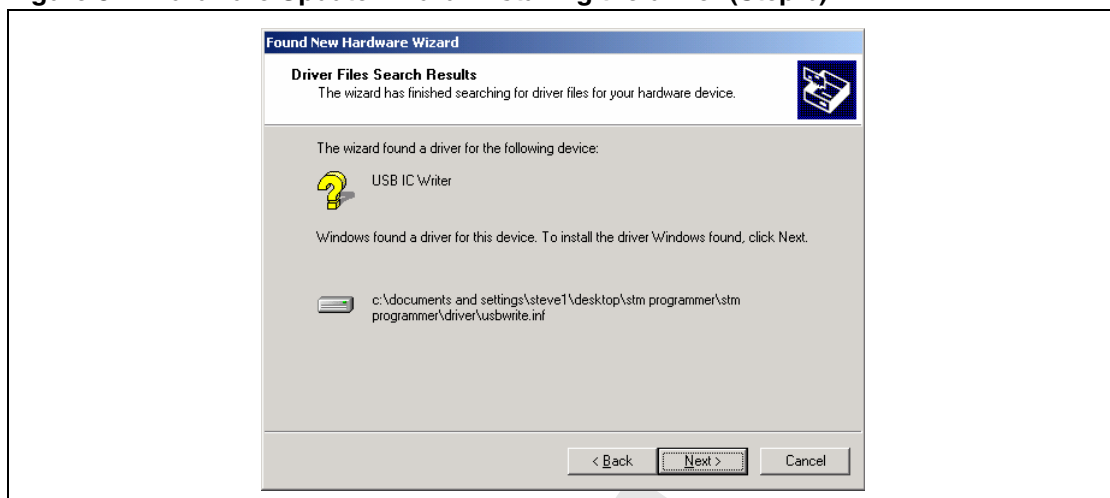
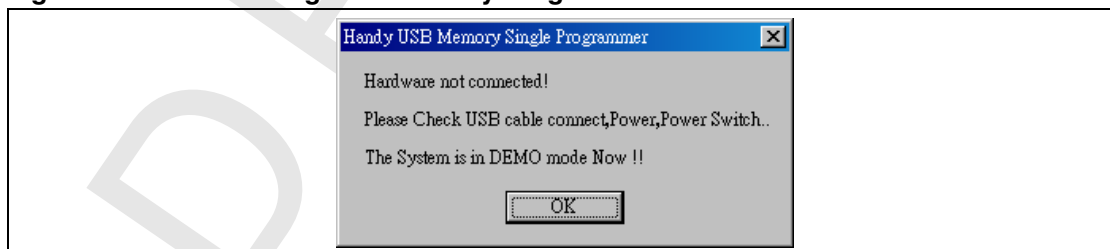


Figure 8. Hardware Update wizard: Installing the driver (Step b)

2.3.4 Troubleshooting

Once the USB driver is installed, if the Host Computer is not able to find the ST Memory Programmer when running the programming software, an error message is displayed on the screen (see [Figure 9](#)). In this case, the programming software runs in DEMO mode.

1. Check that the USB driver has been installed.
2. Check that the programmer power is turned on (see step 5 in [Section 2.2: Installing the hardware](#)).
3. Check that your Windows OS version supports USB Plug-and Play devices.
4. Check that the USB and power cables are properly connected to the programmer and to the computer.
5. If the *Auto Search* command still cannot find the driver, install it manually from the subdirectory *CD-R:\Driver*.

Figure 9. Error Message: ST Memory Programmer Not Connected

2.4 Recommendations and potential problems

- Configuration of the Host Computer:
 - To display correctly the programming software menus, the Host Computer screen resolution must be set to 96DPI and the *Font Size* of Windows must be configured to *small fonts*.
 - It is recommended to use a computer with a built-in USB port to avoid problems that may result from connecting the ST Memory Programmer to an external USB card. ST and Leap assume no responsibility for the consequences of using an external card.
- Use of socket adapters
 - The ST Memory Programmer is delivered with PLCC32, SOP16, TSOP48 and TSSOP8 Leap socket adapters. ST and Leap assume no responsibility for the consequences of using a non-Leap adapter for connecting the memory devices.
- Because the memory device may be submitted to unusual stress conditions during the programming stage, ST cannot guarantee a 100% programming success.
- When programming a master memory, it is strongly recommended to check the memory content by running a *Verify* operation.
- ST and Leap assume no responsibility for the damages or losses caused by inappropriate use of ST Memory Programmer.

2.5 Installing and using socket adapters

The ST Memory Programmer is delivered with a set of Leap socket adapters.

To install them:

1. Move the Textool socket handle up to an angle of 80°.
2. Put the adapter in the Textool socket.
3. Pull down the handle to clip tightly the adapter on the Textool socket.

Figure 10. Installing Socket Adapters



3 How to use the programming software

The programming software can be launched by clicking *STM Programmer* in the Start Menu. The Main Dialog Box is then displayed (see [Section 3.1](#)).

This section describes how to use the programming software to configure and execute programming operations.

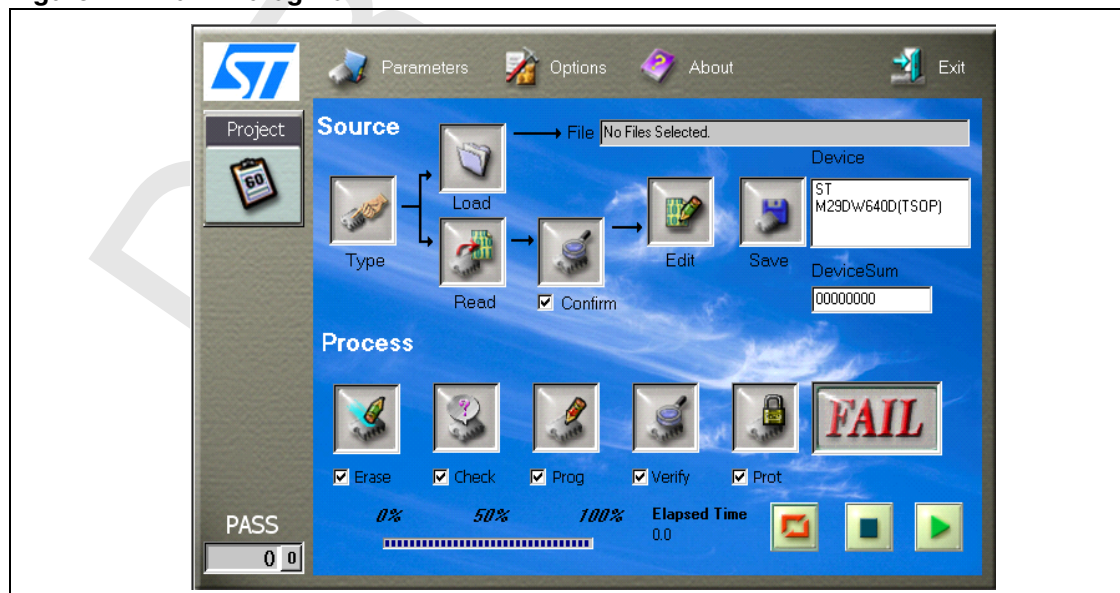
3.1 Description of the Main Dialog Box

The Main Dialog Box allows to access the programming software functions and options, through a set of commands (see [Figure 11: Main Dialog Box](#)):

- **Source section:** this section is used to select the device part number, and manage the data to be programmed (see [Section 3.2: Source section](#)).
- **Process section:** this section allows to start a programming process, and verify the programmed data (see [Section 3.3: Process section](#)).
- **Parameters** menus: these menus are used to define the device settings (see [Section 3.4: Parameters menus](#)).
- **Options** menus: this menu allows to configure the system options and self-test ([Section 3.5: Options menus](#)).
- **Project** menu: this option saves the user settings.
- **About** command: this option displays software and hardware versions
- **Exit** command: this option exits from the Programming Software.

A *Pass* read-only box shows the total number of successful programming processes (see also [Chapter 3.3: Process section](#)). The value can be reset to '0' by clicking the button on the right of the box.

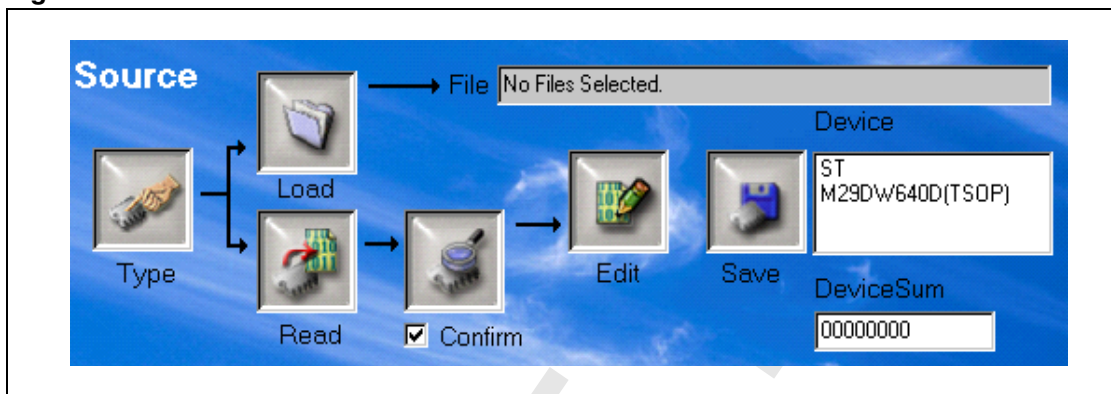
Figure 11. Main Dialog Box



3.2 Source section

The *Source* section is part of the Main Dialog Box (see [Figure 11](#) and [Figure 12](#)). It displays specific information related to the device (manufacturer, part number, data checksum) and the location and name of the file containing the data to be programmed. It is also used to access the *Type*, *Load*, *Read*, *Confirm*, *Edit*, and *Save* submenus.

Figure 12. Source section



3.2.1 Device read-only box

The *Device* read-only box displays the device manufacturer and part number.

3.2.2 DeviceSum read-only box

The *DeviceSum* read-only box displays the checksum of the data contained in the programmer buffer. When reading or loading data in the buffer, this value is computed taking into account the density of the memory device.

3.2.3 Type submenu

The *Type* submenu is displayed by clicking the *Type* icon in the Source section (see [Figure 13](#)). The *Type* submenu is used to select the device manufacturer and part number:

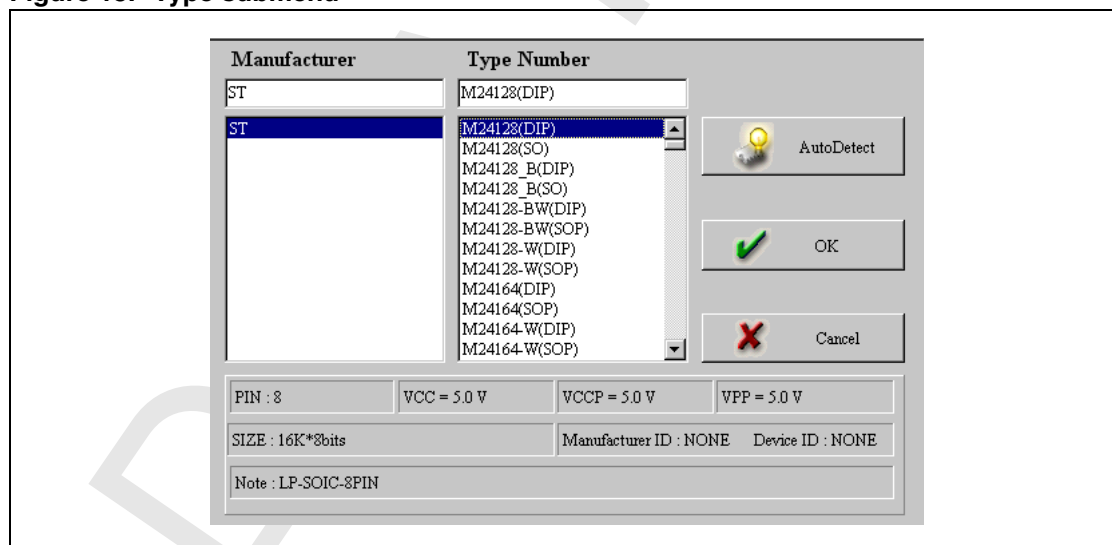
1. In the *Manufacturer* combo box, select the Device Manufacturer. Only STMicroelectronics is available.
2. In the *Type Number* combo box, type or select the device Type Number (part number and package).
3. Validate or cancel the information entered by clicking *OK*, or *Cancel* respectively.

The *Type* submenu also displays device characteristics through seven read-only boxes:

- *PIN*: shows the number of pins
- *VCC*: shows the supply voltage
- *VCCP*: shows the programming voltage
- *VPP*: shows the program voltage
- *SIZE*: shows the memory density
- *Manufacturer ID*: shows the manufacturer code
- *Device ID*: shows the device code
- *Note*: warns the user that an additional adapter is needed to program the device

An additional *AutoDetect* command automatically detects the Flash memory devices brand and part number.

Figure 13. Type submenu



Caution: The device manufacturer and type number must be defined before performing a Load or Read operation on a memory device.

3.2.4 Load submenu

The *Load* submenu is displayed by clicking the *Load* icon in the Source section (see [Figure 14](#)). This submenu configures the source data to be transferred from the Host computer to the ST Memory Programmer buffer and loads it.

To load source data:

1. Select the source data file either by typing the file name and location in the *File* textbox or by clicking the *Browse* command button.
2. Configure the source address space: under *Source Select*, enter the start and end addresses of the data to be loaded into the memory.
3. Configure the destination address space: under *Destination Select*, enter the start address to limit the destination area in the memory. To copy a specific value in the unused addresses of the memory, select the *Fill Unused* checkbox and enter the value in the textbox. The default value is FFh (blank).
4. Select the source data format among seven different formats: MS-DOS fn.exe, MS-DOS fn.com, binary/machine code, Intel hexadecimal, Motorola hexadecimal, Tektronix hexadecimal, and POF. If you are not sure of the format of the source data, select the *Auto Detect* checkbox to detect it automatically.
5. Once configured, click *Load* to confirm the load operation.
 - a) A pop-up window notifies the file format. Click *OK* if the format is correct (see [Figure 15](#)).
 - b) A *CheckSum* pop-up window then shows the checksum, the start and end addresses of the source data, and the start address of the data contained in the memory (see [Figure 15](#)). The checksums can be used to verify and confirm the memory data.
6. Click *Cancel* to abort the load operation.

Note: Steps 2, 3 and 4 are optional.

Figure 14. Load submenu

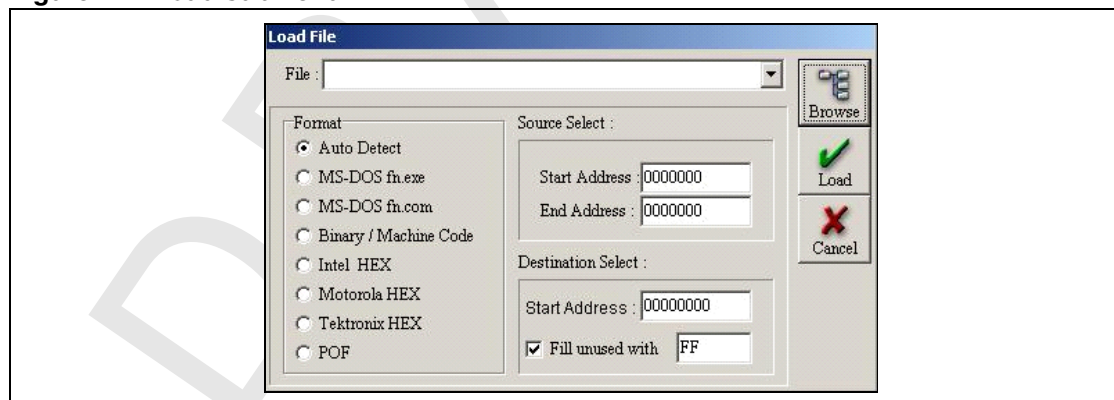
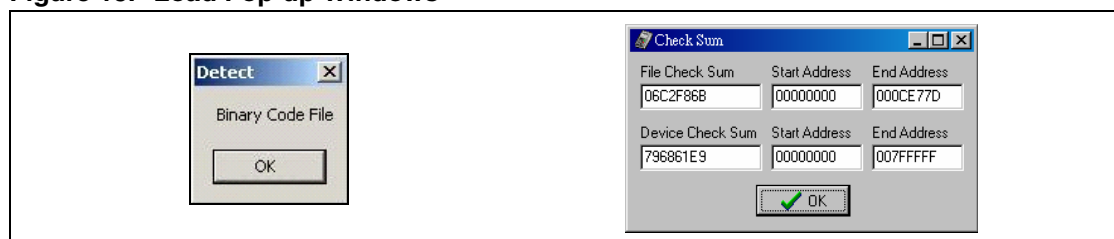


Figure 15. Load Pop-up Windows



3.2.5 Read and Confirm commands

Click the *Read* icon in the Source section to read the data contained in the memory (see [Figure 12](#)). The data is transferred from the memory to the ST programmer buffer and read from the buffer.

If the *Confirm* checkbox has been selected in the Source section of the Main Dialog Box (see [Figure 12](#)), the programming software will check that the data in the buffer area and the data in the memory are exactly the same.

If the device type is different from the type configured using the *Type* submenu, an error message is displayed (see [Appendix B, Section B.3: Device Type Error](#)).

Two pop-up windows are displayed when reading data from the memory device:

1. A pop-up window notifies the format of the read data.
2. Another pop-up window shows the checksum, start and end addresses of the read data.

3.2.6 Edit submenu

The *Edit* submenu is displayed by clicking the *Edit* icon in the Source section (see [Figure 12: Source section](#)). It is used to edit the data contained in the ST Memory Programmer buffer after a Read or Load operation.

Three windows are available by clicking the corresponding tab:

- *8 Bits*
- *16 Bits*
- *Buffer Information*

8 Bits window

The *8 Bits* window is accessed by clicking the *8 Bits* tab. It is divided into three sections that show 8-bit data in hexadecimal format (see [Figure 16: Edit Submenu/8 Bits Window](#)):

- The address section is used to select a hexadecimal address in the buffer. Click or enter the address of the data you want to display or edit.
- The hexadecimal data section displays the data in hexadecimal format. To change the data, click a data location and enter a new value.
- The ASCII data section displays data in ASCII format. To change the data, click a data location and enter a new value.

Starting from the selected address, two Bytes, Words or double Words can be swapped by clicking the corresponding button located on the top right side of the *8 Bits* window.

In addition, the *8 Bits* window allows to access the *Get IC Checksum*, *Block Process* and *Fill Data* submenus:

The **Get IC Checksum** submenu is displayed by clicking the *Sum* button (see [Figure 17](#)). This submenu is used to compute the checksum of a block of data contained in the buffer:

1. Configure the start and end addresses of the block of data - there are four different ways to configure it:
 - a) Under *Range*, enter the start and end addresses of the block, in 8-bit or 16-bit mode. The addresses must be entered in hexadecimal format.
 - b) Click the *IC* button to select a block of data corresponding to the memory device density. The block start and end addresses are updated automatically.
 - c) Click the *Max* button to select a block of data corresponding to the buffer size. The block start and end addresses are updated automatically.
 - d) Click the *File* button to select a block of data corresponding to the size of the file that has been loaded. The block start and end addresses are updated automatically.
2. Compute the data checksum or cancel the operation by clicking *OK*, or *Cancel*, respectively.

The **Block Process** submenu is displayed by clicking the *Block* button (see [Figure 18](#)). This submenu can be used to copy or move a block of data or to swap two blocks of data contained in the buffer:

1. Configure the start and end addresses of the source block - there are four different ways to configure it:
 - a) Under *Source Range*, enter the start and end addresses of the source block, in 8-bit or 16-bit mode. The addresses must be entered in hexadecimal format.
 - b) Click the *IC* button to select a block of data corresponding to the memory device density. The block start and end addresses are updated automatically.
 - c) Click the *Max* button to select a block of data corresponding to the buffer size. The block start and end addresses are updated automatically.
 - d) Click the *File* button to select a block of data corresponding to the size of the file that has been loaded. The block start and end addresses are updated automatically.
2. Under *Destination Range*, enter the start address of the destination block in 8-bit or 16-bit mode. The end address is automatically displayed in the *End Address* read-only box. If the end address is out of the memory device address space, an error message is displayed (see [Figure 19](#)).
3. Select the *Copy* checkbox to copy the source block to the destination address.
4. Select the *Move* checkbox to move the source block to the destination block. In this case, all the bits of the source block are set to '0'.
5. Select the *Swap* checkbox to swap the source block for the destination block.
6. Click *OK* to confirm the operation or *Cancel* to quit.

The **Fill Data** submenu is displayed by clicking the *Fill* button (see [Figure 20](#)). This submenu is used to fill in a block of data contained in the buffer. The block can be filled in bit by bit, Byte by Byte or Word by Word with a user-defined value or with random data or consecutive data.

1. Configure the start and end addresses of the block - there are four different ways to configure it:
 - a) Under *Source Range*, enter the start and end addresses of the block, in 8-bit or 16-bit mode. The addresses must be entered in hexadecimal format.
 - b) Click the *IC* button to select a block of data corresponding to the memory device density. The block start and end addresses are updated automatically.
 - c) Click the *Max* button to select a block of data corresponding to the buffer size. The block start and end addresses are updated automatically.
 - d) Click the *File* button to select a block of data corresponding to the size of the file that has been loaded. The block start and end addresses are updated automatically.
2. To fill in the block with a user-defined Byte or Word, under *User Define*, select the *Byte* checkbox or the *Word* checkbox, respectively. Enter the value of the Byte or Word to be copied at each address of the block.
3. To fill in the block with random data, under *Fill Data Select*, select the *Random Data* checkbox.
4. To fill in the block with consecutive Bytes or Words, under *Fill Data Select*, select the *Sequential Byte* and *Sequential Word* checkbox, respectively.
 - a) In 8-bit mode: the block is filled in with consecutive Bytes, starting from 00h. The data is incremented by 1.
 - b) In 16-bit mode: the block is filled in with consecutive Words, starting from 0000h. The data is incremented by 2.
5. To set all bits in the block with '1' or '0', under *Fill Data Select*, select the *All Bit 1* and *All Bit 0* checkbox, respectively.
6. Click *OK* to confirm the operation or *Cancel* to quit.

Figure 16. Edit Submenu/8 Bits Window

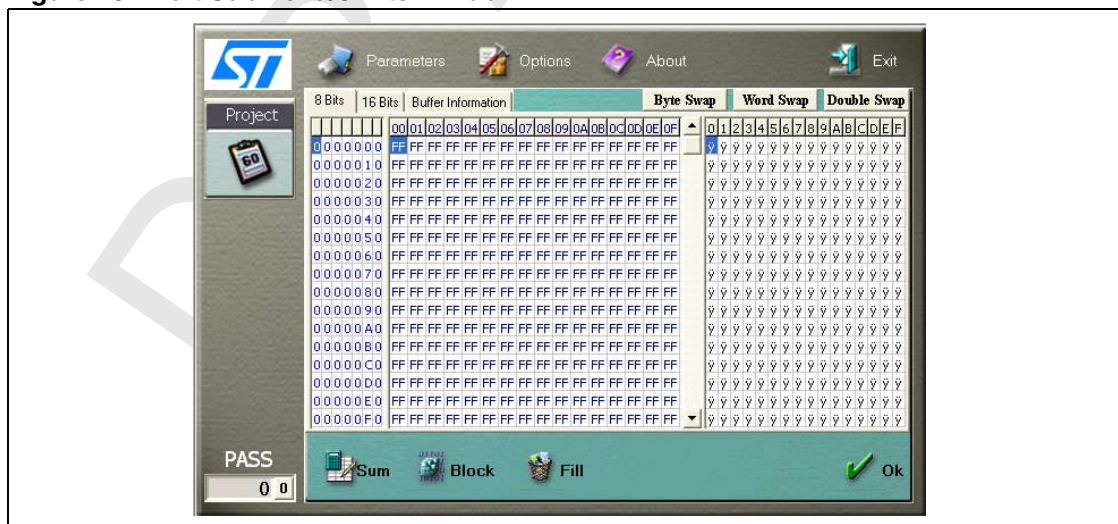


Figure 17. Get IC Checksum Submenu

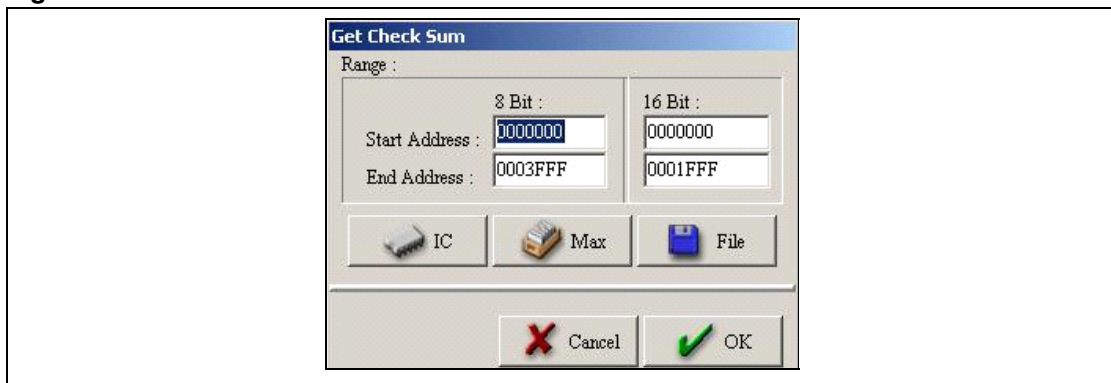


Figure 18. Block Process Submenu

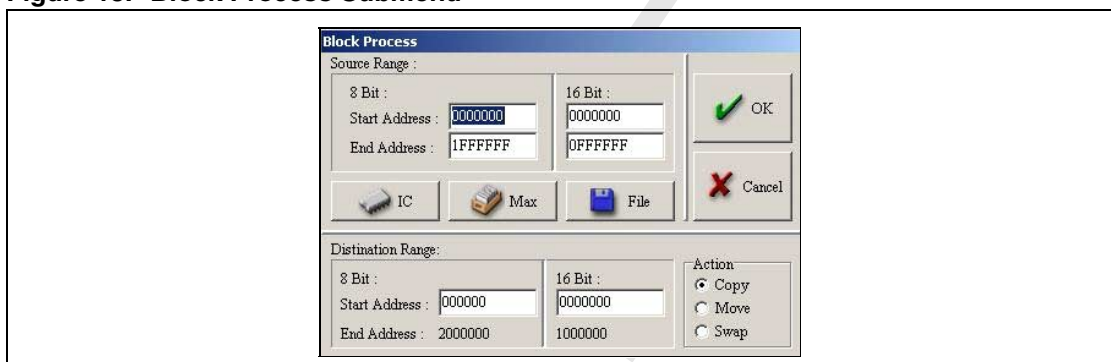
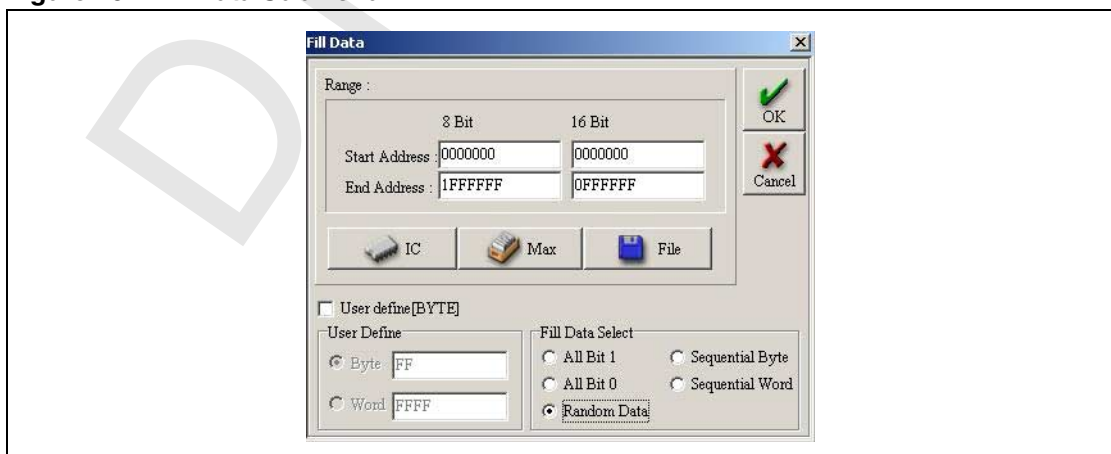


Figure 19. Destination Range Error Message



Figure 20. Fill Data Submenu



16 Bits window

The *16 Bits* window is accessed by clicking the *16 Bits* tab. The *16 Bits* tab shows 16-bit data in hexadecimal format. It is similar to the *8 Bits* window.

Buffer Information window

It is accessed by clicking the *Buffer Information* tab. It gives information concerning the data stored in the buffer area (see [Figure 21](#)).

Figure 21. Buffer Information Tab

The screenshot shows the 'Buffer Information' tab selected. It contains the following fields and controls:

- 8 Bits | 16 Bits | **Buffer Information**
- Last Load File Name :
- File Format : Auto Detect
- Start Address : 0000000 | End Address : 0000000 | Buffer Size : 2000400
- Check Sum : 00000000 H | Get Sum Range : 0000000H <—> 0000000H

3.2.7 Save submenu

The *Save* submenu is displayed by clicking the *Save* icon in the Source section of the Main Dialog Box (see [Figure 22: Save submenu](#)). It is used to save the data that has been edited using the *Edit* command. The data is saved in a file on the host Computer hard disk.

To save data:

1. Select the file name either by typing the file name, patch and subdirectory in the *Name* textbox or by clicking the *Browse* command button.
2. Define the address space containing the data to be saved: under *Save Range*, enter the start address and end address in 8-bit hexadecimal format.
3. Select the file format among six different formats: binary/machine code, Intel hexadecimal, Motorola %S(S1), Motorola S2, Motorola S3, Tektronix hexadecimal. If an inappropriate format is selected, wrong data will be saved.
4. Allocate the memory space to save the data:
 - a) Click *File* to allocate it according to the size of the selected file.
 - b) Click *Max* to allocate it according to the buffer size.
 - c) Click *IC* to allocate it according to the memory density of the selected device.
5. Once configured, click *Save*, to confirm the save operation.
6. The save operation can also be aborted by clicking *Cancel*.

Figure 22. Save submenu

The screenshot shows the 'Save File' dialog box with the following elements:

- Name : [text box]
- Browse button
- Format section with radio buttons:
 - ☒ Binary/Machine Code
 - ☐ Intel Hex
 - ☐ Motorola %S(S1)
 - ☐ Motorola S2
 - ☐ Motorola S3
 - ☐ Tektronix HEX
- Save Range section:
 - Start Address [HEX-Byte] : 000000
 - End Address [HEX-Byte] : 1FFFFFFF
- File, Max, and IC buttons (each with a corresponding icon)
- Save button (green checkmark)
- Cancel button (red X)

3.3 Process section

The *Process* section is part of the Main Dialog Box (see [Figure 11](#) and [Figure 23](#)).

It is used to configure the sequence of operations that make up the programming process:

- *Erase* operation: select the *Erase* checkbox to add an erase operation to the programming process.
- *Check* operation: select the *Check* checkbox to add a check operation to the programming process. This operation verifies that the content of the memory is blank.
- *Prog* operation: select the *Prog* checkbox to add a program operation to the programming process. This operation programs the data loaded or read into the target memory.
- *Verify* operation: select the *Verify* checkbox to add a verify operation to the programming process. This operation compares the data programmed in the memory to the source data.
- *Prot* operation: select the *Prot* checkbox to activate the memory protection configured using the Device Option submenu (see [Section 3.4.3](#)).
- ►: this button starts the programming process.
- ■: this button stops the programming process.

A progress bar shows the progression of the programming process.

A read-only box shows the status of the programming process:

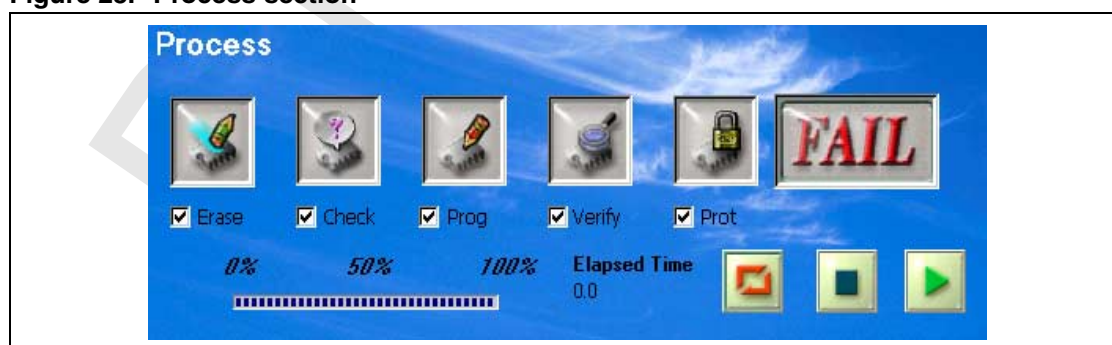
- FAIL: at least one of the operations in the programming process failed.
- PASS: all the operations succeeded.
- WAIT: the programming process is ongoing.

The background color of this read-only box gives an additional indication on the programming process status:

- The yellow color indicates that the programming process is ongoing.
- The green color indicates that the programming process is over and that a new process can be started.

Note: By default, the *Erase*, *Check*, *Prog* and *Verify* checkboxes are selected. Unselect the checkbox if you do not want the corresponding operation to be performed.

Figure 23. Process section



3.4 Parameters menus

The *Parameters* menus are displayed by clicking the *Parameters* icon in the Main Dialog Box toolbar (see [Figure 11](#)). The *Parameters* menus are used to set more precisely some device features, to configure the verify operations, and the memory protection.

The parameters differ from one memory device to another. They must be consistent with the ones specified in the datasheet.

Three submenus can be selected by clicking the corresponding tab:

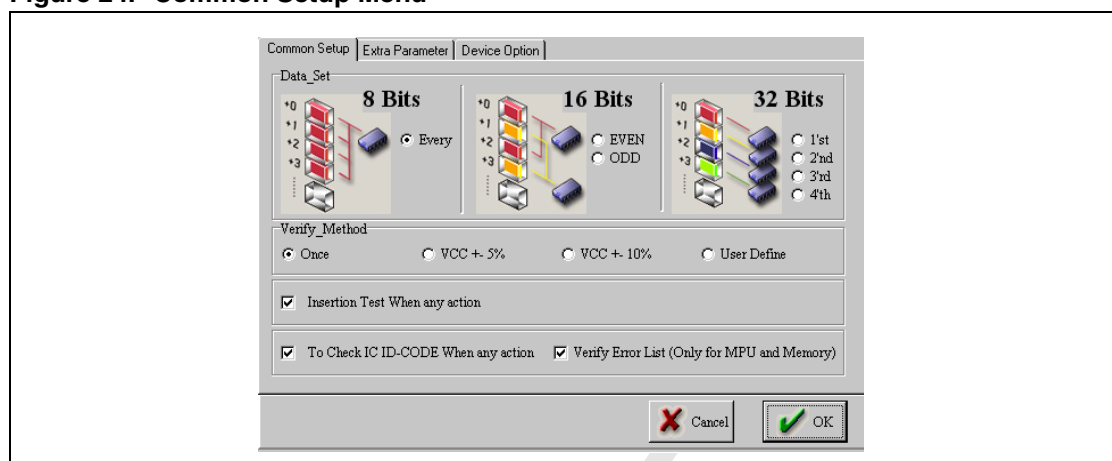
- *Common Setup*
- *Extra Parameter*
- *Device Option*

3.4.1 Common Setup submenu

The *Common Setup* submenu is used to configure the type of data and the verify operations performed by the ST Memory Programmer (see [Figure 24](#)):

1. Under *Data_Set*, configure the type of data to be programmed in the memory:
 - *8 Bits-Every* to load 8 bits of data in one shot.
 - *16 Bits* to split data into Lower and Upper bytes, loaded them successively in the memory.
 - *32 Bits* to split data into four bytes, loaded successively in the memory.
2. Under *Verify_Method*, configure the minimum and maximum values of the supply voltage used by the ST Memory Programmer, together with the number of verify operations:
 - $V_{CC} \pm 5\%$
 - $V_{CC} \pm 10\%$
 - *User define*
 - *Once*: the verify operation is performed once.
3. Configure the verify operations:
 - Select the *Insertion Test When any action* checkbox to configure the programmer to check that the device is correctly clipped into the socket.
 - Select the *Check IC ID-CODE When any action* checkbox to configure the programmer to check the Device Code.
 - Select the *Verify Error List* checkbox to list the address and data where an error was detected while running a verify or a confirm operation (see [Section 3.2: Source section](#)).
4. Validate or cancel the information entered by clicking *OK*, or *Cancel* respectively.

Figure 24. Common Setup Menu



3.4.2 Extra Parameter submenu

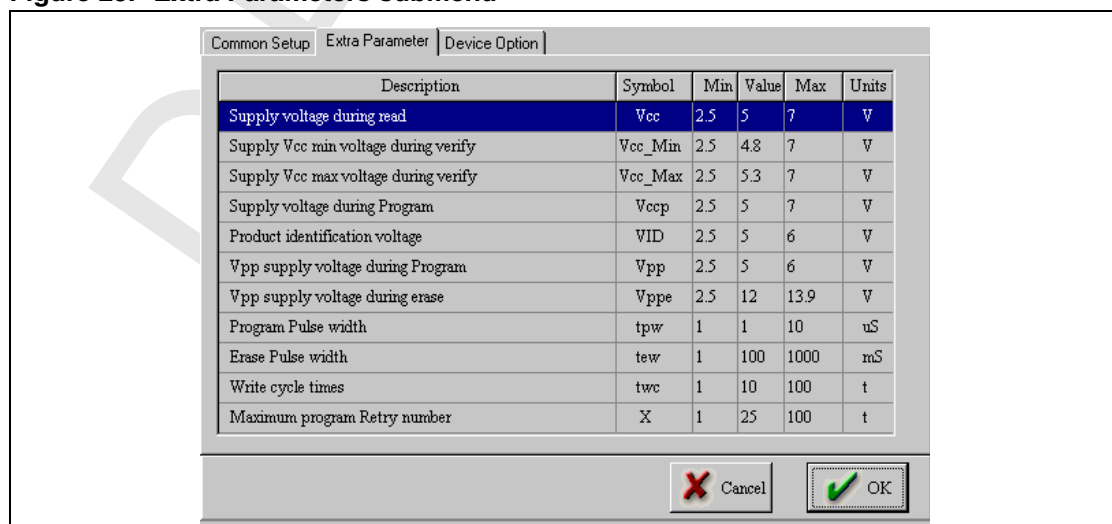
The *Extra Parameter* submenu shows detailed electrical characteristics of the memory device (see [Figure 25](#)):

- *Description*
- *Symbol*
- *Min*: minimum value
- *Value*: typical value
- *Max*: maximum value
- *Unit*

The user can modify the typical value of an electrical characteristic by clicking the corresponding row and entering the new value (EPROMs only). The information entered can then be validated or cancelled by clicking *OK*, or *Cancel* respectively.

Minimum and maximum values are fixed and cannot be changed by the user.

Figure 25. Extra Parameters submenu



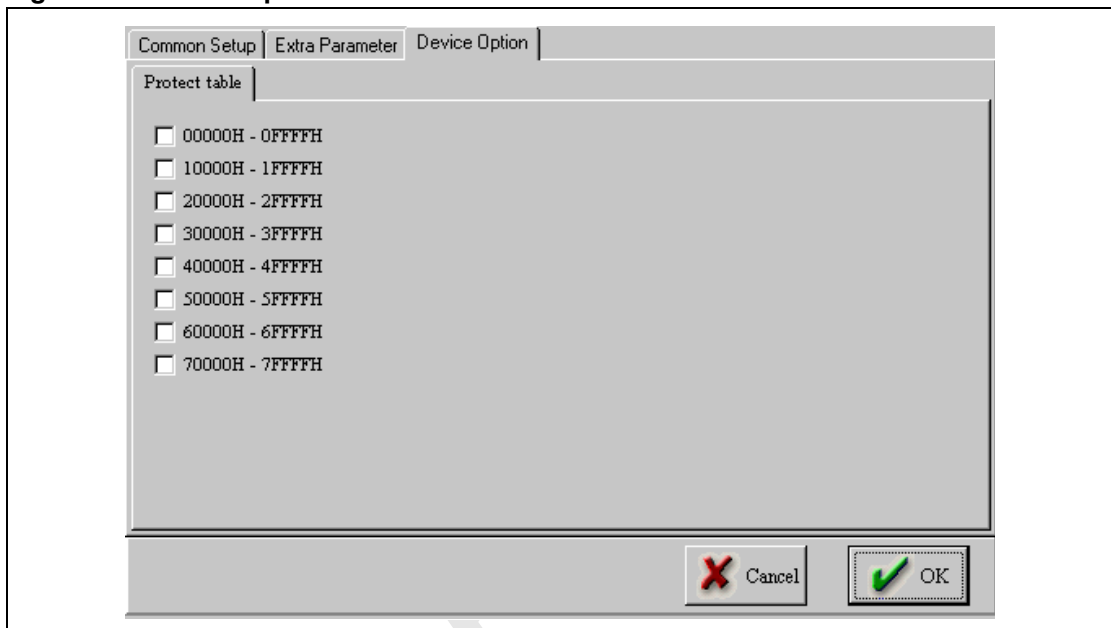
3.4.3 Device Option submenu

The *Device Option* submenu is used to configure the memory areas that are protected from program and erase operations (see [Figure 26](#)). One or multiple areas can be selected to be protected.

The protection is activated by clicking the *Prot* checkbox in the Process section of the Main Dialog Box (see [Section 3.3](#)).

Note: This function is used only for some memory devices.

Figure 26. Device Option submenu



3.5 Options menus

The *Options* menus are displayed by clicking the 'Options' icon in the Main Dialog Box toolbar (see [Figure 11](#)).

Two submenus can be selected by clicking the corresponding tab:

- *System Option* submenu
- *Self test* submenu

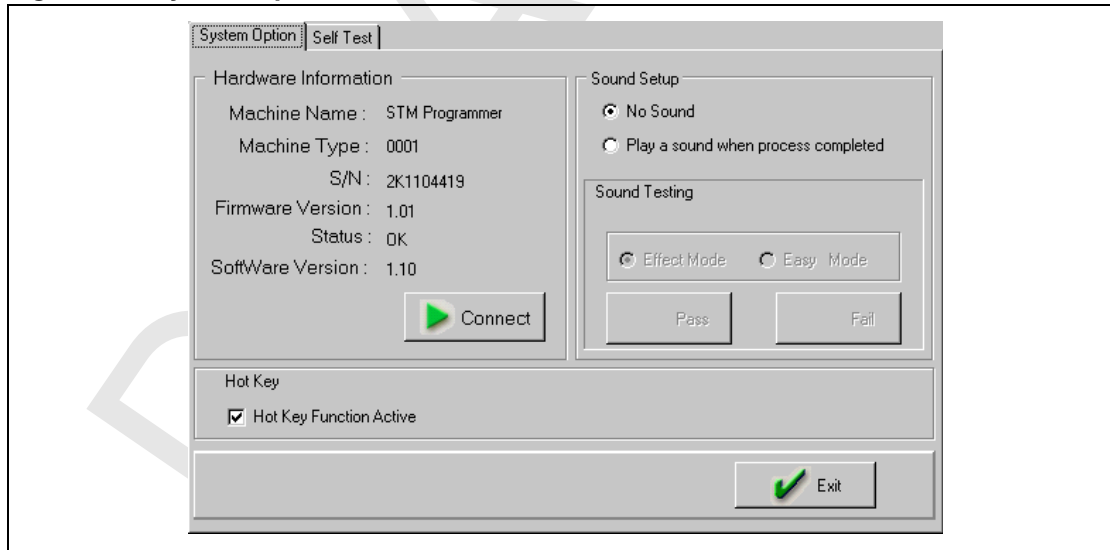
3.5.1 System Option submenu

The System Option submenu displays ST Memory Programmer software and hardware global settings (see [Figure 27](#)):

- *Hardware Information*: ST Memory Programmer information and status. Click the *Connect* button to check the connection status between the ST Memory Programmer and the Host Computer.
- *Sound Setup*: the programming software can be configured to produce a sound when the programming process is completed.
- *Sound Testing*: this function allows to test the sound.
- *Pass and Fail buttons*: click the *Pass* or *Fail* button to check the sound produced when the programming process passes or fails, respectively.
- *Hotkey*: select the *Hotkey Function Active* checkbox to enable the hotkeys (see [Section Appendix A: Hotkeys](#)).

The *System Option* submenu is exited by clicking the *Exit* button.

Figure 27. System Option submenu



3.5.2 Self test submenu

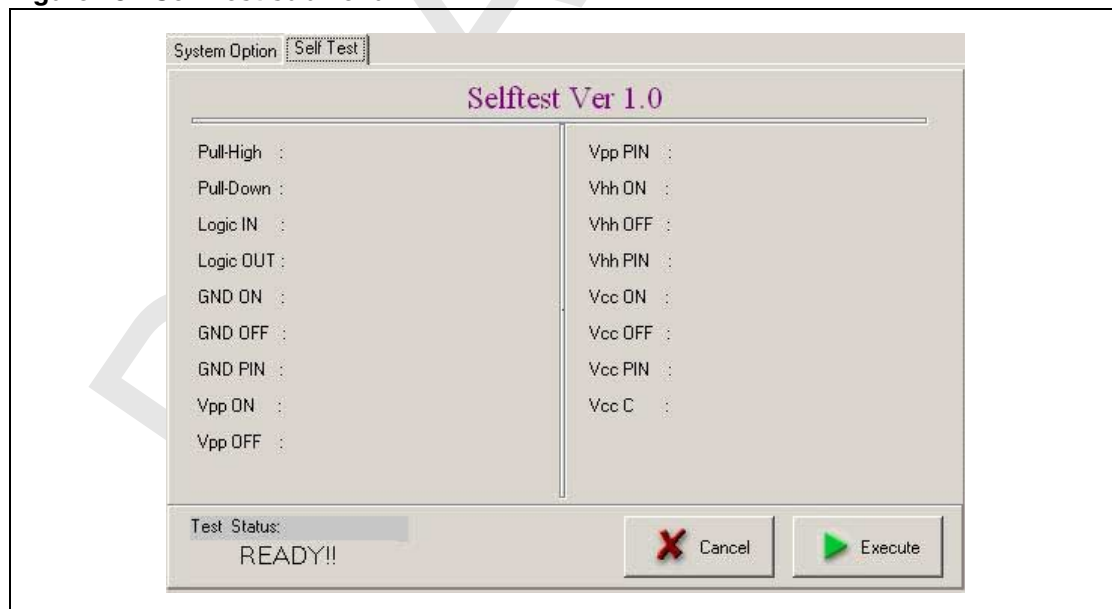
The ST Memory Programmer can be configured to perform a hardware self-test. This may help maintenance people to quickly identify the problems and find adapted solution.

The *Self Test* submenu displays the ST Memory Programmer hardware status (see [Figure 28](#)):

- *Selftest Ver* is the version of the self-test program.
- Results of the hardware tests: the test points are displayed together with the results of the test. The result can be either PASS or FAIL.
 - If the result is FAIL, it means that the ST Memory Programmer does not work correctly. Please contact Leap support service. Do not try to dismantle the equipment. STMicroelectronics and Leap assumes no responsibility for resulting damages.
 - Otherwise, the result is PASS.
- Test Status: the test status can be
 - *READY!!* if the ST Memory Programmer is ready to start performing a self-test.
 - *Not Connected* if the ST Memory Programmer is not connected to the Host Computer.
 - *Time Out Stop detecting*: if the programmer detection timeout has been exceeded.
 - *Completed*.
- *Cancel*: click *Cancel* to go back to previous menu.
- *Execute*: click *Execute* to run a self-test.

Note: Make sure to remove the memory device from the ST Memory Programmer before running a self-test, otherwise the device may be damaged. STMicroelectronics and Leap assumes no responsibility for the resulting damages.

Figure 28. Self Test submenu



3.6 Project menu

When programming a device, the user may modify some ST Memory Programmer parameters, load specific data files, etc. The user settings can be saved in a file, by clicking the *Project* icon in the Main Dialog Box (see [Figure 11](#)). When starting a new programming process, the user can select the Project containing the customized settings

The *Project* menu is then displayed, showing the list of existing projects (see [Figure 29](#)). A project is characterized by the information listed below:

- *Description*: name and description of the project.
 - *Device*: memory brand, Part Number and package.
 - *Auto*: enable/disable auto programming devices.
 - *Lock*: enable/disable project protection
1. To create and save a new project:
 - a) Double click on a blank row to create the project.
 - b) Click the *Create* button. The *Save Project* submenu is then displayed (see [Figure 30](#)).
 - c) Under the *Save Project* submenu:
 - Enter the Project Description and Device Information
 - Select the Auto Run configuration: select *Once* to perform single programming operation, and *Batch* to perform a sequence of consecutive programming operations.
 - d) Click *Cancel* to abort the operation or *OK* to save the project. All the settings will be recorded (device type, parameters, data, etc.) together with the current date and time and the process selected by the user (see [Section 3.3: Process section](#)).
 2. To delete a project:
 - a) Click the project row to select the project
 - b) Click the *Delete* button.
 3. To run a project:
 - a) Either double click the project row to run the project
 - b) or Click the *Execute* button.
 4. To exit from the Project menu, click the *Cancel* button.

Figure 29. Project menu

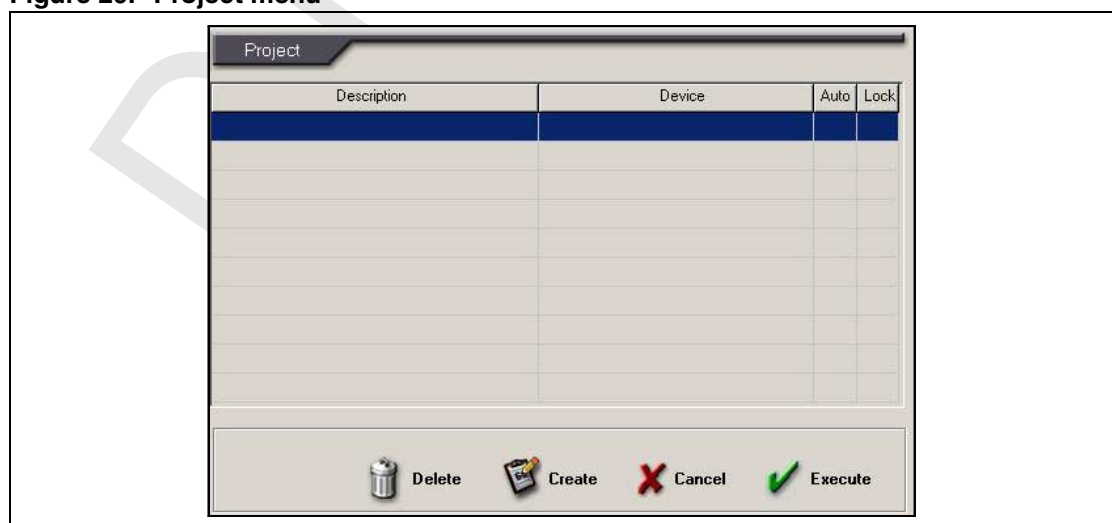
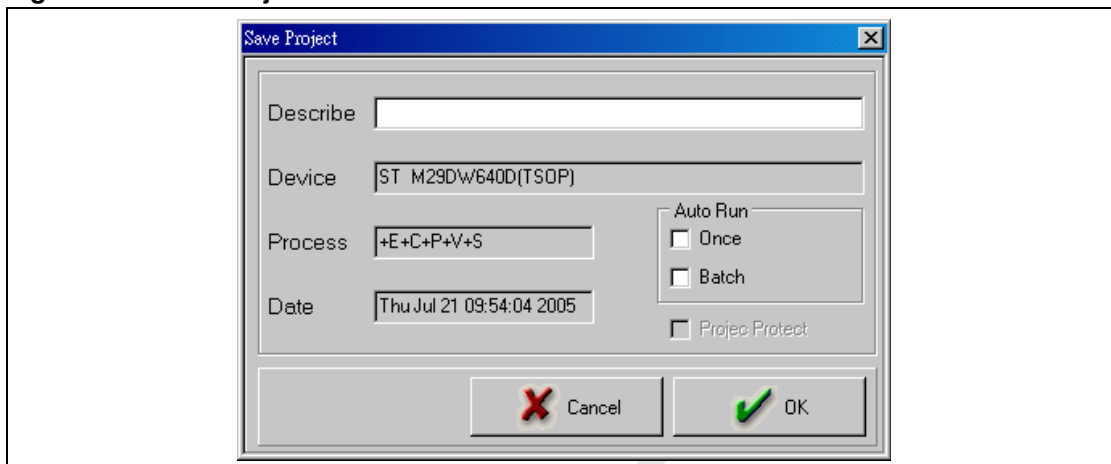


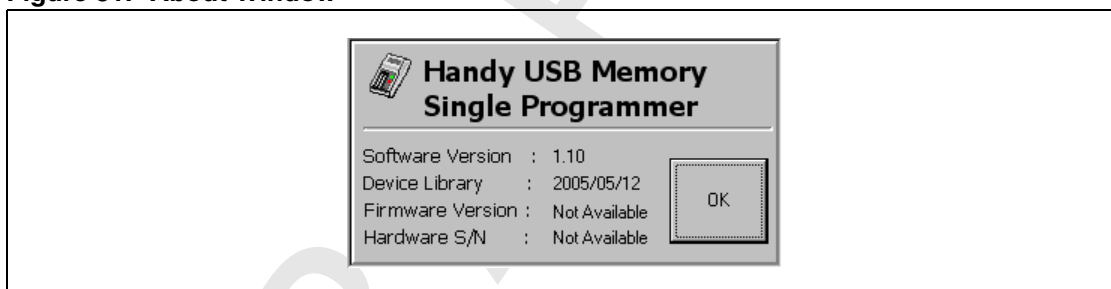
Figure 30. Save Project submenu



3.7 About command

When selecting the *About* command, an information window is displayed containing the ST Memory Programmer hardware and software versions (see [Figure 31](#)).

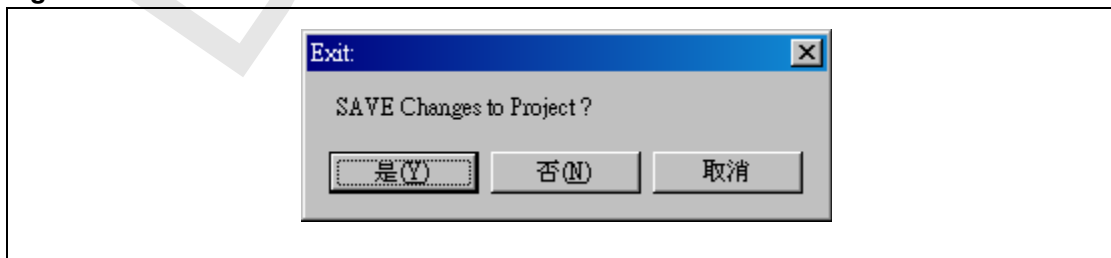
Figure 31. About Window



3.8 Exit command

The *Exit* command closes the programming application. The user can either save its settings in a Project file (see [Section 3.6: Project menu](#)) or exit without saving (see [Figure 32: Exit window](#)).

Figure 32. Exit window



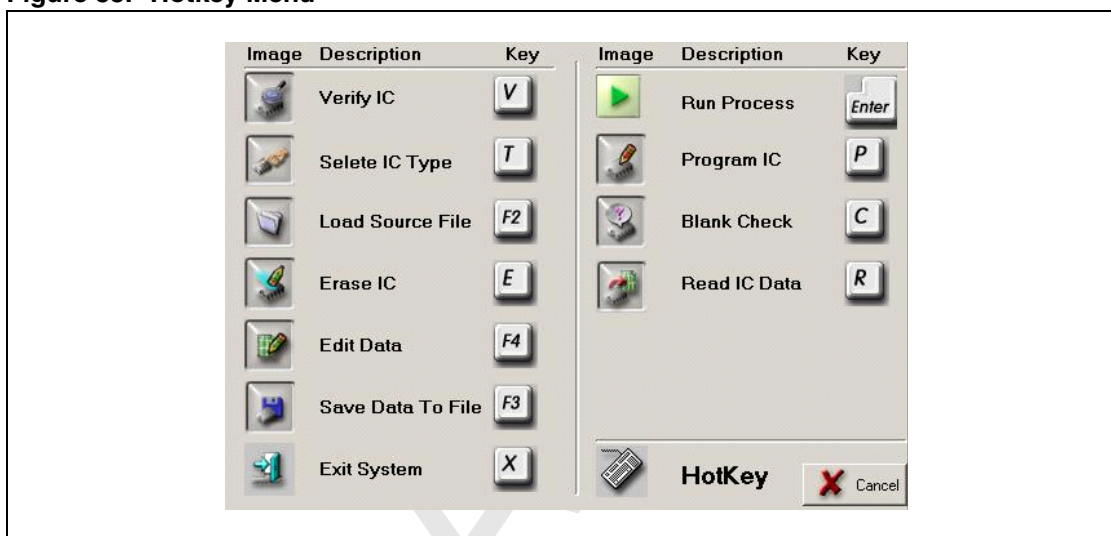
Appendix A Hotkeys

A set of hotkeys are available to select the main programmer operations. The Hotkeys menu can be accessed by pressing the keyboard space bar when the Main Dialog Box is open. The Hotkey menu opens (see [Figure 33](#)). It shows the Hotkey corresponding to each command of the Main Dialog Box.

The user can activate a hotkey by clicking the corresponding button.

The Hotkey menu is exited by clicking *Cancel*.

Figure 33. Hotkey Menu



Appendix B Error Messages

The ST Memory Programmer can perform verify operations on the memory device. The programmer can be configured to perform them automatically when a memory device is clipped into the Textool (see [Section 3.4.1: Common Setup submenu](#)).

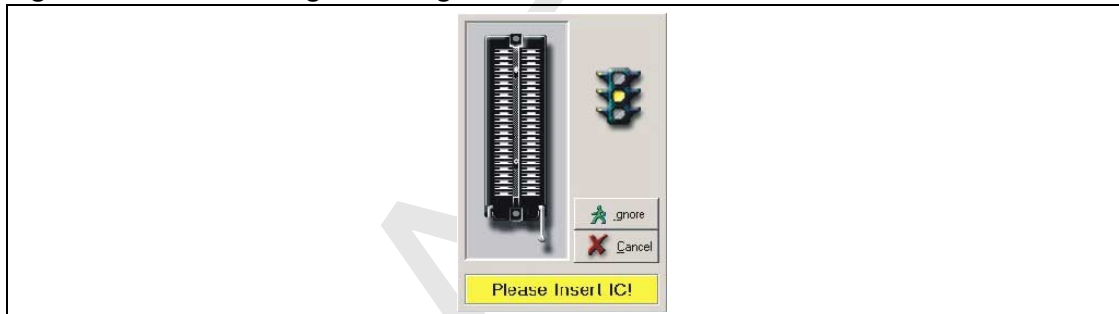
B.1 Missing Device Error

If no device has been inserted in socket, the ST Memory Programmer displays an error message to remind the user of inserting the device (see [Figure 34](#)). The error message shows:

- the device insertion status
- the working LED is yellow to indicate that a device must be inserted in the socket.
- the recommended action: 'Please Insert IC'

Click *Cancel* to abort the ongoing operation, and *Ignore* to carry it on.

Figure 34. Error message: Missing Device



B.2 Device Insertion Error

If the device is not correctly clipped on socket, the ST Memory Programmer displays an error message to warn the user (see [Figure 35](#)). The error message shows:

- the device insertion status
- the working LED is yellow to indicate that the device is not correctly inserted in the socket.
- the recommended action: 'Please Replace IC'

Click *Cancel* to abort the ongoing operation, and *Ignore* to carry it on.

Figure 35. Error message: Device Insertion Problem



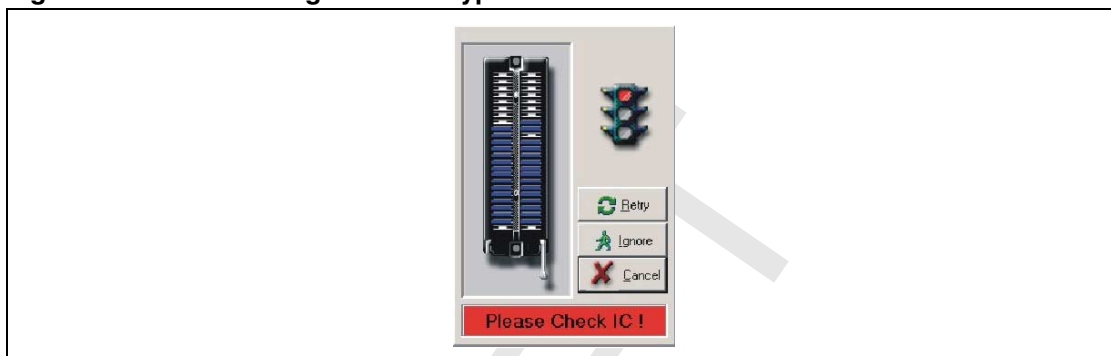
B.3 Device Type Error

If the type of the device clipped into the socket is not consistent with the one specified using the [Type submenu](#), the ST Memory Programmer displays an error message to warn the user (see [Figure 36](#)). The error message shows:

- the working LED is red
- the recommended action: 'Please Check IC'

Click *Cancel* to abort the ongoing operation, *Ignore* to carry it, or *Retry* to restart it.

Figure 36. Error message: Device Type To Be Checked



4 Revision history

Date	Revision	Changes
17-Oct-2005	1	Initial release.
xx-xxx-2005	2	Description of USB driver installation on Windows XP and 2000 deeply updated (Section 2.3.2: Installing the USB driver on Windows XP and Section 2.3.3: Installing the USB driver on Windows 2000).

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics.
All other names are the property of their respective owners

© 2005 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan -
Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com