User's Guide

XELTEK

Superpro®5000

Ultra-Fast, Stand-Alone, 144pin Programmer of the Future

XELTEK

1296 Kifer Rd. Unit 605
Sunnyvale, CA 94086
Tel: (408) 530-8080
Fax: (408) 530-0096
www.xeltek.com

XELTEK

1296 Kifer Rd. Unit 605
Sunnyvale, CA 94086
Tel: (408) 530-8080
Fax: (408) 530-0096
www.xeltek.com

SUPERPRO is the registered trademark of XELTEK.

Distribution and sales of this product are intended for use by the original purchaser under the terms of the License Agreement.

This document may not, in whole or part, be copied, photocopied, reproduced, translated or reduced to any electronic medium of machine-readable form without prior consent in writing from XELTEK.

The information in this document is subject to change without notice.

Software Copyright **2008-2011** XELTEK

User's Guide Copyright **2008-2011** XELTEK

Table of Contents

Welcome 6
Feature Highlights 6
User Manual Organization 7
Chapter 1 General Description 8
Introduction to SUPERPRO 8
Package Contents 8
Software Features 8
System Requirements 9
Chapter 2 System Installation 10
Software Setup 10
CD-ROM 10
Xeltek Website 11
Setup Process 11
Hardware and Driver Installation 16
Chapter 3 SUPERPRO Software Features 18
Main Screen 18
Main Menu 19
File Menu 19
Load File 19
Save 22
Recent Projects 23
Exit 23
Buffer Menu 23
Edit 23
Encryption table 28
Test Vector Table 28
Device Menu 29
Select Device 30
Dev. Information 31
Device Configuration Word 33

Option Menu 35
Operation Option 35
Edit Auto 40
Production Mode 41
Log File 42
Project Menu 42
Load Project 43
Save Project Files 43
Downloading Library (Standalone) 43
Help Menu 44
Tool Bar 44
Checksum 45
Main Screen Device Information Display 46
Main Screen File Information 46
Operation Option Bar 47
Data Compare 48
Device Operation Window 49
Operation Information Window 52
Status Bar 52
Statistic Window 52
Chapter 4 Procedures 54
Programming Procedure 54
Prepare the Hardware 54
Select the device 54
Load Data into Buffer 54
Set Options 55
Edit Auto 55
Configure the Device 55
Review the Device Information 56
Execute Programming 56
Execute Production Mode to Program Chips in Batches 56
Security and Password Input 57

Downloading Library Creation 58
Creating Project Files 58
Create a downloading library and download. 59
Error messages: 59
Making a Chip Blank 59
Identifying the Sector 60
Factory Mode 61
Factory Mode Report 62
Chapter 5 Detail Explanations on Common Problems 63
Dealing with Data Files 63
Solve Invalid File Type or File Data Overflow Error 63
Separate File Data for Programming 63
Program Two or More Files to One Chip 64
POF Format Files 65
Adapter Selection 65
Selecting the Correct Adapter 66
Other Messages 67
Appendix 70
Troubleshooting 70
Contact Information 70
License Agreement 71
Warranty 72

Welcome

Welcome to the world of Xeltek Superpro programmers. Xeltek has produced the Superpro line of IC programmers since 1985. Our motto is to provide high value products at affordable prices. In the past 23 years, Xeltek has produced a string of highly popular and successful programmers including Superpro, SuperproII, SuperproIII, SuperproV, Superpro680, Superpro8000, Superpro3000U, and Superpro9000 models.

Superpro3000U is the first model to incorporate both PC connected and stand-alone modes operations combined into one programmer. It is also the first programmer to break the barrier of 40,000 devices supported in the industry. Superpro3000U is by far the most popular IC programmer in the industry today.

Pushing the barrier further yet, Xeltek is proud to announce the family of Superpro 5000 programmers. Superpro5000 retains all the good features of Superpro3000U, but, in addition, it is designed to program today's large size devices efficiently and effectively.

Feature Highlights

- The SUPERPRO includes ultra high speed programming via 32bit RISC MCU device. This is especially suitable for programming high capacity NAND Flash devices
- A 144 pin-driver support is built-in to provide efficient programming of large pin count devices, an additional pin expander module is not necessary to program large pin-count devices.
- The SUPERPRO operates with a PC for engineering purpose or in stand-alone mode, without a PC, for repeat production mode programming. This provides seamless migration from engineering to production.
- The SUPERPRO communicates through a USB 2.0 port and operates on most Pentium-based, IBM compatible desktop/notebook computers.
- The SUPERPRO comes with a standard 48-pin DIP ZIF socket and supports optional socket adapters to accommodate PLCC, TSOP, SOIC, SOP, QFP, TSSOP and BGA package types.
- The programmer and software supports Windows XP and Windows Vista.
- The programmer is scheduled to support 43,000+ IC devices by the end of 2008. Please check www.xeltek.com for the list of the currently supported devices.
- For authorized users, the SUPERPRO can provide for automatic generation of electronic serial numbers.

IMPORTANT!	Make sure to complete the software installation before connecting the programmer USB cable.

User Manual Organization

This manual includes the following:

- An introduction to the SUPERPRO series, including the system requirements and the software and hardware installation
- Detailed explanations on the commands and the programming procedures
- Information about error messages and common problems
- An Appendix that includes the customer support information

Note: The software prevails in any discrepancy between it and the user manual, due to any upgrades of the software.

The information in this document is subject to change without notice.

Chapter 1 General Description

This chapter provides information on the package contents, software features, and system requirements for SUPERPRO.

Introduction to SUPERPRO

The SUPERPRO is a menu-driven software that operates a series of cost-effective, reliable, and high-speed universal programmers. SUPERPRO communicates through a USB 2.0 port and operates with most IBM-compatible desktop and notebook computers based on Pentium and AMD or other processors.

Package Contents

The SUPERPRO programming hardware includes the following items:

- A programming module (48-pin ZIF socket)
- A power supply unit
- A USB connection cable

The SUPERPRO package also includes a software installation CD, this User's Guide and a registration form.

Optional socket adapters to accommodate PLCC, TSOP, SOIC, SOP, QFP, TSSOP and BGA package types are available.

Software Features

The USB 2.0 Series supports:

- Windows XP and Windows Vista
- Over 25,000 (40,000+ by end of 2008) types of devices, including PROMs, E/EPROMs, PLDs and MCU from more than 150 manufacturers
- Many types of formats such as Binary, Intel (linear & segmented) Hex, Motorola S, Tektronix (linear & segmented), JEDEC, POF, etc.

SUPERPRO also includes a device insertion test (48 pins or less) to detect defective chips, improperly inserted devices and pins of poor contact (model dependent).

SUPERPRO includes an integrated full screen buffer editing environment with commands such as fill, copy, move, swap, etc. It automatically generates electronic serial numbers. Authorized users can use a user-defined mode.

System Requirements

The minimum system configuration is as follows:

- A desktop or laptop computer with Intel Pentium or an equivalent processor
- Windows XP or Windows Vista operating system
- At least 1 GB free space on the hard drive
- At least one standard (USB 2.0) compliant connector
- A CD drive

Chapter 2 System Installation

This chapter provides information on how to install the software and connect the programmer hardware properly.

To avoid complications during the installation process, you must install the software before connecting the programming hardware to your computer. Installing the software sets up the INF files and drivers automatically instead of going through the Windows New Hardware Wizard. This might install the wrong drivers with a compatible ID.

Software Setup

Select the software to setup your SUPERPRO programmer either from the CD-ROM or by downloading the program from the Xeltek website. Instructions for both methods are described below.

CD-ROM

1. Insert the CD into the CD-ROM drive.
2. If the setup program does not start automatically, run SETUP. EXE located in the root directory.

The system displays a dialog box to select the programmer model, illustrated below.

3. Select the appropriate programmer model. Make sure to select the model you purchased, since each model has its own software.
4. Select **Setup** to install the application software for the programmer.

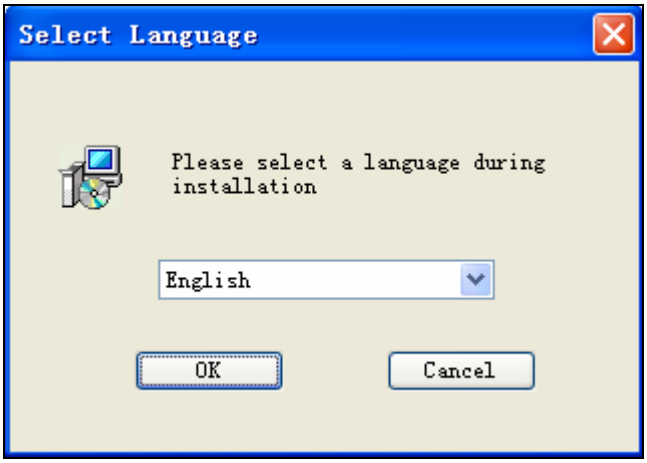
Xeltek Website

You can download the specific software for a certain model at Xeltek website: http://www.xeltek.com. Select the icon to download the appropriate file for your programmer. Once you have saved the file to your computer, run it to setup the software.

Setup Process

The following procedure explains the setup process.

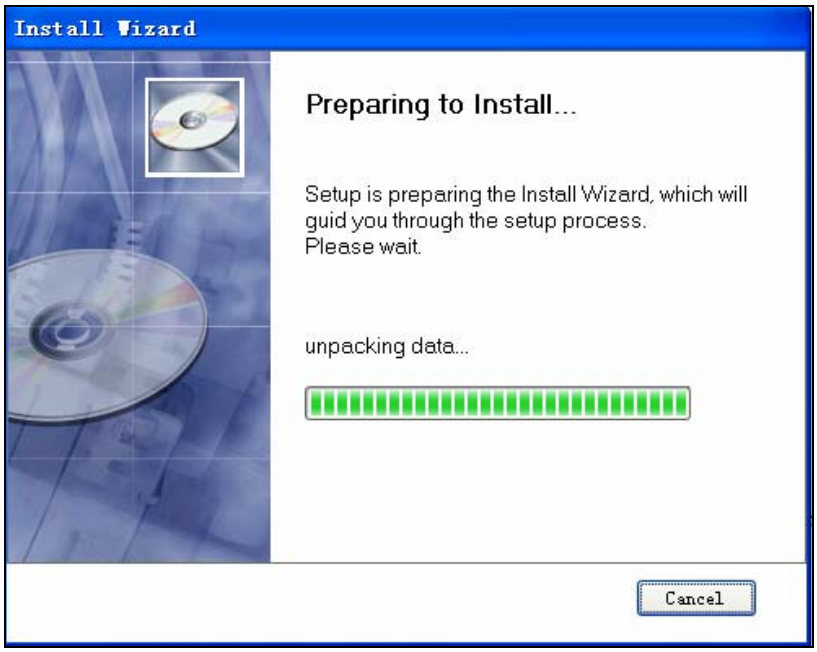
1. The system displays the **Select Language** dialog box, illustrated below.



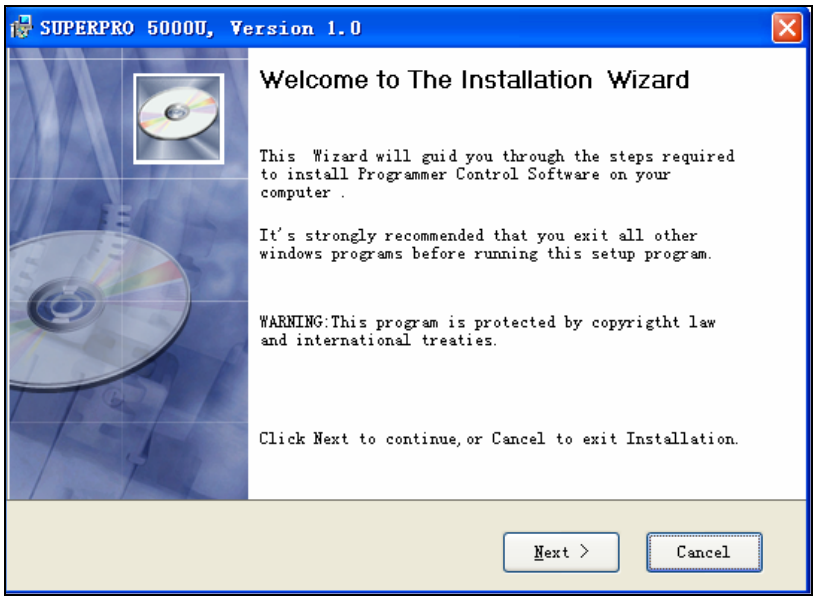
   Select your preferred language for the setup process from the drop down list.

2. Select **OK.**

The system displays the **Install Wizard** dialog box, illustrated below.

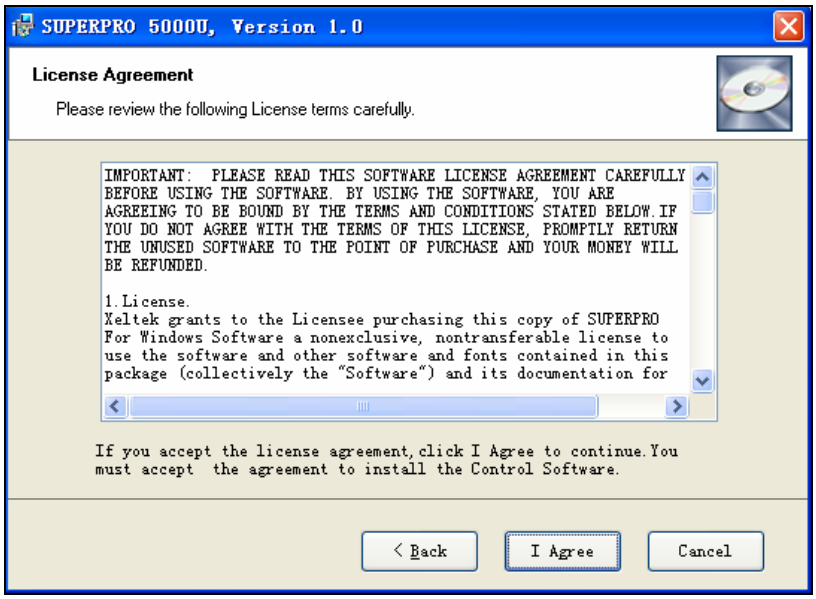


Next, the system displays the following dialog box.



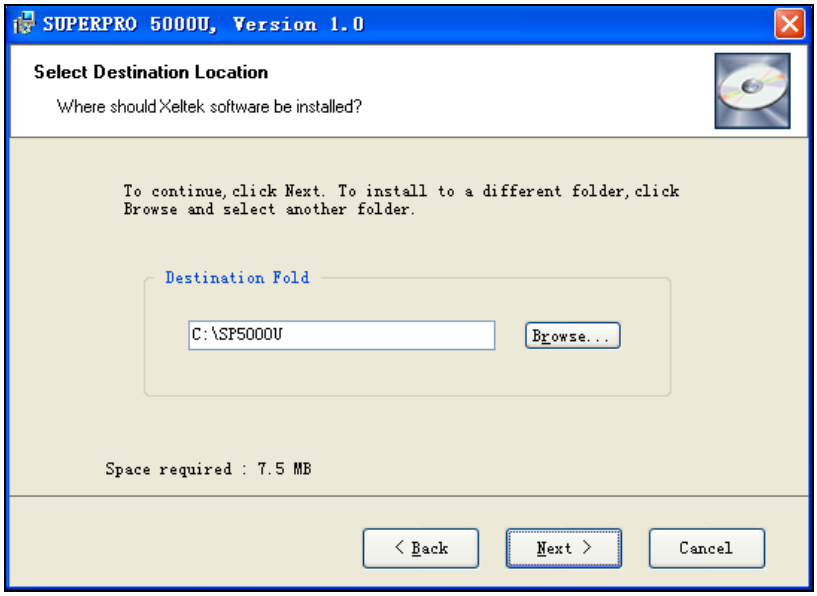
3. Select **Next** to proceed to the next step.

The system displays the License Agreement, illustrated below.



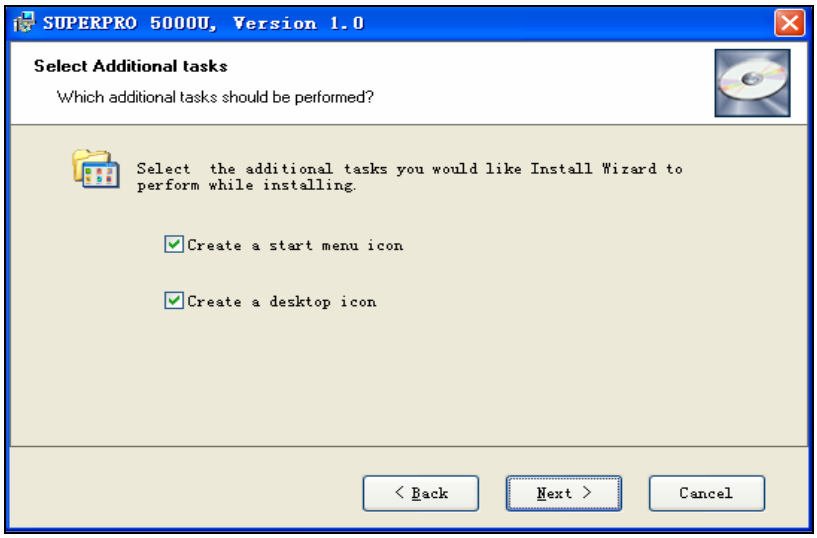
4. Please read the license agreement carefully. Select **I Agree** to continue the installation. To cancel the installation and exit the installation process, select **Cancel.**

The system asks for the **Destination Location**, illustrated below.



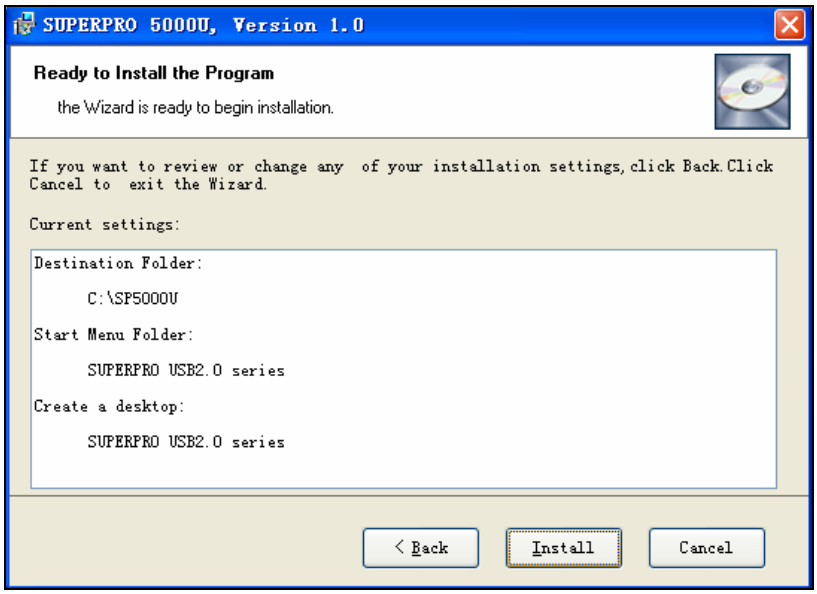
5. Select **Browse** to choose the installation path. After selecting the desired destination folder, select **Next**.

The system displays the **Additional tasks** dialog box, illustrated below.



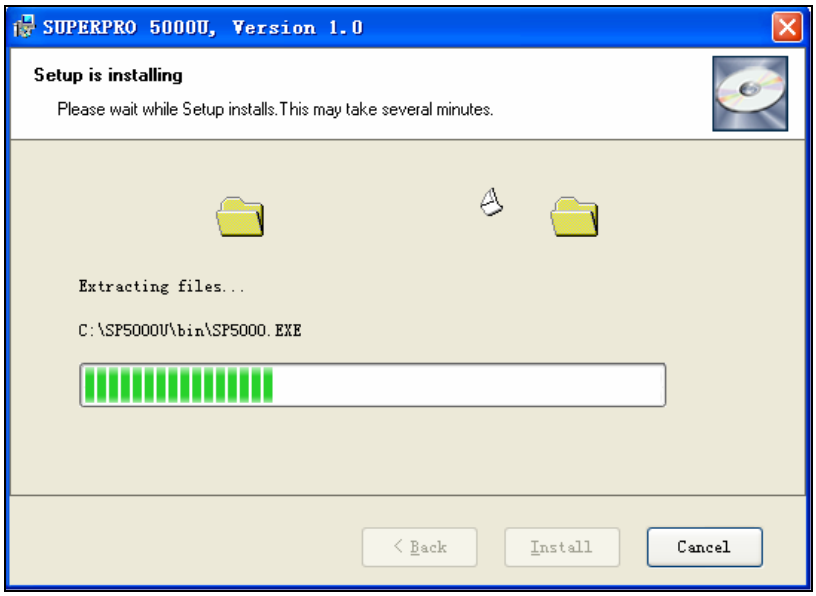
6. Check the **Create a start menu icon** box to add an icon on the Start menu during the installation. Check the **Create a desktop icon** box to add an icon to the desktop during the installation. Select **Next**.

The system displays the following dialog box.



7. Review the settings you selected. Select **Install** to proceed to the next step. Select **Back** to return to a previous step to change your selections.

The system displays the following dialog box.

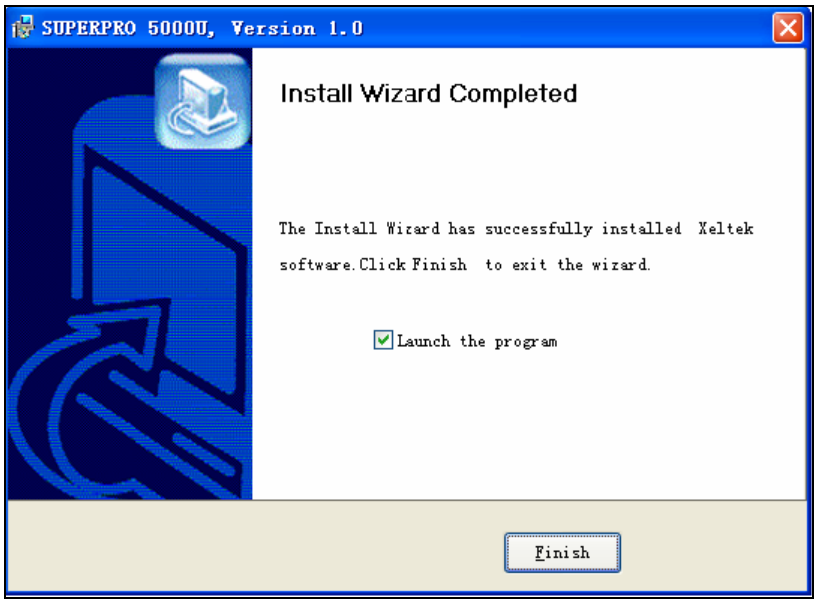


The system copies the files, installs the drivers and registers the software.

If you are using Windows Vista, the system displays following dialog box.

8. Select **Install this driver software anyway**.

The system displays the following dialog box.



9. Select **Finish** to complete the setup process.

Hardware and Driver Installation

If you connect the hardware before installing the software, the system opens the **Found New Hardware Wizard**, illustrated below.

1. Select **Cancel** to make sure the system does not install the wrong driver.
2. Return to the ***Software Setup*** process to setup the software before installing the hardware. Refer to page 10 for instructions.

After you set up the software, you can install the hardware and driver. The following procedure explains the installation process.

1. Make sure all other programs are closed during the installation process.
2. Connect the programmer module to the computer through the USB port.
3. Turn on the power switch of the programmer module.

When you connect the new hardware, the system initiates the **New Hardware Wizard**, which locates the driver from the software you have just installed. The system displays alerts one at a time, illustrated below. The alerts on your system may be slightly different.

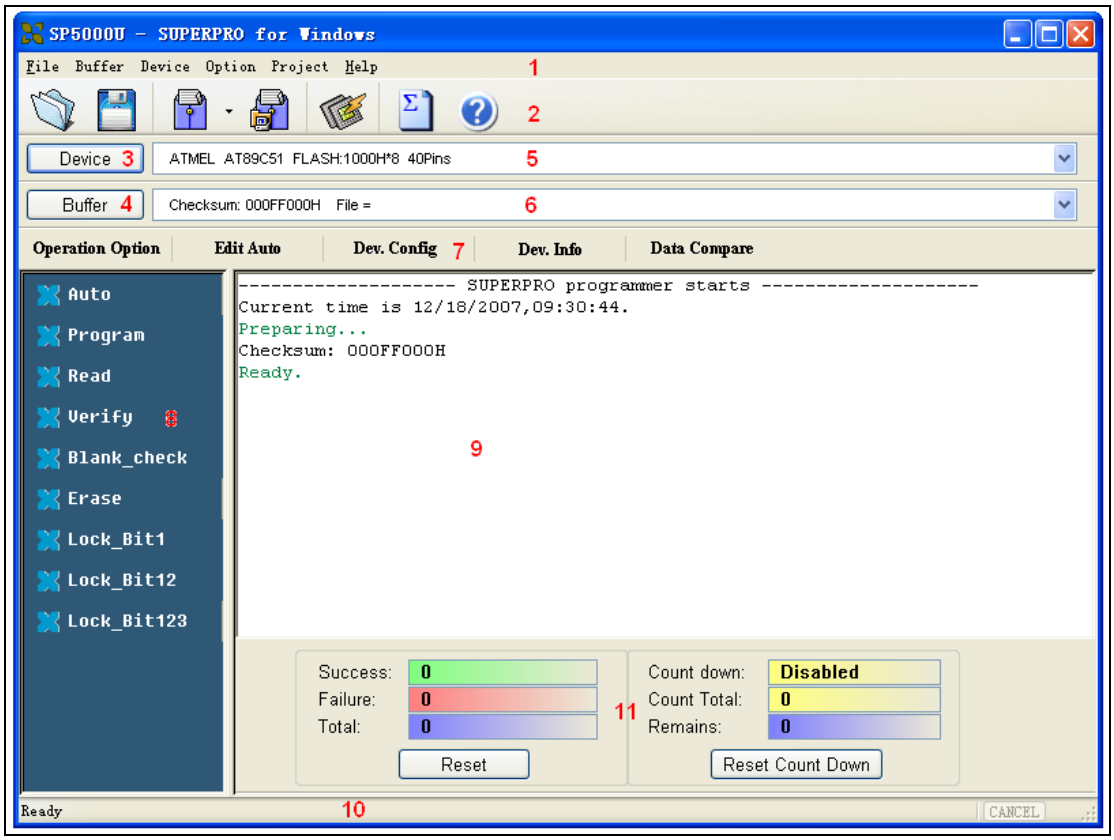
4. After you open the application software for the programmer, the system automatically starts the initialization. If it does not, make sure the programmer is securely connected to the computer and that the power switch is on.

Chapter 3 SUPERPRO Software Features

This chapter gives detailed explanations of the software functions of the programmer.

Main Screen

The main SUPERPRO screen is illustrated below. Each part of the user interface is labeled with a number and described below.



1. The Main Menu provides access to the File, Buffer, Device, Option, Project, and Help menus. Refer to the Main Menu section beginning on page 19 for more information.
2. The Toolbar includes icons for quick access to common functions. Refer to Tool Bar section on page 44 for more information.
3. Select **Device** to open the Device Selection dialog box. Refer to Select Device on page 30 for more information.
4. Select **Buffer** to open the Buffer Edit window. Refer to Buffer Menu on page 23 for more information.
5. The system displays information about the selected device. Refer to Main Screen Device Information Display on page 46 for more information.
6. The system displays information about the file currently loaded in the buffer.

Refer to Main Screen File Information on page 46 for more information.

7. The Operation Option toolbar facilitates setting the operation options properly. The items correspond to the items in the Option Menu. Refer to the Option Menu section beginning on page 35 for more information.
8. The left side of the screen lists device operation functions. Refer to the Device Operation Window section on page 49 for more information.
9. The Operation Message Window displays the operation process, the operation result and the historical records of information. Refer to the Operation Information Window section on page 52 for more information.
10. The Status bar displays the menu item info, the progress of operation, and a Cancel button. Refer to the Status Bar section on page 52 for more information.
11. The Statistical Window displays statistical information. Refer to the section on page 52 for more information.

Main Menu

File Menu

The **File** menu provides access to the **Load**, **Save**, **Recent Projects**, and **Exit** functions. Each function is described in the following sections.

Load File

There may be one or more data buffers in the device. If several data buffers are displayed after you have selected the device, refer to the name and the data manual for the meanings of the buffers.

The two data types of the device are **Data (HEX/ASCII)** and **Fuse**.

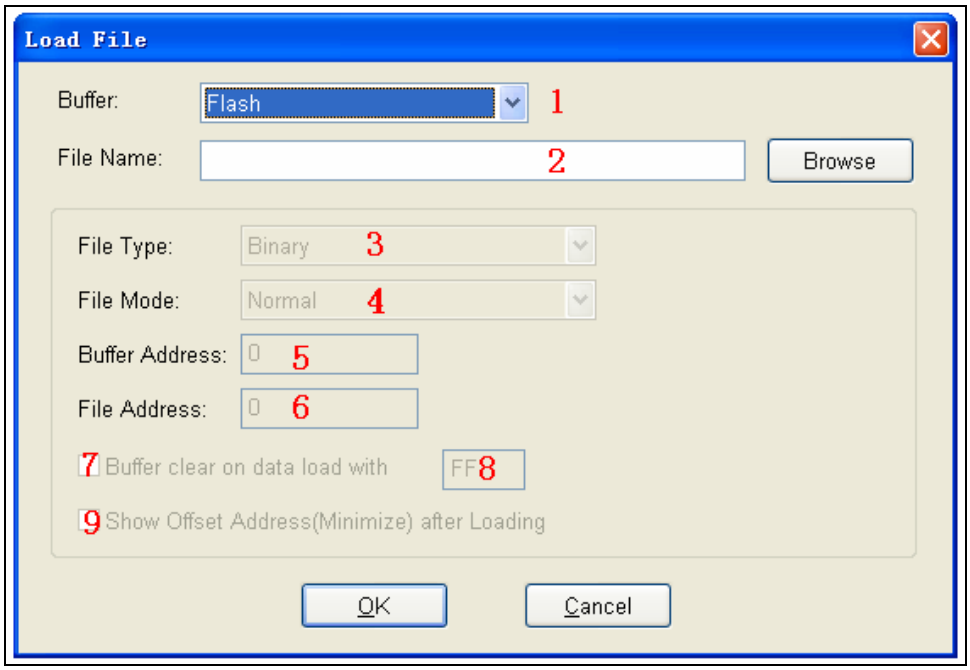
- For most EPROM and SCM, the data type is **Data (HEX/ASCII)**.
- The data type for PLD device is **Fuse**.

After you select the device, the software automatically identifies the data buffer type. You can review the buffer data types if you open the **Edit** dialog box for the data buffer. Refer to the **Buffer Menu** on page 23 for more information.

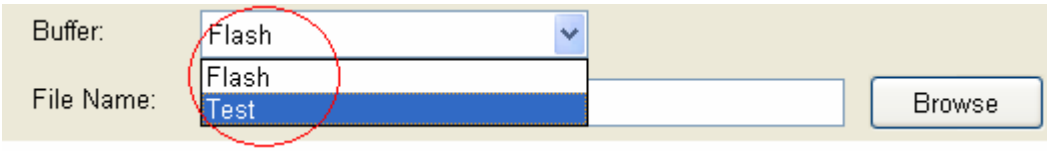
When some file data are loaded into the data buffer, the following rules apply:

- With a HEX/ASCII data buffer (EPROM, MCU etc.), the system assumes that 8 bits of the data are valid.
- With a JEDEC buffer (PLD/PAL), the system considers the lowest bit (1 bit) of the file data valid.

Select **Load** from the **File** menu to open the **Load File** dialog box, illustrated below. The red labels refer to the numbered explanations below.



1. Select the data **Buffer** name from the drop down list, as illustrated below.

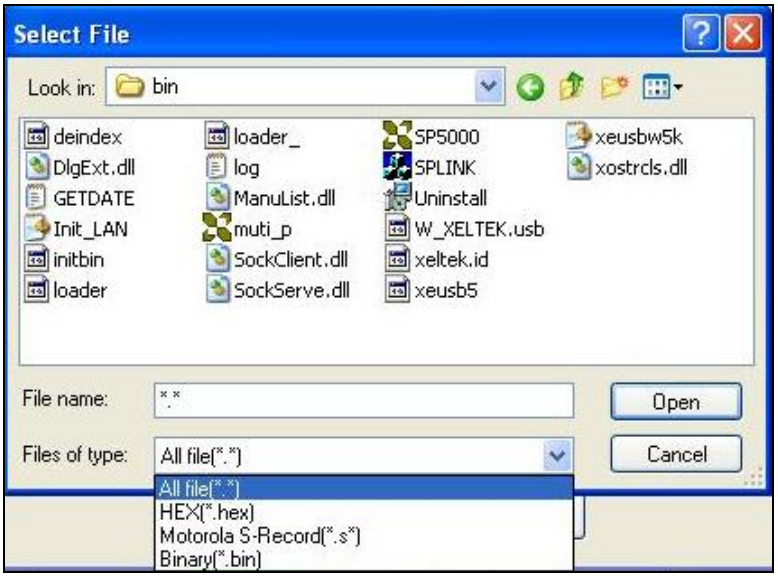


- The default option is considered the programming zone of the device as described in the data manual.
- To choose between several buffers, determine the corresponding programming zones according to the names and the data manual.

The system loads the file data into the data buffers according to the data buffer names.

2. Enter the name of the data file to load in the **File Name** field.

   Alternatively, you can select **Browse** and choose the file using the **Select File** box, which is illustrated below.



   First select the file type from the **Files of type** drop down list.

   - Data (HEX/ASCII data) file types can be further divided into many different types according to their storage forms. You must select the appropriate file type to ensure the data loads correctly.
   - The file type for Fuse data files is JEDEC and the file extension is "jed." After selecting the file, you do not need to select the data file type.

Altera Corporation uses POF format files. To program an Altera Corporation chip, refer to POF Format Files on page 63.

3. Select the **File Type** from the drop down list.
4. Select one of the following **File Modes** from the drop down list:
   - **Norm**al to load the whole file
   - **Even** to load the first byte and discard the second byte out of every two bytes
   - **Odd** to discard the first byte and load the second byte out of every two bytes
   - **1st byte of 4** to load first byte and discard the other three bytes out of every four bytes
   - **2nd byte of 4** to load the second byte and discard the other three bytes out of every four bytes
   - **3rd byte of 4** to load the third byte and discard the other three bytes out of every four bytes

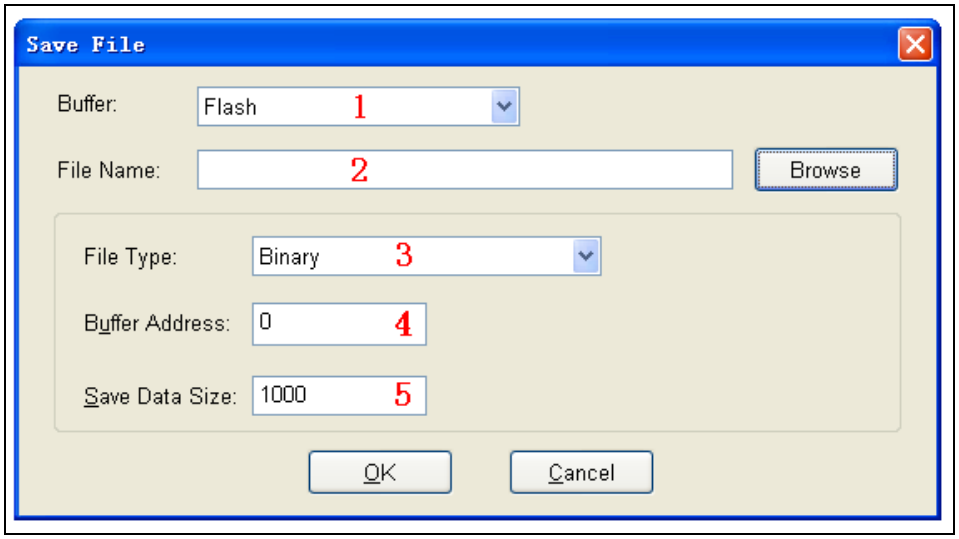
- **4th byte of 4** to load the fourth byte and discard the other three bytes out of every four bytes
- **1st 2_byte of 4** to load the first two bytes and discard the other two bytes out of every four bytes
- **2nd 2_byte of 4** to load the last two bytes and discard the other two bytes out of every four bytes

Refer to Separate File Data for Programming on page 63 for more information.

5. The **Buffer Address** field displays the start address of the data loaded into the buffer.
6. The system displays the minimum offset address in the **File Address** field. If the file type has a non-zero start address, enter the data offset address in the **File Address** field. Make sure to enter the correct address, because an incorrect file offset address causes part of the buffer to be filled with FF (or incorrect data). This causes a system overflow or failure. Refer to Solve Invalid File Type or File Data Overflow on page 63 for more information on this problem.
7. Select the **Buffer clear on data load with** checkbox to fill the data buffers with the specified data.
8. Enter the data input type.
9. Check the **Show Offset Address (Minimize) After Loading** to show the minimized file offset address or initial address on the screen.
10. Select **OK** to load the file. Otherwise, select **Cancel**.

Save

Select **Save** from the **File** menu to save data in the current buffer to a disk file. The system opens the **Save File** dialog box, illustrated below. The red labels refer to the numbered explanations below.



1. Select the Data Buffer Name from the **Buffer** drop down list.
2. Enter **File Name** for Data or select **Browse** to select a file name.
3. Select the type of file from the **File Type** drop down list.
4. Enter the start address of the data buffer in the **Buffer Address** field. The saved data starts at the start address.
5. Enter the save data size in number of bytes in the **Save Data Size** field.

Recent Projects

The **File** menu lists the full file names, including the full path, of any recent projects. Refer to the **Project** menu for a definition of the project.

Exit

Select **Exit** to close the programmer software.

Buffer Menu

The process of programming a chip is the process in which the system writes buffer data into the chip storage units according to the manufacturer's requirements. The **Buffer** menu manages data in the buffer. Each option is described in the following sections.

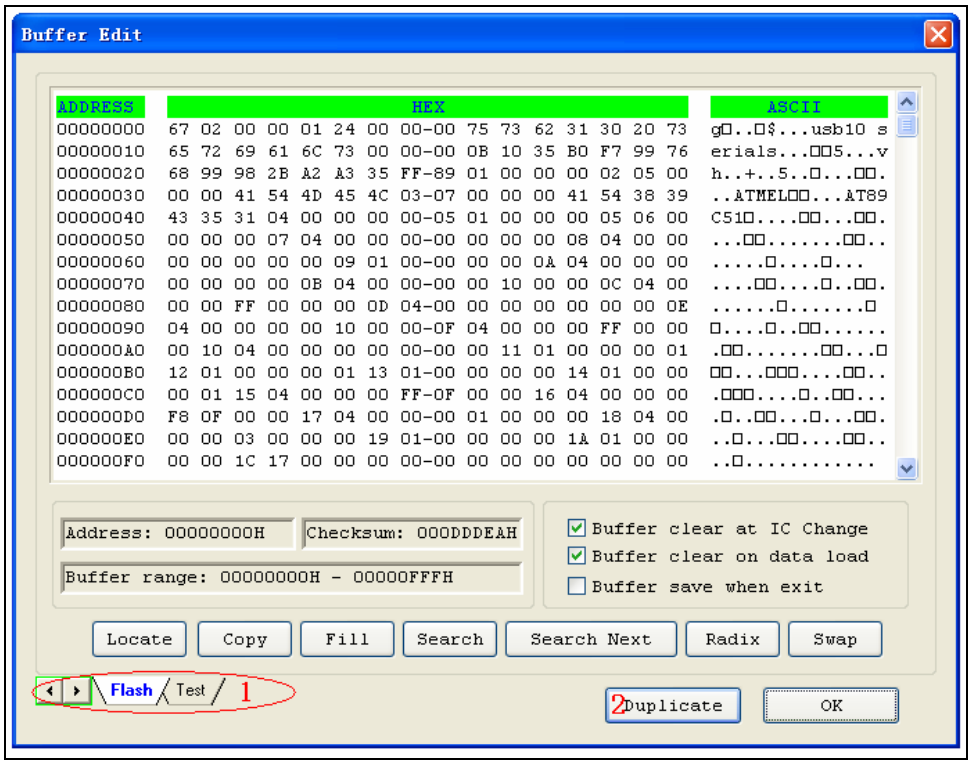
Edit

Select **Edit** from the **Buffer** menu to open either the Fuse Buffer Edit window or the Data Buffer (HEX/ASCII) Edit window.

Use the following keys to edit the buffer data:

- Press the **Page Up** key to Page up in the data window.
- Press the **Page Down** key to Page down in the data window.
- Press the **Home** key to move the cursor to the beginning of the line.
- Press the **End** key to move cursor to the end of the line.

The following sample illustrates an example with two data buffers. The red labels refer to the numbered explanations below.



Note: You can open several data buffers at the same time from the toolbar.

1. The tabs in the lower left corner allow you to switch from one data buffer to another. The current data buffer name is highlighted.
2. Select **Duplicate** to save the data in the buffers in text form. The following examples show the text form of the saved buffer data.

   - HEX/ASCII data buffer:

     00000000 4D 5A 90 00 03 00 00 00-04 00 00 00 FF FF 00 00 MZ..._..._.....

     00000010 B8 00 00 00 00 00 00 00-40 00 00 00 00 00 00 00 .....@.....

     00000020 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....

     00000030 00 00 00 00 00 00 00 00-00 00 00 00 80 00 00 00 .....

   - Fuse data buffer:

     00000768 1001101010101010101010101010101010

     00000800 0110101010101010101010101010101010

     00000832 1010011010101010101010101010101010

     00000864 1010101001101010101010101010101010

     00000896 1010101010100110101010101010101010

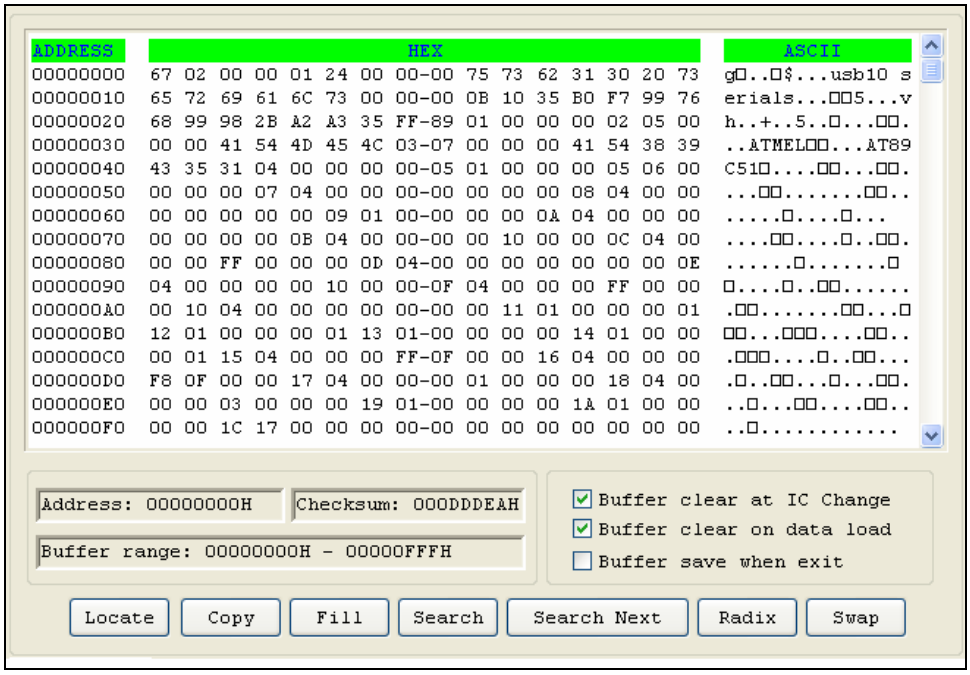
     00000928 1010101010101010011010101010101010

     00000960 1010101010101010101001101010101010

     00000992 1010101010101010101010101001101010

HEX/ASCII Data Buffer

When you select **Edit** from the **Buffer** menu with the appropriate file type, the system opens the Data Buffer (HEX/ASCII) Edit window, illustrated below.

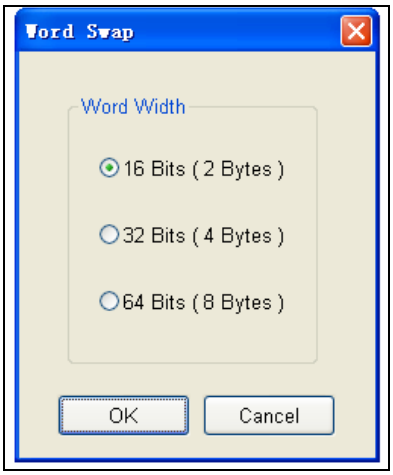


The HEX/ASCII data buffer is 8-bit wide. Press the **TAB** key to switch between HEX and ASCII data for editing.

Note: Remember that the value of start address must be less than that of the end address.

1. Select **Buffer clear at IC change** to clear the buffer after selecting the device.
2. Select **Buffer clear on data load** to clear the buffer before loading the file data.
3. Select **Buffer save when exit** to save the buffer data when exiting. This option also loads the saved data automatically the next time you enter the system.
4. Select **Locate** to open the Locate Buffer dialog box. Enter the address of the data you wish to display and select **OK** to quickly move the cursor to the desired location.
5. Select **Copy** to open the Copy Buffer dialog box.
   - Enter the start address in the **Start Address** field.
   - Enter the end address in the **End Address** field.
   - Select **OK** to copy the data between the start address and the end address, beginning with a new address. Otherwise, select **Cancel**.

6. Select **Fill** to open the Fill Data Into Buffer dialog box.
   - Enter the desired data in the **Fill Data** field. The data consists of HEX numbers, such as AA, 55, E4 for the Data Buffer (HEX/ASCII data) Edit window.
   - Enter the start address in the **Start Address** field.
   - Enter the end address in the **End Address** field.
   - Select **OK** to fill the data. Otherwise, select **Cancel**.
7. Select **Search** to search for a specified string.
8. Select **Search Next** to begin the next search for the specified string.
9. Select **Radix** to toggle between the HEX and DEC memory address display.
10. Select **Swap** to open the Word Swap dialog box, illustrated below.



Select the **Word Width** option to swap high byte and low byte according to the specified width in the address range and select **OK**. Otherwise, select **Cancel**.

For example, if the data buffer address 0-10(HEX) is:

12 34 56 78 90 AA BB CC - DD EE FF 11 22 33 44 55

Then after swapping the data, it would look like this according to the selected word width:

- 16 Bits (2 bytes) -

  34 12 78 56 AA 90 CC BB - EE DD 11 FF 33 22 55 44

- **32 Bits (4 bytes)** -

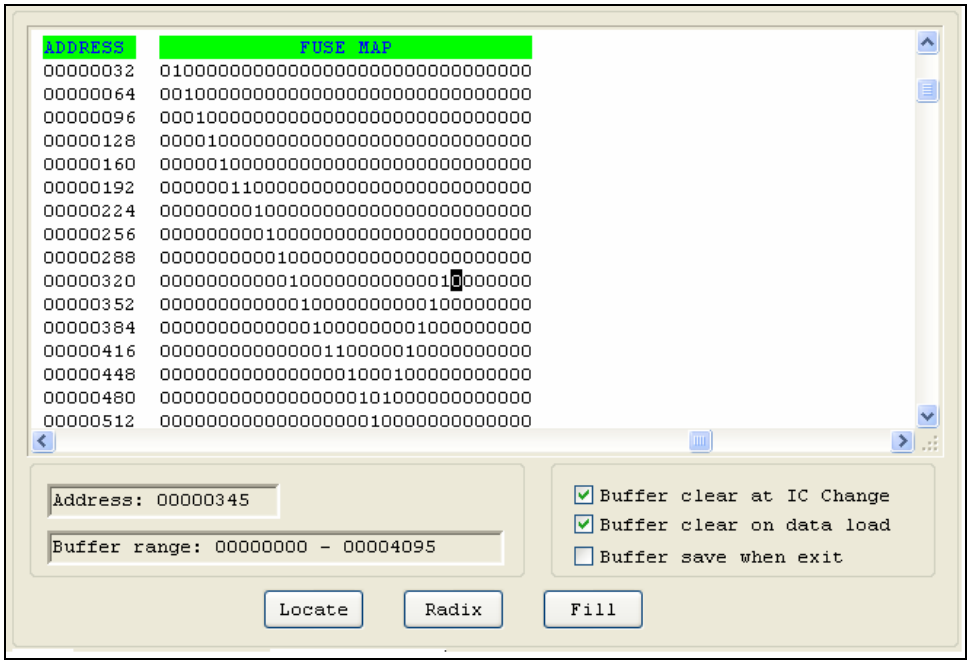
  78 56 34 12 CC BB AA 90 - 11 FF EE DD 55 44 33 22

- **64 Bits (8 bytes)** -

  CC BB AA 90 78 56 34 12 - 55 44 33 22 11 FF EE DD

Fuse Buffer

When you select **Edit** from the **Buffer** menu with the appropriate file type, the system opens the Fuse Buffer Edit window, illustrated below.



The data for editing is either 0 or 1. Refer to the data manual of the device and the JEDEC to determine which of the following explanations for the data apply.

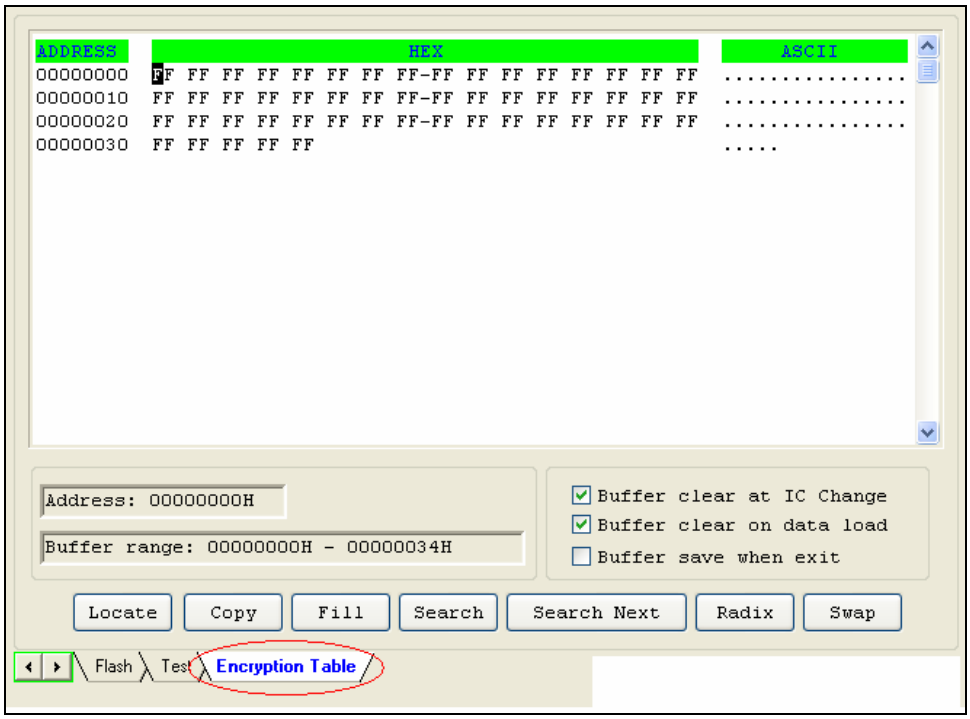
- 1 represents an intact fuse and 0 represents a blown fuse
- 1 represents a blown fuse and 0 represents an intact fuse

Refer to the HEX/ASCII Data Buffer section on page 25 for more information about the options on this screen.

Encryption table

A data buffer corresponding to the encrypted data is called an encryption table. Whether an encryption table exists depends on the data manual.

Select **Encryption Table** from the **Buffer** menu to open this special data area, illustrated below.



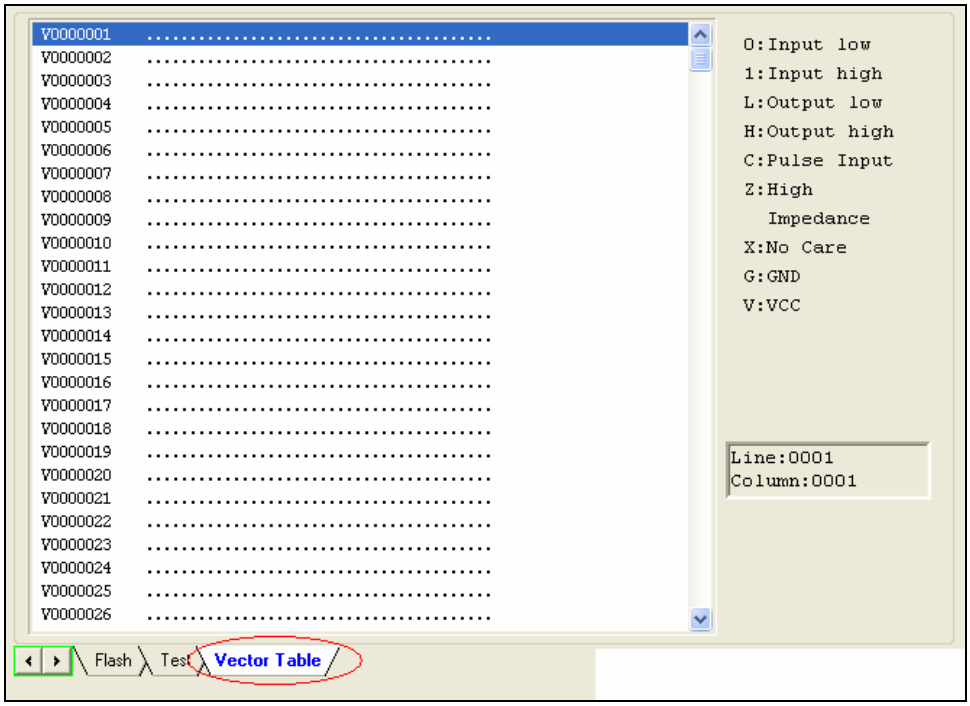
The Encryption function on the Function Items for Operations on Devices screen writes the encryption table data into a chip. In most cases, you cannot check the chip after the data is written into the encryption table. The encryption table data also cannot be read.

Test Vector Table

The test vector table is a special data area. The data buffer corresponding to the data is called the test vector table.

If a test vector table is included in the JEDEC file, the software loads the test vector table into the buffer when you load the file. Each line displays one vector. Each bit represents one pin's test characteristic. The bits from left to right correspond to the pins from the first to the last respectively.

A sample vector table is illustrated below, with definitions following.



- **Z:** High impedance state
- **X:** No care state
- **N:** VCC or GND (non-test input pins)
- **H:** Output Logic High (VOH)
- **L:** Output Logic Low (VOL)
- **C:** Clock pin
- **1:** Input Logic High (VIH)
- **0:** Input Logic Low (VIL)

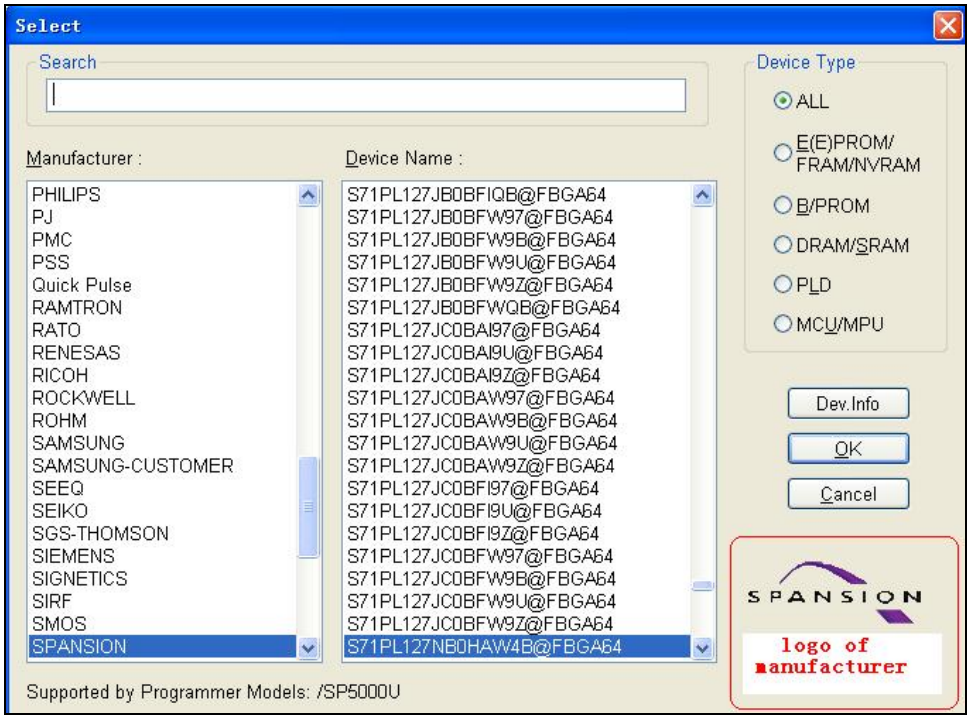
Device Menu

The Device menu allows you to select the device.

Important:	Select the device before any other operation to ensure the programmer selects the proper algorithm for the name and manufacturer of the device.

Select Device

The **Device** menu opens the Device Selection dialog box, illustrated below.



To select a device, complete the following steps.

1. Select one of the following **Device Type** options. If you are not sure which option to choose, select **All.**
   - E/EPROM (including EPROM, EEPROM, FLASH)
   - PLD
   - B/PROM
   - DRAM/SRAM
   - MCU
2. Select the manufacturer in the **Manufacturer** column. The dialogue box displays the manufacturer icon in the lower right corner.
3. Select the device name from the **Device Name** column. You can also double-click the device name to confirm your selection.

NOTE: The chip's printed name may include the parameters (i.e., speed, temperature, encapsulation, etc.). If the name is different than what appears in the **Device Name** column, you may ignore the difference or ask for help.

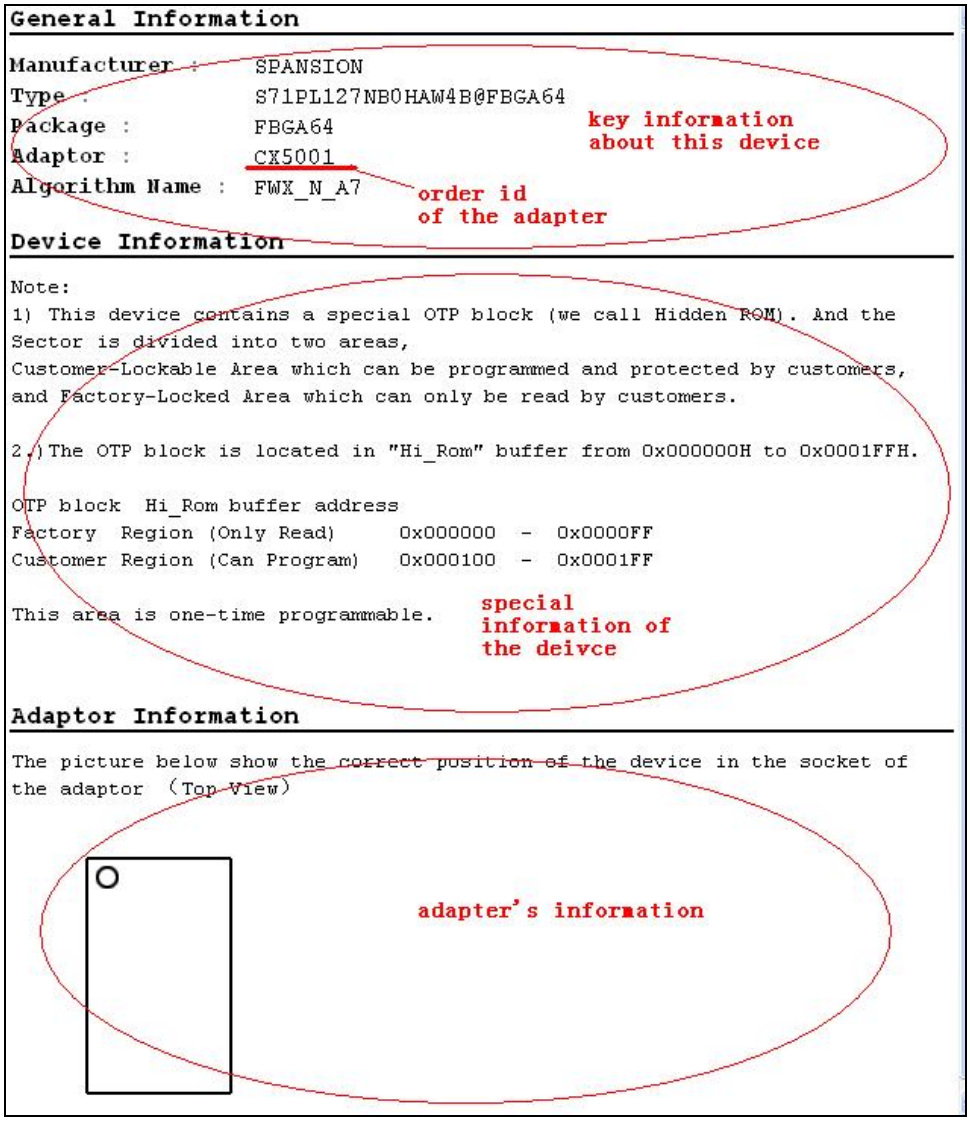
You may use the **Search** box to help select the device from the large quantity of devices listed. Enter the first few characters of the device name to limit the number of choices listed in the **Manufacturer** and **Device Name** columns. For example, if you enter **89** in

the **Search** box, the list only shows Manufacturers and Devices that contain '89'. The sequence of the characters is important. For example, if you enter '89', the list includes '819', and 'DA8S9' but not '98'.

4. Select **OK.**

Dev. Information

Select **Dev Information** from the **Device** menu to open the Device Information. A sample is illustrated below.



The device information includes the following information:

- Key information about the devices
- The manufacturers' requirements
- The conventions on the special devices

  Note: Some devices may have special requirements, including the arrangement of the data buffer content, and the explanations on the special operations of the device. Read the information carefully.

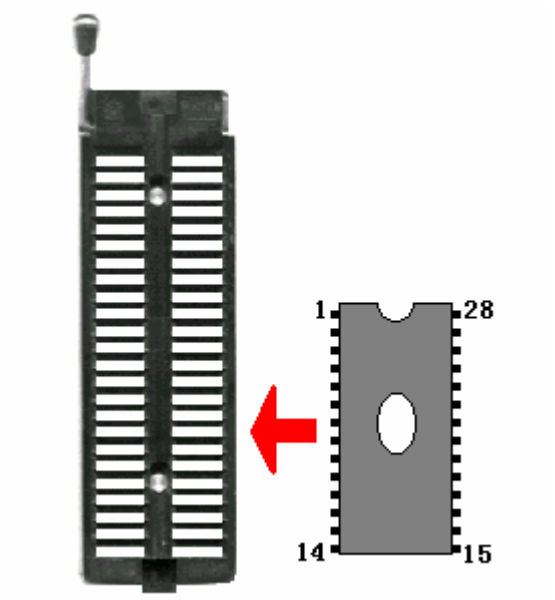
  To ensure correct programming, arrange the data and prepare the corresponding settings according to the requirements. You can also find this information in the data manual of the device.

- The adapter information, which may include the order number (such as CX5001), the size of the chip, and the chip insertion direction.

  Note: To program chips with other than the standard encapsulation, you must purchase the appropriate adapters.

Chip Insertion – DIP Encapsulation

Insert the chip with the notch upwards, aligning the bottom line with the bottom line of the socket. The diagram below illustrates the process of inserting a 28-pin chip.



Chip Insertion – None-DIP Encapsulation

Socket adapters are available for other encapsulations, including PLCC, QFP, and SOIC. Refer to the Adapter Information on the Device Information screen. The insertion follows the position of pin 1.

To identify pin 1 of the chip, look for a solid square or dot on the chip that gives the information about the chip. It also indicates the first pin of the chip.

Note: Generally, the pins of the chip are downward when they are inserted. However, as for the chip PLCC20, its pins should be upward.. This insertion is called the "Dead Bug" method.

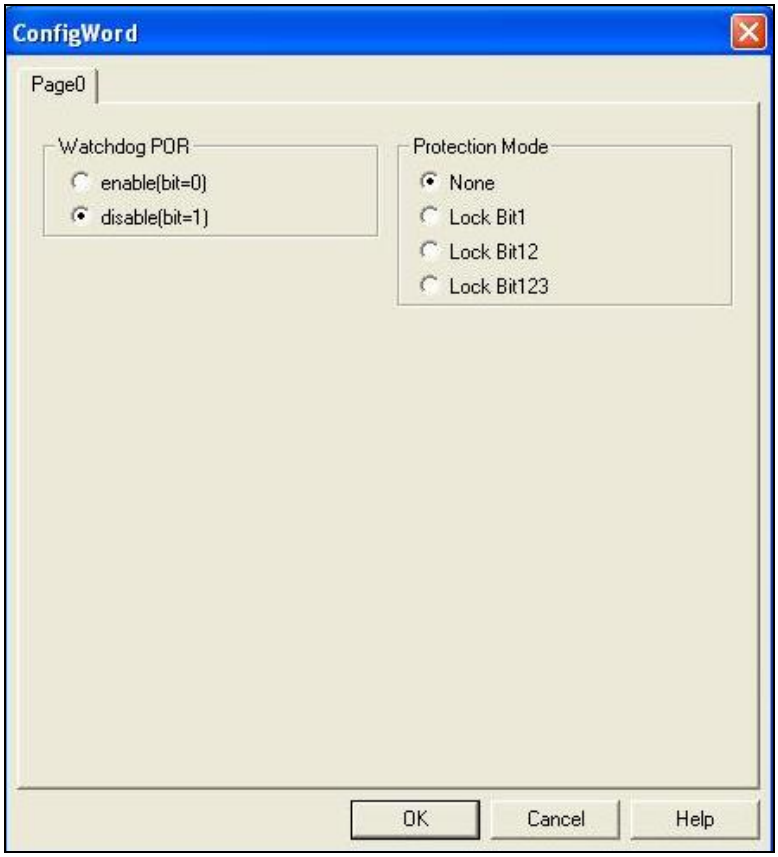
Device Configuration Word

Important: The device configuration word varies from device to device. Refer to the device data manual for more information about the configuration words. Set the configuration words according to the requirements of your target system. Otherwise, you cannot use the device normally even if it is normal when preparing and verifying the program.

Some single-chip microcomputers allow special working modes, such as the storage area mapping, the watchdog time, the clock, or the encryption. Set these special modes through the Device Configuration Word option.

The user files contain the configuration words of some devices. When you load these files, the system automatically loads the configuration words into the configuration word buffer. However, you must select the configuration words for some devices manually.

Select **Device Configuration Word** from the **Device** menu to open the ConfigWord dialog box. The following sample illustration shows the configuration word for DALLAS DS89C420. It contains the watchdog and three protection modes.



Some single-chip microcomputers include many items for the configuration word. The items may appear on separate tabs or pages. Make sure to set all the items on all pages before programming a chip.

The input of the configuration word can be divided into several ways:

- The 8-bit (byte) edit box input
- The 16-bit (word) edit box input
- The single choice input (select and deselect, choose one out of two)
- The multiple choice input (choose one out of many)

The configuration words for chip programming are classified into two categories:

- If the configuration word contains some protected items, then you cannot verify the items after preparing the program. When preparing the program, the programmer automatically shields the protected items and writes the other items. Select **Security or Protection** to write the protected items.

  Note: If you select **None** as the Protection option, the system does NOT carry out an operation when performing Security or Protection.

- In some devices, the configuration words cannot be separated, so the system does not write the configuration word when preparing the program. To write the configuration word in these cases, select the corresponding item, such as **Write_Option**.

Some FLASH devices also use the configuration word for the Protect operation and to execute and display the segment protection. Refer to the Device Operation Window section on page 49 for more information.

Option Menu

The Option menu includes items to help you program in a specific mode to meet the targeted system requirements. Before programming a chip, make sure to select the necessary settings.

Operation Option

Select **Operation Option** from the **Option** menu to open the Operation Option screen.

Note: Using the Operation Option to change the settings may produce minor changes from the following descriptions.

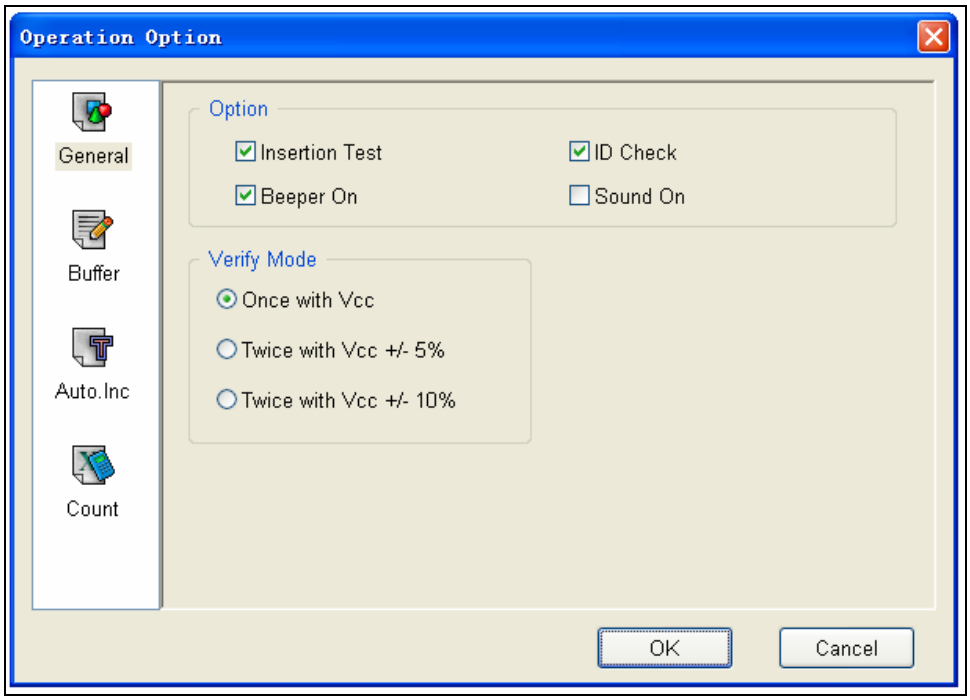
The Operation Option screen includes four views:

- General
- Buffer
- Auto.Inc
- Count

Each view is explained in the following sections.

General

The General view of the Operation Option screen is illustrated below.



1. For chips with 48 pins or less or for chips with an adaptor that reduces the chip to 48 pins or less, select **Insertion Test** to have the programmer check the pin contact status before programming. The test includes a check for poor contact, wrong direction, chip insertion, and chip insertion orientation errors. Refer to Insertion Test on page 68 for more information.
2. Select **ID Check** to have the programmer read the Electronic Identifier Code from the chip to identify the manufacturer, the device type and the programming algorithm code. If you have already selected the device, the default for this option is checked. Refer to Device ID Check on page 69 for more information.
3. Select **Beeper On** to turn the beeper on or off. The beeper makes a sound to indicate the results of operations such as Insertion Test error, ID unmatched, programming successfully completed or failed.

Note: Not all programmers include a beeper.

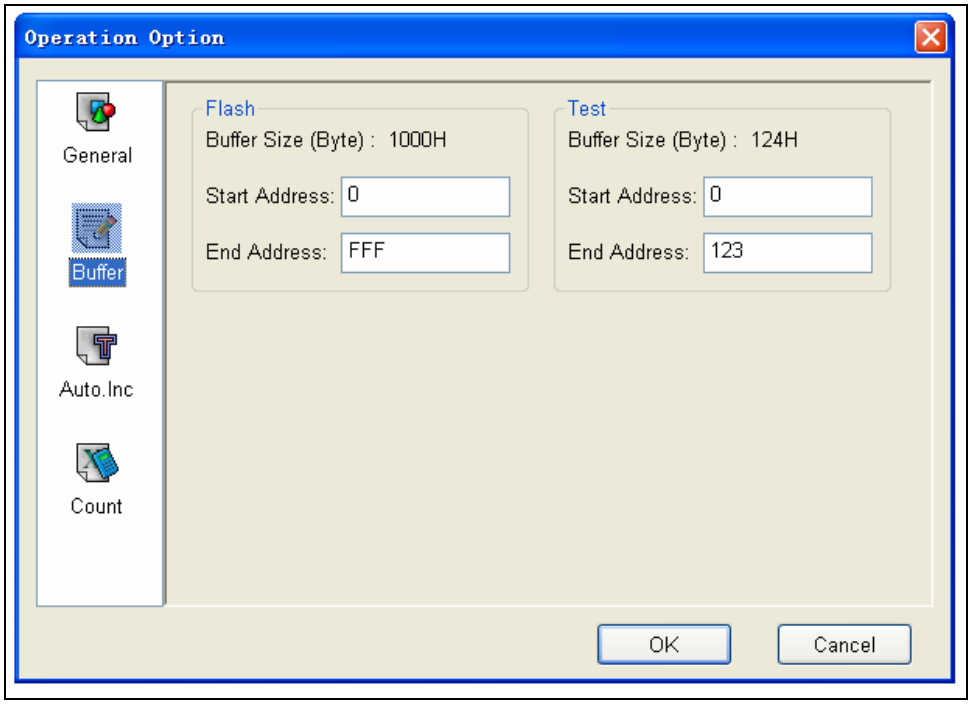
4. Indicate the mode for verifying the data by selecting one of the **Verify Mode** options. These options refer to the voltage of the pin VCC, which may vary when the programmer verifies the data. The manufacturer provides the MinVcc and MaxVcc values for verifying the data.

   VCC(+/- 5%) or VCC(+/- 10%) is adopted for the verification. If VCC = 5.00V, select one of the following options:

   - VCC = 5.00V and verify the data once
   - VCC = 4.75V and VCC = 5.25V, and verify the data twice (+/- 5%)
   - VCC=4.50V and VCC=5.50V, and verify the data twice (+/- 10%)

Buffer

The Buffer view of the Operation Option screen is illustrated below.



The default programming method is to program the device from the start address to the end address. However, you may choose to program only a part of the chip, such as with most E/EPROM (FLASH) devices.

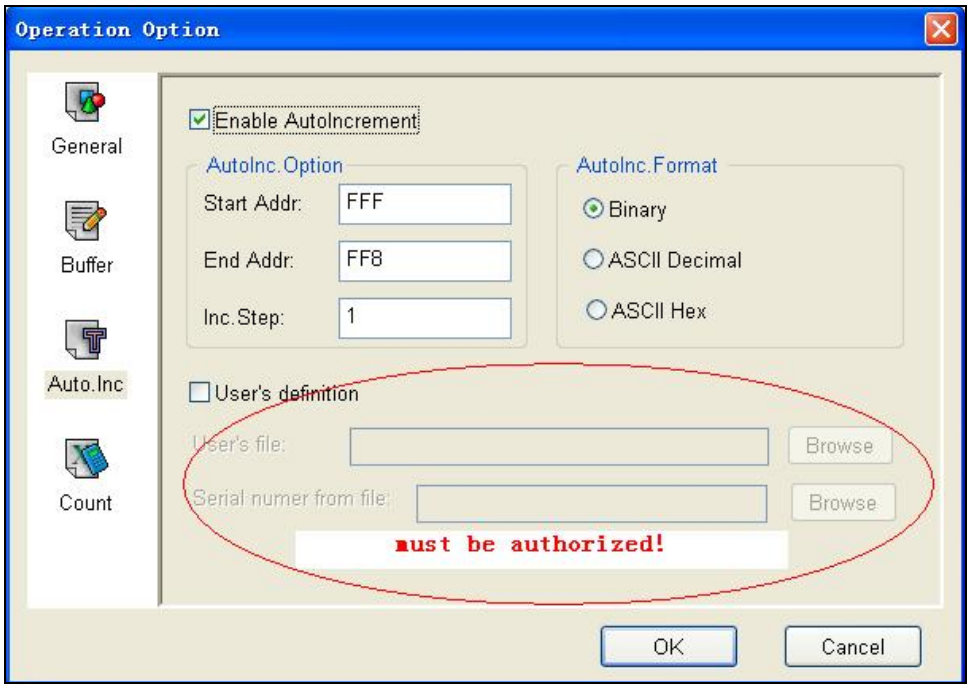
Note: Most single-chip microcomputers do not allow a change to the start address of the device.

1. For each file, enter the **Start Address**.
2. For each file, enter the **End Address**.

EXAMPLE:	The device **MACRONIX MX29F200B TSOP48** is a 16-bit flash device. Its programmable zone starts at the address 0 and ends at the address 1FFFF (hexadecimal system). The buffer size in bytes is (1FFFF+1) * 2 = 40000 (hexadecimal system). Because it is a 16-bit (2 bytes) device, the buffer size is twice as large as the device size. To program only the second part of the device, enter 10000 as the start address and leave the end address unchanged. This correlates the programming data to the buffer data between the address 10000*2 = 200000 and 40000.

Auto Increment

The Auto.Inc view of the Operation Option screen is illustrated below.



Auto Increment provides two methods of generating the serial numbers: Auto Increment and User Defined. This function is not available in stand-alone mode.

Note: The serial number is the content required by some applications that must be written in a certain area for every chip. This information includes the product sequence number and the MAC address.

To have the software generate the serial number, select **Enable AutoIncrement**.

1. Enter the automatic start buffer address in the **Start Addr** field.
2. Enter the automatic end buffer address in the **End Addr** field.

3. Enter **1** in the **Inc. Step** field.
4. The **AutoInc Format** defines the start value and the overflow value. The overflow value is the number at which the increment ceases and carries to the next address, which becomes the new start value.
   - Select **Binary** to set the start value as 0 and the overflow value as 256.
   - Select **ASCII Decimal** to set the start value as 30 (hexadecimal representation of 0) and the overflow value as 39+1 (the hexadecimal representation of 9).
   - Select **ASCII Hex** to set the start value as 30 (hexadecimal representation of 0) and the overflow value as 46+1 (the hexadecimal representation of a number greater than F, where 9+1=A).

Example: There are 64 chips with IDs from 0001 to 0064, which are in the end address of 89C51 EPROM. The selected mode is 4 bytes, using the decimal system.

1. Enter **FFF** in the **Start Address** field.
2. Enter **FFC** in the **End Address** field.
3. Enter **1** in the **Inc.Step** field.
4. Select **ASCII Decimal** as the **AutoInc. Format**.

In the Buffer Edit window, make the following change:

FFC: 30 FFD: 30 FFE: 30 FFF: 31

The Operation Information window displays the following information during programming:

Current AutoIncrement Data=0001

In subsequent programming operations, the information in the last 4 bytes of the buffer data is:

FFC: 30 FFD: 30 FFE: 30 FFF: 31

After successful programming, the AutoIncrement value increases by 1. The Operation Information window displays the following information:

Current AutoIncrement Data = 0002

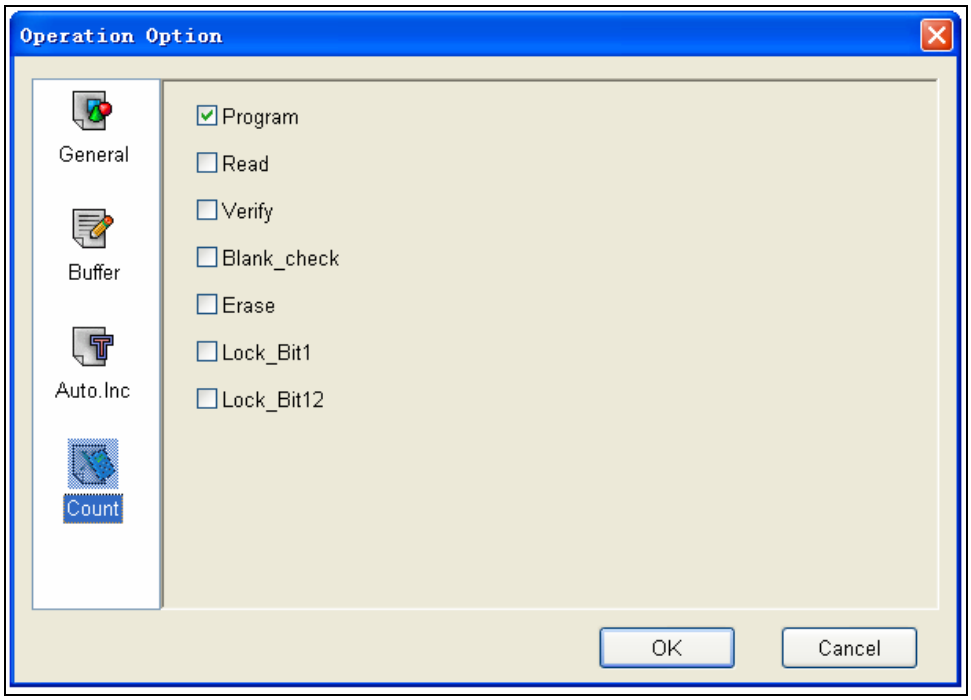
The last 4 bytes of the buffer data are:

FFC: 30 FFD: 30 FFE: 30 FFF: 32

To define the increase of the serial number, select **User's Definition**. You must be authorized to use this function. If so, you can change the data as desired in the 4K byte range.

Count

The Count view of the Operation Option screen is illustrated below.

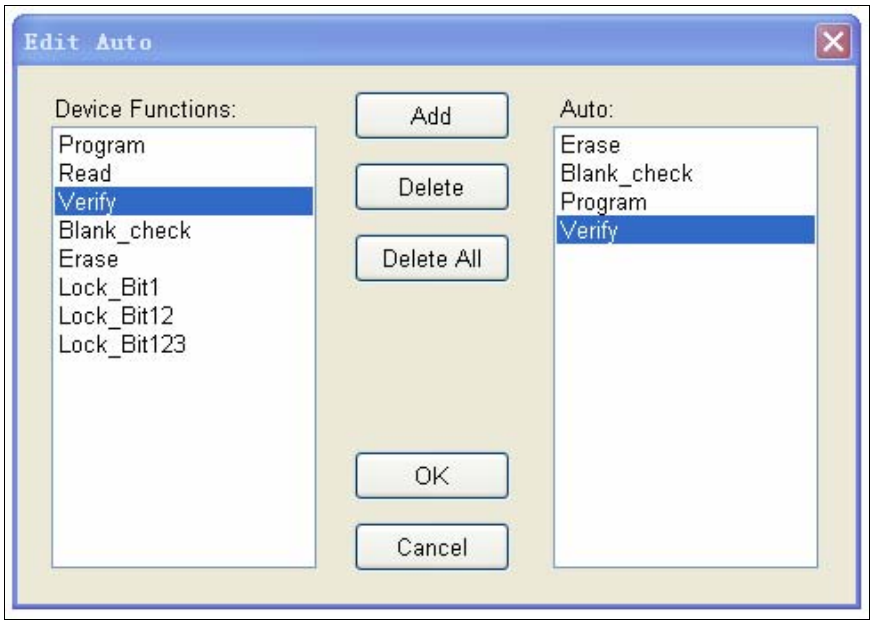


This screen allows you to change the default setting regarding the statistical work of the programming status. In the default case, it is only effective for the function "program".

Edit Auto

The Auto function organizes the different functions of the device into a sequential group and carries out the functions in order, similar to a batch command.

Select **Edit Auto** from the Option menu to open the Edit Auto screen, illustrated below. The illustration is a sample for the device ATMEL AT89C51.



The **Device Functions** column displays all operations for the device.

The **Auto** column lists the operations in order of execution. For the sample in the illustration, the operation execution order is Erase, Blank_check, Program, Verify, Lock_Bit123.

1. Select **Add** to add the highlighted function in the Device Function column to the end of the Auto column. You can also double-click the highlighted function in the Device Function column to add it to the Auto column.
2. Select **Delete** to delete the highlighted function in the Auto column. You can also double-click the highlighted function in the Auto column to delete it.
3. Select **Delete All** to remove all the functions from the Auto column.

Production Mode

Select **Production Mode** from the Option menu to program devices in large quantities. With the help of the Insertion Test, the programmer automatically detects the insertion and taking-out of the devices and carries out the whole programming process. In this mode, the user does not need to use his mouse or keyboard.

If you receive a non-support information message, the selected device does not support production mode.

Refer to the Execute Production Mode section on page 56 for more information.

Log File

You can create a log file to have the system save Operation Information window content into a log file when you exit the program. Select **Log File** from the Option menu to open the Log File dialog box, illustrated below. The red labels refer to the numbered explanations below.

1. Select **No** to delete the log file. The system will not create a log file on exit.
2. Select **New** to overwrite the existing log file every time the program starts.
3. Select **Append** to add the log file information at the end of the existing log file each time the program starts.
4. Enter, edit or **Browse** for the full path name of the log file in the **Log file's name:** field.

Project Menu

The project file is a place to save all of the preparations before programming. You can also use it to restore the saved working environment. The project file includes:

- The current device's info, such as:
  - The manufacturer name
  - The device type
  - The information about the pin
  - The size
  - The data file name
  - The current buffer data

  Note: The buffer data may differ from the data file due to revisions. The revised buffer data prevails.

- All Operation Option settings
- The content of the Auto mode

Note: The project content is related to the programmer software. If the software has upgraded or has been re-installed to another folder, the saved project files become ineffective.

Load Project

Select **Load Project** from the **Project** menu to load a project file. Select the file in the File dialog box.

Save Project Files

Select **Save Project Files** from the **Project** menu to save the current working environment to a specified project file. To ensure the security of the data, you can encrypt the project file. Refer to Security and Password Input on page 57 for more information.

Downloading Library (Standalone)

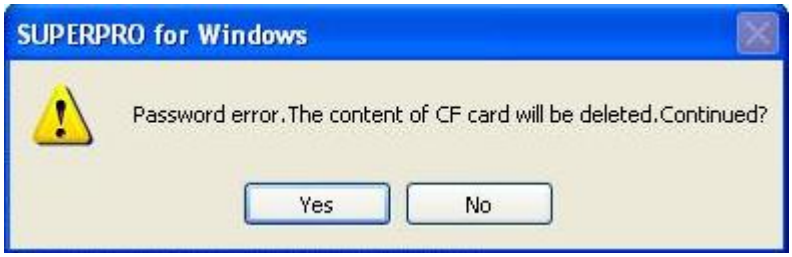
The standalone operation is when the programmer hardware is not connected to a computer, and uses information saved in its own module for volume production.

The standalone file includes all the information needed for programming the device. Make sure to prepare the standalone file before starting the programmer's standalone operation. Refer to Downloading Library Creation on page 58 for more information.

Select **Standalone** from the **Project** menu.

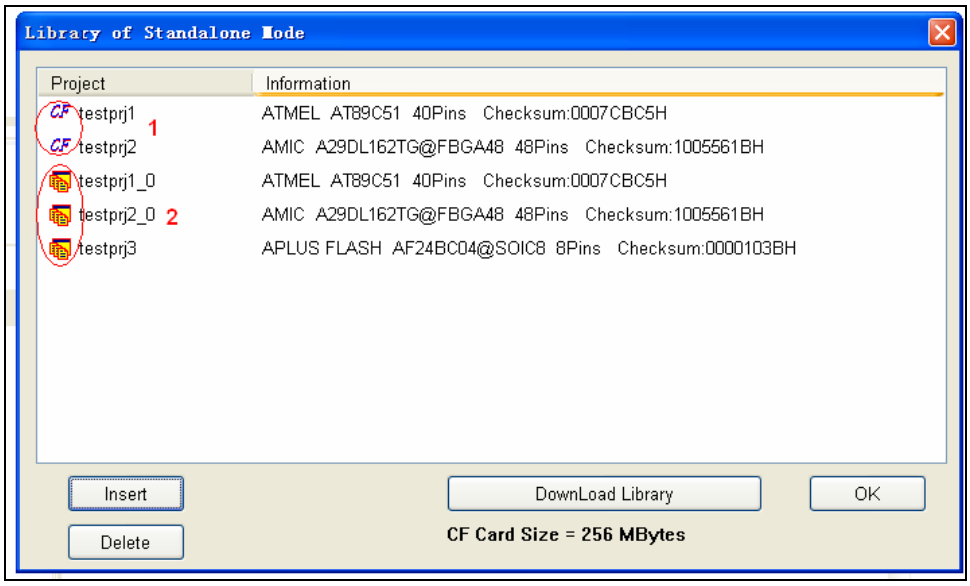
The system first checks for data in the storage module. If the system reads data in the module, it reads the data position allocation table.

The system prompts the user for the password if password protection is required. The system displays the following dialog box.



Select **Yes** to delete the data in the storage module (CF card).

The system displays the Library of Standalone Mode dialog box, illustrated below.



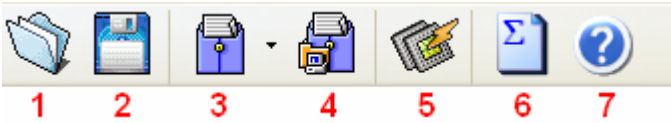
1. The Project column first lists projects in the storage module where data already exists.
2. The Project column then lists projects in the storage module where data is added.
3. Select **Insert** to add files to the project library.
4. Select **Delete** to delete files from the project library.
5. Select **Download Library** to start the downloading.

The system displays the storage capacity of the programmer hardware. In the above illustration, **CF Card Size = 256 Mbytes** means the storage capacity is 256 MB.

Help Menu

Tool Bar

The tool bar provides a quick way to execute common functions. The toolbar from the main SUPERPRO screen is illustrated below. Each tool is labeled with a number and described below. Refer to the corresponding functions in the Menus section, beginning on page 19, for more information.



1. Load
2. Save
3. Load Project
4. Save Project

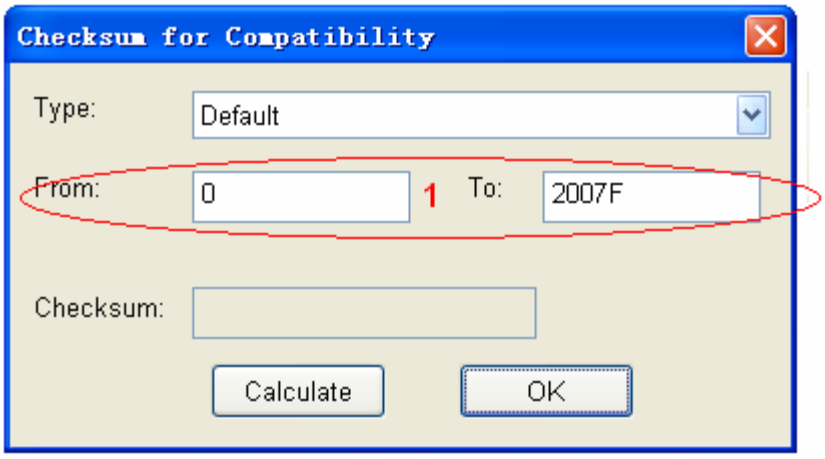
5. Factory Mode (Refer to Factory Mode on page 61 for more information.)
6. Calculating the checksum (Refer to the next section for more information.)
7. Help

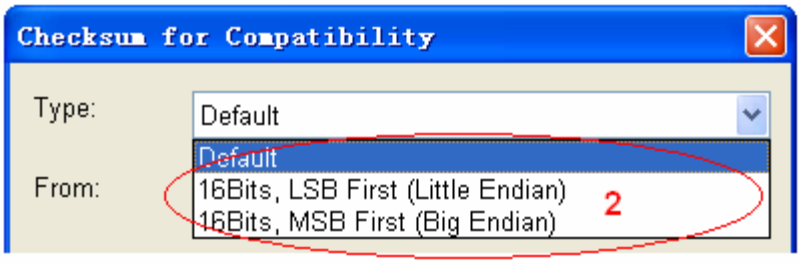
Checksum

When the data type of the data buffer is 'Data' (HEX/ASCII data), the default calculation for the checksum is to put together all the data from Address 0 to the maximum address byte by byte.

You can use Checksum to specify a different way to put together the data.

Select the Checksum tool to open the Checksum for Compatibility screen, illustrated below.



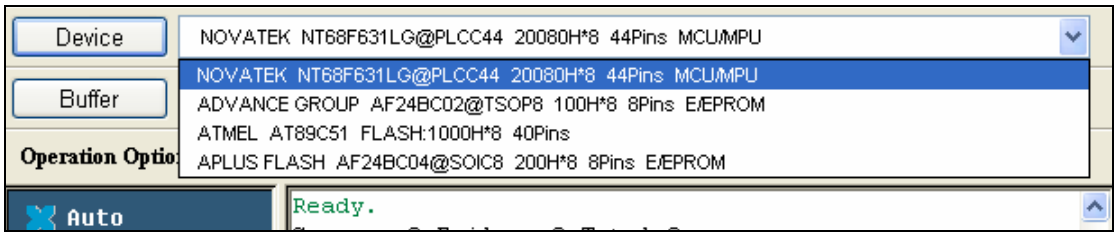


1. Enter the calculation range in the **From** and **To** fields.
2. Select the accumulation method from the **Type** drop down list.

Main Screen Device Information Display

Next to the **Device** button on the main screen, the system displays information about the recently selected device, including the manufacturer name, the device name, the size, the number of the pins, and the device types. A sample is illustrated below.

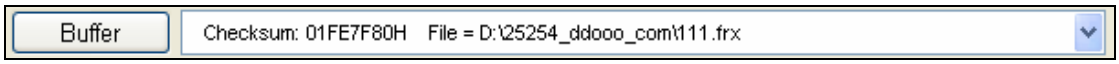
Select the drop down arrow to choose a device from the device history. This only changes the device information and is not the same as loading a project file. A sample is illustrated below.



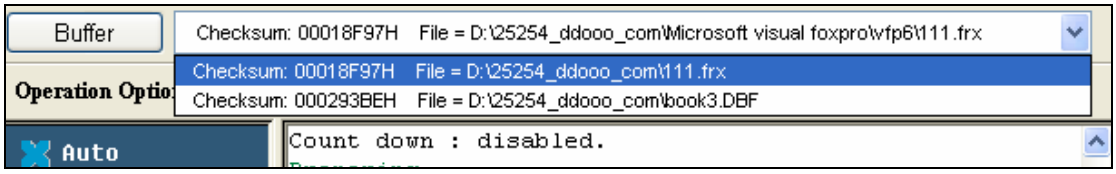
After you have selected a device, the system displays a dialog box for confirmation. Select **OK** to confirm or **Cancel** to cancel and select a different device.

Main Screen File Information

Next to the **Buffer** button on the main screen, the system displays information about the recently loaded file. It includes the checksum information and the full path and file name. A sample is illustrated below.



Select the drop down arrow to load a file from the file history. This only changes the file, not the device. A sample is illustrated below.



After you have selected a file, the system displays a dialog box for confirmation. Select **OK** to confirm or **Cancel** to cancel and select a different device.

Operation Option Bar

The Operation Option toolbar provides a quick way to execute functions from the Option menu. The user can find the corresponding operation in the menu.

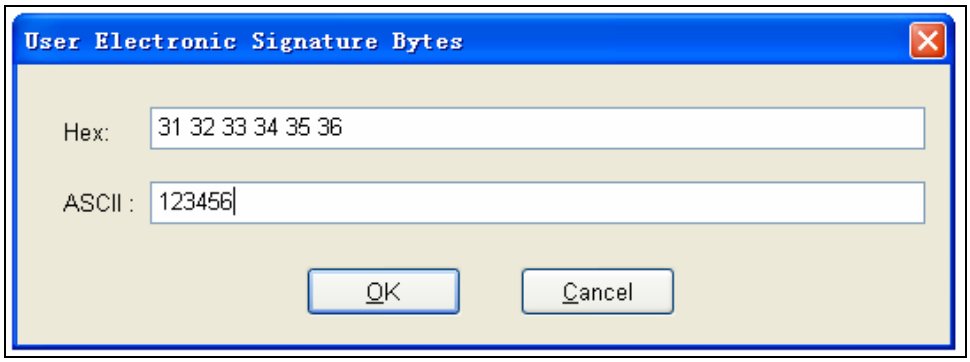
Operation Option	Edit Auto	Dev. Config	Dev. Info	Data Compare

- Operation Option: Refer to Operation Option on page 35 for more information.
- Edit Auto: Refer to Edit Auto on page 40 for more information.
- Device Configuration Word: Refer to Device Configuration Word on page 33 for more information.
- Device Info: Refer to Dev. Information on page 31 for more information.
- Data Compare: Refer to the section that follows for more information.

If you select a PLD device with a U-Field, the Operation Option Bar will look like the following:

Operation Option	Edit Auto	U.E.S.	Dev. Info

U.E.S is the edit user electronic signature. The system may change this content while reading the device or loading files. Users can view and edit U.E.S. (two display methods, Hex and ASCII).



Data Compare

The Data Compare function is applicable to most memory chips and SCM chips. Select this option to compare data in the buffer and the chip byte by byte.

If the system finds differing data, it saves the address, the data records in the buffer and the chip, and the generated list file in a 'Bin' directory file. The Operation Information screen displays the file name and path of the file.

Example: Compare result in file:
C:\SP3000U\BIN\27256.CMP

The file format is shown in the following:

Title : SUPERPRO Data Compare Result file

Displacement Buffer address	Chip Data	Buffer Data
000000D0	04	12(EPROM)
000000D1	00	34(EPROM)
000000D2	00	56(EPROM)
000000D3	00	78(EPROM)
000000D4	00	90(EPROM)
000004A5	00	AA(EPROM)
000004A6	00	BB(EPROM)
000004A7	00	CC(EPROM)

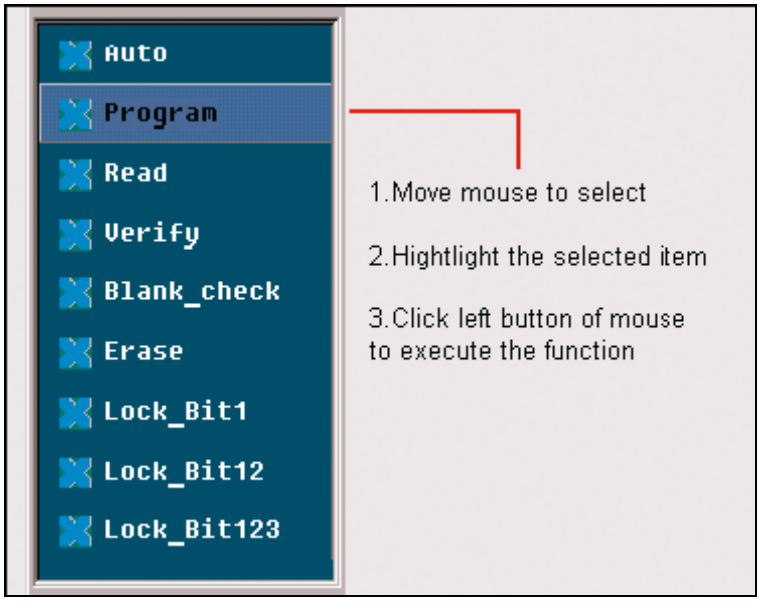
Note: If there are many buffers, the buffer names (such as EPROM) will be added. However, most buffers do not have the names.

Device Operation Window

This window displays the operations that can be executed during programming, as provided by the manufacturer. After you have prepared for programming, use this window to execute operation items quickly.

The display varies depending on the chip. A sample Operation Window is illustrated below for Atmel At89c51. Explanations for the most common operations follow.

Note: The Device Operation Window may be different for some devices with special requirements and functions. To obtain detailed information for those functions, refer to the manual of the device data or the Dev. Information screen, or contact XELTEK technical support.



- **Auto** is the automatic programming mode, which executes a series of operations on the device in a preset order. Refer to Edit Auto on page 40 for more information.
- **Program** writes the data from the buffer into the chip and verifies the accuracy of the programming in some cases.

  Note: The Verify function is performed in two modes. The first mode checks for accuracy on completion of programming a chip. If an error occurs, the system displays an error message and ceases the programming.

  The second mode does not perform verification during programming. The process continues even if an error occurs. Upon completion, the system displays a successful programming message. However, the message only indicates successful completion of programming. You must perform the verify function to ensure accuracy.

- **Read** reads the content of the chip into the buffer. Open the Buffer Edit dialog box to view the content.

  Notes: The system cannot read the content of encrypted chips.

  The system cannot read PLD devices since the test vector is not in the chip.

  If a chip includes ConfigWord, the system reads the ConfigWord data and saves it to the ConfigWord buffer.

- **Verify** compares the content of the buffer with that of the chip.

  Notes: If there is any discrepancy, verification ceases and the system displays a failure message. For most memory chips and SCM chips, the failure message includes address and data. For PLD devices, the message displayed depends on the programming reference. Only the chips that pass verification can be recognized as the chips being programmed correctly.

  Some chips do not provide the cell to cell check function. Instead, these chips provide only the accumulation check function similar to the VerifyCRC.

  A few chips do not provide the accumulation check function. The system performs the check automatically in the programming process.

- **Blank_check** checks if the device is in blank state, which is dependent on the data manual. If filling the blank data buffer is selected when the devices are selected, the programmer will fill all the HEX/ASCII buffer data with FF or 00 (Hex), and the fuse buffer with 1 or 0 after the device is selected. An encrypted chip may be displayed as a blank chip.

  Notes: EEPROM devices do not need Blank_check because the chip can be overwritten.

  Refer to Making a Chip Blank on page 59 for more information.

- **Erase** erases the chip and makes it blank. This operation can only be executed on the chip that can be erased electrically. EEPROM does not need to execute this operation. If you need a chip with a blank state, you can directly write the blank data to the chip. Some chips that can be erased electrically are able to become an OTP by configuring the ConfigWord. Please refer to the device data manual.

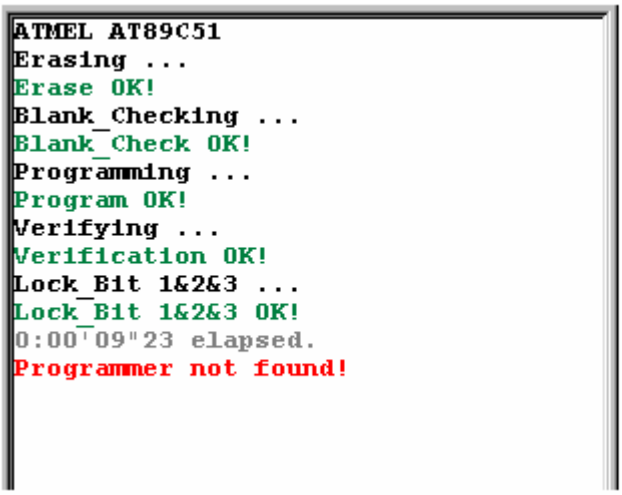
- **Protect** encrypts data to make it unreadable. Protect affects the operation function whether it is readable or not.
  - When the protection state is unreadable, the system displays the **Protect** operation in the operation function window. Execute the **Protect** function to protect the chip.
  - If the protection state is readable, the Protect options are part of the ConfigWord. Select **None** to have the system perform no protect function.

    Note: If the chips made by some manufacturers are fully protected, the data cannot be erased even with the ultraviolet ray eraser. These chips cannot be re-used.
  - The Protect function protects the data in FLASH chips. You cannot rewrite a protected chip until you have removed the protection. Use the Erase operation to remove the protection. You must use this operation with Dev. Config and identify the sector. Refer to Identifying the Sector on page 60 for more information.
- **Security** encrypts data to make it unreadable.
- **Erase_All** is the same as **Erase**.
- **Protect_All** is the same as Protect.
- **Lock_Bit** is a protection method. Please refer to the Device Data Manual for the detail information about the operations.

  Example: Atmel At89c51 has three operation items: Lock_Bit1, Lock_Bit12, Lock_Bit123. The data manual defines the meaning of each.

Operation Information Window

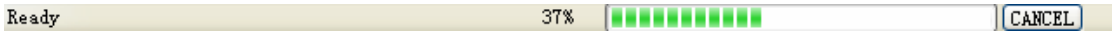
The Operation Information Window displays interaction between the software and the user. The system displays the operation process, the operation result and the historical records of information in this window. A sample is illustrated below.



The system displays successful operations in green and errors in red. If errors occur, read the error information to determine the problem.

Status Bar

The status bar displays the menu item info, the progress of operation, and a Cancel button. A sample status bar is illustrated below.

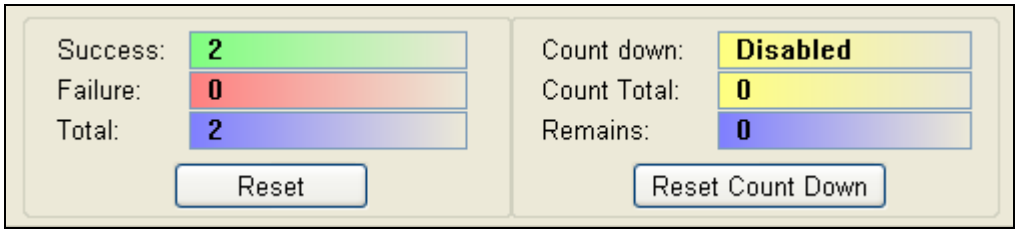


Select **Cancel** to stop the current operation. The system displays a dialog box to confirm the cancellation. The operation continues until you confirm the cancellation.

IMPORTANT!: The 'Cancel' operation may destroy the chip. Use it carefully.

Statistic Window

The Statistic Window displays statistical information. A sample is illustrated below.



The system displays the number of times according to the default counting rules:

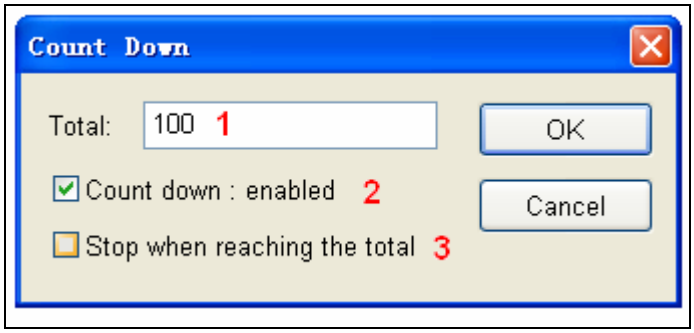
- **Success**: successful programming
- **Failure**: failed programming
- **Total**: total programming

The default counting rules:

- After successful Program execution, the Success count increases by 1. Otherwise the failure count increases by 1.
- After successful Auto execution (if Auto includes the Program function), the Success count increases by 1. Otherwise the failure count increases by 1.

Select **Reset** to set the successful times and the failed times back to 0.

Select **Reset Count Down** to open the Count Down dialog box, illustrated below. The red labels refer to the explanations that follow.



1. Enter the number of chips you are programming (greater than 0) to have the system count down. The **Count Down** number decreases by 1 every time a chip is programmed successfully, until it reaches 0.
2. Check the **Count Down: enabled** box to enable the count down function.
3. Check the **Stop when reading the total** box to ensure that no more programs are written until you reset the option.

Chapter 4 Procedures

This chapter provides information on steps for common procedures.

Programming Procedure

This section presents the operation process for the chip programming, from the beginning to the end, including the user interface and the device programming steps.

Before you start to program any devices, make sure that you have correctly installed the programmer and that the computer is successfully communicating with the programmer.

Prepare the Hardware

Before using the programmer to program a device, make sure that the programmer is installed properly, and that the PC and programmer are communicating successfully.

Make sure that you have the appropriate adapter for your SMD devices, if applicable.

Insert the chip correctly. Follow the guidelines of the standard chip insertion markings near the pin-driver socket. However, if you select SMD or a device that requires non-standard insertion, the system prompts you with the appropriate chip insertion.

Select the device

To open the Device Selection window, choose one of the following options:

- Select the **Device Selection** icon.
- Select the **Device** menu and select the **Select Device(Device)** option.

The system displays the **Device Selection** screen. Refer to Select Device on page 30 for detailed information on the Device Selection screen.

Load Data into Buffer

Load data into the buffer by reading data either from a file or a master chip.

- To read data from a file, select **Load File** from the **File** menu. The system displays the Load File dialog box. Refer to Load File on page 19 for detailed information on the Load File screen.

  After confirmation, the system loads the data into the buffer.

  Make sure the loaded data is correct by viewing on the Buffer Edit window. Refer to Buffer Menu on page 23 for detailed information.

  Note: Some Hex or S record files contain non-zero file start address. In this case, enter the start address in the File Address box.

- To read the data from a master chip, complete the following steps:
  - Insert the master chip in the socket.
  - Select **Read** on the Device Operation window of the main screen. The system copies the data from the master chip into the memory buffer.
  - Make sure the loaded data is correct by viewing on the Buffer Edit window. Refer to Buffer Menu on page 23 for detailed information.
  - If desired, you can save the data to a disk for later use. Refer to Save on page 22 for detailed information on saving files.

  Note: The read function is not available for some devices, including those that have been encrypted.

Set Options

Select **Operation Option** from the **Option** menu to open the Operation Option screen. Set each of the following options:

- Insertion Test option to indicate whether to check the pin contact before programming
- Device ID Check option to indicate whether to check the ID of the device before programming
- Beeper option to indicate whether the beeper beeps when the operation succeeds or fails
- Auto Increment Function option to indicate an automatic increase in the label number written to each chip
- Address Change option to indicate a different start and end address for the programming zone of the device
- Verification Mode option to select a specific VCC voltage to ensure that the chip is programmed correctly

Edit Auto

To set the order of the batch processing functions, select **Edit Auto** from the **Options** menu. The system displays the Edit Auto screen. Refer to Edit Auto on page 40 for detailed information on the Edit Auto screen.

Configure the Device

For devices that include the configuration word, you must set the configuration word before programming to ensure that the chip can be used on the target system.

The configuration words for some devices are contained in the data file. Therefore, the system automatically loads the configuration word into the configuration word buffer when loading the data file.

On some devices, you must manually make the selection. Select **Device**

Configuration Word from the **Device** menu to open the ConfigWord screen. Refer to Device Configuration Word on page 33 for detailed information about the ConfigWord screen.

Review the Device Information

Some devices must meet some special requirements for programming a chip, including special algorithms conventions. Read the device manual or instructions carefully and adjust the operation steps or buffer data as needed. If an adapter is required for programming, the system displays information about the adapter in the device information screen. Refer to Dev. Information on page 31 for detailed information about the device information display.

Execute Programming

Insert the chip properly into the socket. If the prompt information indicates a need for an adaptor, you may purchase the adapter.

The operation steps for programming a chip are follows:

- Blank Check

  Notes: You can skip this step if the chip is brand new.

  The Blank Check will fail on chips containing some information that can be erased. In this case, perform the Erase operation first. You do not need to erase EPROM chips for which the data can be overwritten.

- Program
- Verify

  Note: This step is necessary. The chip must pass Verification before the programming can be implemented. Some chips provide only the accumulation check function, such as VerifyCRC instead of a unit-to-unit check function. Very few chips do not provide the accumulation check function.

- Security or Protect if encryption is required

  Note: You may need to set the configuration word before performing encryption. Refer to Device Configuration Word on page 33 for more information.

Select Auto to finish all the operations in one step.

Execute Production Mode to Program Chips in Batches

Select **Production Mode** from the Option menu to program devices in large quantities. This function facilitates the job of the user who programs chips in batches.

In order to program chips in batches, the programmer always conducts the insertion test.

The programmer waits for the chip insertion. The system displays the following

prompt: **Please, insert a device into the socket**. Refer to Insertion Test on page 68 for detailed information on chip insertion error messages.

Once you have properly inserted the chip, the auto batch-processing command starts automatically.

The programmer waits for the chip removal. When chip programming is complete, the system displays the following prompt: **Please, remove the device from the socket**.

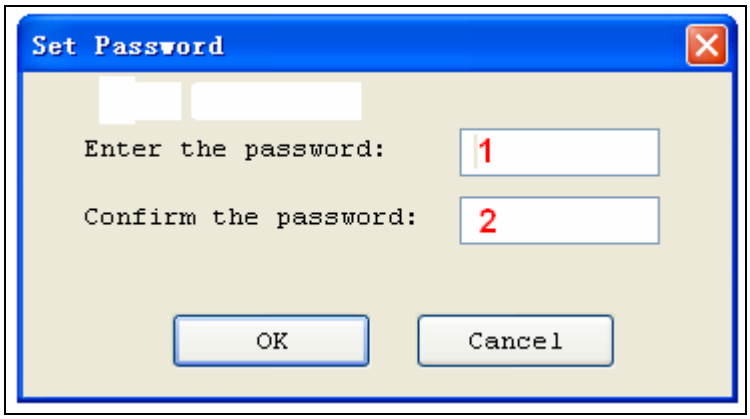
Review the Operation Information window to check the programming result. Refer to Operation Information Window on page 52 for more information.

To end production mode, select **Cancel** on the right side of the status bar.

Security and Password Input

To prevent the user's data from being used by another, the program asks the user to input the password in the operation process.

The Set Password screen is illustrated below.



1. Enter the new password in the **Enter the password** field. Leave the field blank to indicate no password is required.
2. Re-enter the password. Both fields must match for a valid password.

Note: Spaces are characters that can be used in the password.

The system asks for a password whenever performing a function that requires a password. The Enter Password screen is illustrated below.

Downloading Library Creation

Creating Project Files

Under the standalone mode, the programmer can program the devices of the same model according to the different requirements, or program different devices. The maximum number of the programming methods that the programmer supports depends on the storage capacity of the programmer hardware. The selection of the programming method depends on the information saved in the project file.

Example: Two Atmel AT89C51 chips are going to be programmed with two different files. If the storage capacity is enough, add an AMD AM27C128.

You would complete the following steps:

1. Select Atmel AT89C51.
2. Load the first file for programming.
3. Edit the Auto function to include **Erase, Blank_check, Program**, and **Verify.**
4. Execute the **Auto** function. After the successful completion of the programming operation, check the chip in the target system.
5. Save the current working environment to the specified file 'sample1.prj'.
6. Load the second file for programming.
7. Add **Lock_Bit123** to the Auto function.
8. Execute the **Auto** function. After the successful completion of the programming operation, check the chip in the target system.
9. Save the current working environment to the specified file 'sample2.prj'.

10. Select AMD AM27C128.
11. Load the file for programming.
12. Edit the **Auto** function, adding the **Blank_check**, **Program**, and **Verify** functions.
13. Execute the **Auto** function. After the successful completion of the programming operation, check the chip in the target system.
14. Save the current working environment to the specified file 'sample3.prj.

Create a downloading library and download.

Select **Standalone** from the **Project** menu.

Add the project files to the library. Refer to Downloading Library (Standalone) on page 43 for more information.

Connect the programmer to the PC. You cannot use standalone function without first updating the storage module (CF card) of the programmer hardware.

Select **Download Library** to start the downloading. To ensure the security of the data, you may set a password to protect the data. Refer to Security and Password Input on page 57 for more information.

Wait until the downloading is finished.

Now the programmer is able to work under a standalone mode. Refer to Execute Production Mode to Program Chips in Batches on page 56 for more information.

Error messages:

"Please check connection of the programmer": The programmer hardware is not connected or an error occurs when the storage capacity is read. In this case, the generated standalone library cannot be used.

"Library size larger than capacity of CF card": The added project needs a larger storage capacity. Please increase the storage capacity (CF card capacity).

"Download library OK": The library has been downloaded into the programmer hardware successfully and now the programmer can be used in the standalone mode.

"Download library error": The library has failed to download into the programmer hardware. Check whether the CF card is normal.

Making a Chip Blank

For the chip that can be erased electrically, execute the **Erase** operation.

For the chip that can be erased with ultraviolet ray, use the ultraviolet ray eraser for 3-30 minutes.

Note: OTP cannot be blanked once written.

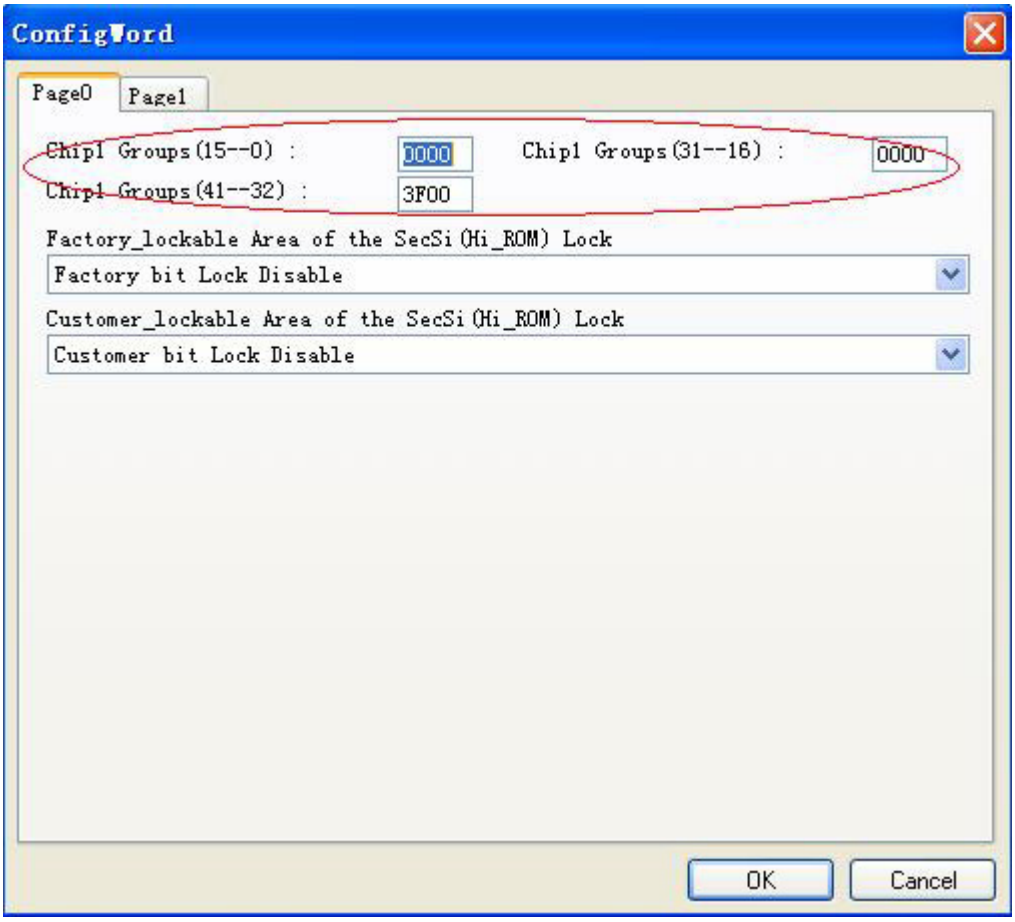
Identifying the Sector

You must identify the Sector to protect some chips.

There are two ways to select the Sector:

1. Enter the address of several sectors (maximum of 8)to decide the sectors to protect. After the **Protect** operation is performed, the identified sectors are protected.

2. If the chip has been divided into several sector groups in advance, you can only choose these groups to be protected, as shown in the following illustration.



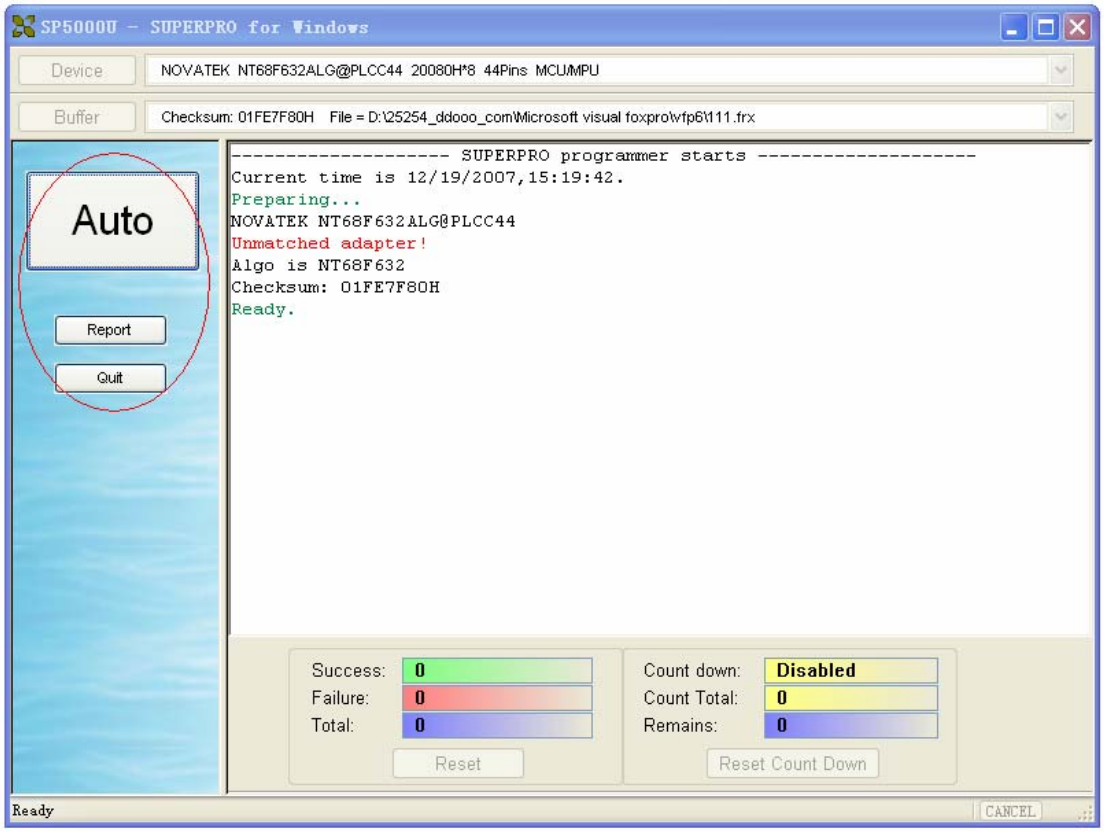
Factory Mode

This mode is designed for factory volume production. This mode only allows a few functions (such as Auto) to prevent operation errors from destroying the chips or to prevent wrong data from being written to the chip.

The administrator can set a password for entering the system to ensure the data security. If the password has been set, you will need to enter the password when exiting the system.

Select Factor Mode from the Toolbar, as illustrated below.

The system displays a modified main screen, as illustrated below.

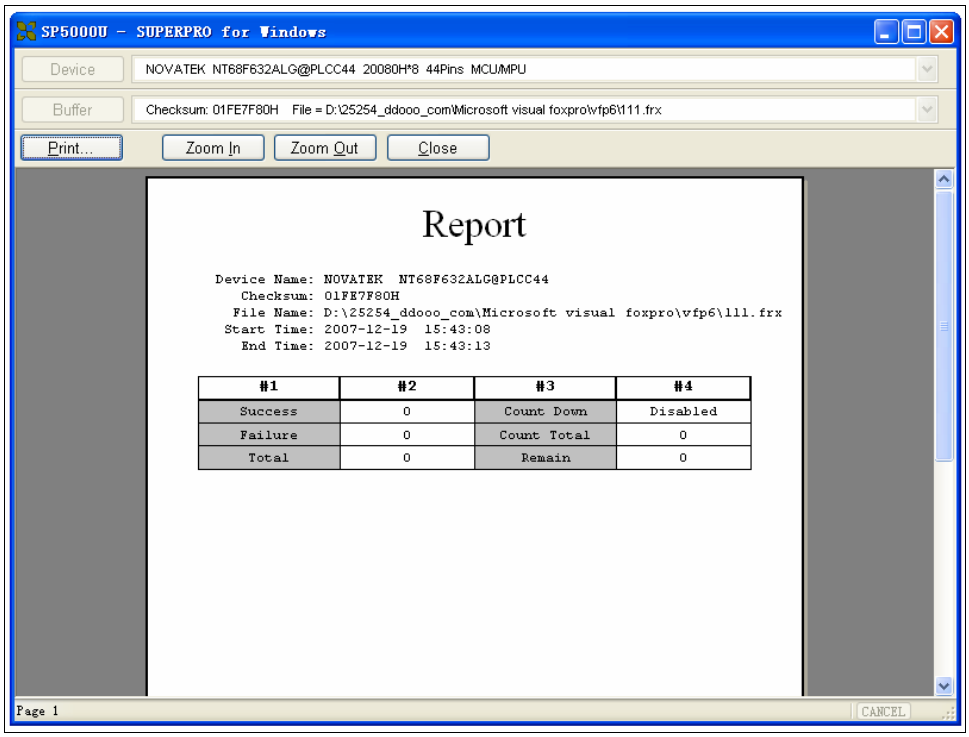


Only the following three options are active. The system does not allow any other operations.

- **Auto**: Execute the Auto programming operation.
- **Report**: Generate the statistical report and display it.
- **Quit**: Exit.

Factory Mode Report

Select **Report** from the Factory Mode Main Screen to display the statistical report. A sample is illustrated below.



You can print the report on A4 size paper.

Chapter 5 Detail Explanations on Common Problems

You can monitor the programmer operations through the Operation Information Window and message window. This chapter helps define common problems related to programmer actions and errors.

Dealing with Data Files

This section explains common problems involving the File Type dialog box.

Solve Invalid File Type or File Data Overflow Error

The data for programming is usually stored in the data file. Commonly used formats are Intel Hex, Motorola and Tektronix. The files of these formats keep data in text mode, which include data and address information. Because the offset address is not always zero, you may incur errors when loading the file directly.

For example, the location of data in buffer (as viewed in the Edit Buffer window) may be incorrect or you may receive an error message when loading the file, such as **"Invalid file type, or all file data overflow".**

Select one of the following options to solve this problem.

- Confirm the file type and ensure the buffer can load all the files.
- Find the offset address of the file. When loading the file, select the **Show Offset Address (Minimize) After Loading** option to display the start address of the file after the file is loaded.

  Example: **File Offset Address (min): 0x0000E0000 (Hex)**

  This means the offset address of the file is not 0. You must move the data at the address to the address 0 of the buffer. To do that, load the file again, and enter **E000** as the **File Address**.

Refer to the Load File section on page 19 for more information on loading files.

Separate File Data for Programming

By using the File Mode selection in the File Type dialog box, you can select the data of a file and write it to several chips of the same type. This allows you to load the file several times according to your needs.

There are three methods for separating the file data.

1. Based on byte (8bit), divide the file into two parts by the odd and even address. Program the two parts into two chips respectively.

   When loading the file, select **Even** as the File Mode to write the data in the addresses 0,2,4,6, etc. to one chip. Select **Odd** as the File mode to write the data at the address 1,3,5,7, etc. to the other chip.

2. Based on the byte (8-bit), divide the file into four parts by address of the first byte, the second byte, the third byte and the fourth byte out of every four bytes. Program the four parts to four chips respectively.

   When loading the file, select the **1st byte of 4** as the File Mode to write the data in the addresses 0, 4, 8, 12 (or C if HEX), etc. to one chip. Then select **2nd byte of 4**, **3rd byte of 4**, and finally **4th byte of 4** to write the remaining data to the other three chips.

3. Based on the word (16-bit), divide the file into two parts by address of the first two bytes and the last two bytes out of every four bytes. Program the two parts to two chips respectively.

   When loading the file, select the **1st 2_type of 4** as the File Mode to write the data in the addresses 0, 1, 5, 6, etc. to one chip. Then select **2nd_byte of 4** to write the remaining data to the other chip.

Program Two or More Files to One Chip

You can load several files into the buffer and write them to the chip. The following explanation gives an example of programming three files (Sample1, Sample2, and Sample3) to one chip. The example assumes the following:

- Write the data from the address 200 (Hex) of the file Sample1 to the address 0 of the chip.
- Write the data from the address 0 (Hex) of the file Sample2 to the address 3000 (Hex) of the chip.
- Write the data from the address 4000 (Hex) of the file Sample3 to the address 4000 (Hex) of the chip.

1. Select **Edit** from the **Buffer** menu. Make sure that the **Buffer clear on data load** option is not checked.
2. Select **Load** from the **File** menu to load the Sample1 file.
   - In the **Buffer Address** field, enter 0.
   - In the **File Address** field, enter 200.
3. Select **Load** from the **File** menu to load the Sample2 file.
   - In the **Buffer Address** field, enter 3000.
   - In the **File Address** field, enter 0.
4. Select **Load** from the **File** menu to load the Sample3 file.
   - In the **Buffer Address** field, enter 4000.
   - In the **File Address** field, enter 4000.
5. Continue with programming.

If there are no changes in the file requirements or content of the three files (Sample1, Sample2, and Sample3), you can save the data in buffer to a new file, such as Sample4, for easy loading and programming next time.

POF Format Files

When programming ALTERA PLD devices the system saves the data in POF format files. When loading a POF Format file, the system displays a File Type dialog box. Select one of the following:

1. POF file: The system transforms the data into the Fuse data and checks the data when you load it to the buffer. You may receive one of the following errors:
   - **Unmatched file to this chip**: This POF file does not match this chip. The POF file is related to the device when it is compiled. The error indicates that this POF file is not for this device. Replace the chip with a matched one.
   - The POF file has errors.
   - **Load POF file dll error (not found)**: An auxiliary file has errors. Contact technical support for assistance.
2. Files of other formats are mainly the JED files. The system saves data in the buffer in a JED file after reading the chip. You can use this format the next time you load the file.

Adapter Selection

The standard programmer configuration supports the DIP devices with 48 or less pins. If you want your programmer to support a device with more than 48 pins, you will need an adapter.

There are two types of adapters: universal and specific.

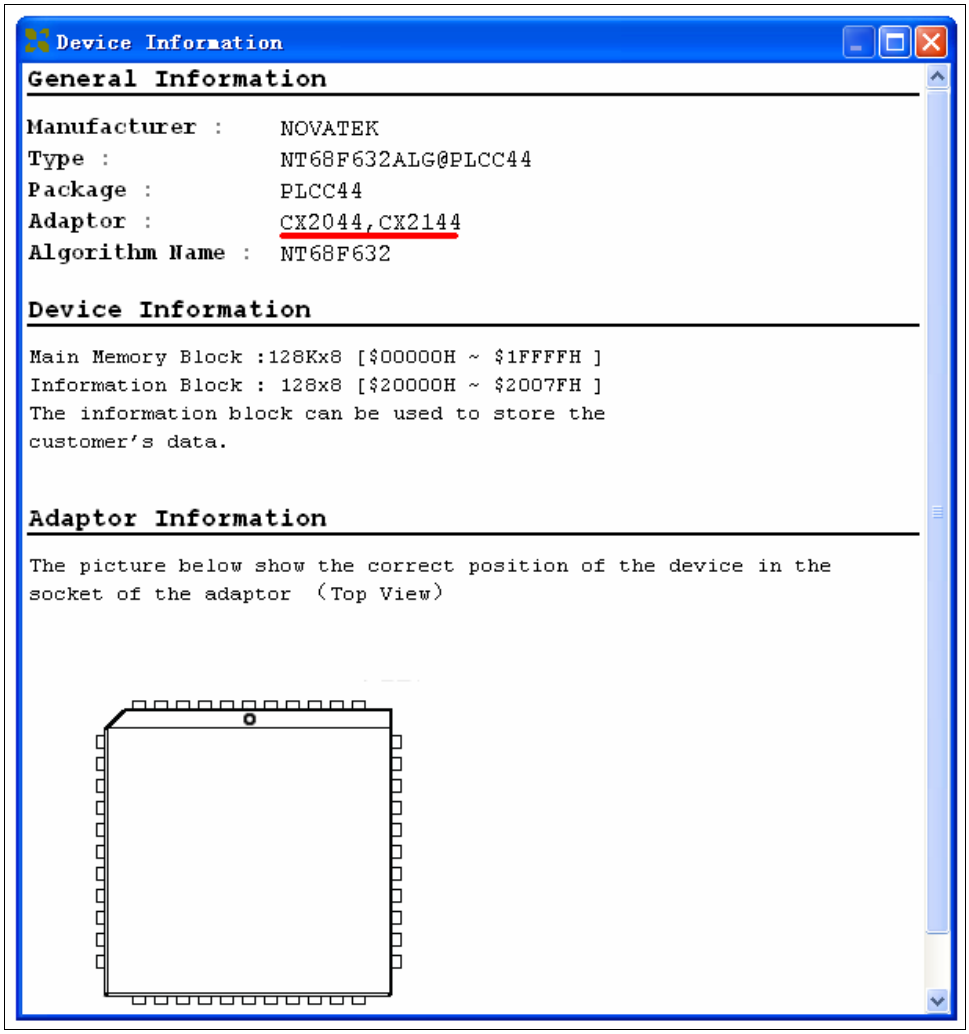
- Universal adapters are applicable for all devices of the same encapsulation and the same number of pins.
- Specific adapters are applicable for certain devices only. For example, if the number of the pins for economical programmer hardware is not enough, specific adapters may solve the “insufficient number of the pins” problem. This way, the programmer can also support devices with more than 48 pins

To use devices with pins between 48 and 144, you can choose either the universal adapter or the specific adapter.

To use devices with pins more than 144, you must use the specific adapter.

Selecting the Correct Adapter

The Device Information screen displays adapter information. The following illustration shows PLCC44 as an example.



In this example, the adapter order number is **CX2044** or **CX2144**.

There are different order numbers for adapters because the sockets are from different manufacturers. There may also be a minor difference in size.

Note: To ensure success, do not use self-made adapters. For best results, obtain the adapter made by Xeltek.

Other Messages

Other error messages are explained below.

- **Please edit 'Auto' first**: The Auto consequence is blank. Edit Auto and try again.
- **Enter a string to search for**: The Hex Edit Search dialog box is blank. Make sure to enter a specified string or ASCII codes to search for in the Buffer Edit dialog box.
- **The string for search is blank. Input it in Search dialog**: Enter a string in the Buffer Edit dialog box. Then you can use **The first time search**, **Continue to search**, or **Search Next** functions.
- **Search pattern not found !**: The system did not find a matched string or ASCII code.
- **User chip is MfgID = 0089, DeviID = 0051**: This display on the Operation Information window indicates that the chip's ID detection function detected a different ID than specified in the manual.
- **ID check error. Ignore**: You can ignore the detected ID error and continue the operation.
- **ID check error**: The operation stops because of the ID check error. If you have good knowledge of this chip, you may ignore the ID error and continue the operation. Exercise caution to avoid damaging the chip.
- **Pin check error. Ignore**: The system indicates an error during the pin check. Ignore it and continue.
- **Pin check error**: An error occurred during the pin check, and the operation ceased.
- **Programmer not found**: The programmer is not connected to the PC or it is connected improperly. Check the connection. Refer to Chapter 2 System Installation on page 10 for more information.
- **Programmer not ready**: Turn off the programmer power switch and turn it on again after a few seconds.
- **Programmer is running**: The programmer hardware is operating. Wait until the operation is finished.
- **File open error**: The open file operation failed.
- **Out of Memory**: The memory overflows.
- **Production mode isn't available for this chip**: Since the production mode is dependent on the pin check, the chips without pin check function are not applicable for production mode.
- **Cancel production mode**: Cancel the production mode.
- **Too long file name**: The file name is too long.

- **User cancel**: The system displays this message after you select **Cancel** on the Status bar.
- **Can't cancel!**: The **Cancel** operation has failed.
- **Not a project file**: The file selected is not a project file. After the software is updated, the software may not be able to identify a previously created file. You will need to create the file again.

Insertion Test Errors

The following error messages may occur during the Insertion Test.

- **No device in the socket.**: The system did not detect a device in the socket. Insert a chip.
- **Find non-connection or poor pin contact**: The system detected poor pin contact. Check the pin contact. If you find nothing wrong with the connection of the noted pin(s) replace the chip with another in case the first chip is broken.

  The pin number refers to DIP chip's pin number or the adapter's pin number.
- **Device inserted backwards or damaged.**: Take the chip out, reverse it, and insert it again properly. If the problem still exists, the chip may have been damaged.
- **Device inserted upwards 3 slots!**: The chip has been inserted improperly. Refer to Dev. Information on page 31 for more information.
- **An unmatched device (40 pins) inserted!**: If the system detects pin contact errors for a chip with more than 48 pins, it displays the serial number of the DIP socket of the programmer. Check the adapter.

For any insertion test error messages, choose one of the following options.

- Select **Abort** to stop the programming.
- Select **Retry** to run the Insertion Test again after correcting the problem.
- Select **Ignore** to continue the program.

Device ID Check Error

If you selected the **ID Check** option on the Operation Option (General) screen, and the system checks the device ID. The system displays the following error message if it detects an error.

- Select **Yes** to ignore the error and continue programming.
- Select **No** to abort the programming. When the ID is unmatched, the system searches for matched devices and display them in the Operation Information window.

Appendix

XELTEK periodically publishes upgrade software on the XELTEK website. You can download and upgrade your software from the website. Non-users may download the software for evaluation.

Troubleshooting

If the User Manual does not answer your questions, first contact your sales agent or the distributor. If you still need technical assistance you can call XELTEK between 8:30 and 17:30. Make sure you have your product serial number before calling.

Before contacting XELTEK, check the following to ensure you get the best service.

- Read the User Manual.
- If you receive an error message that is not explained:
  - Make sure you can repeat the circumstances that created the error.
  - Write down the error message.
- Make sure you have your product serial number.
- Check your computer configuration, including computer brand, free memory size before starting the software, the video adapter brand, and the operating system.
- Make sure you are at your computer when you call so the engineer can guide you through the solution.

Contact Information

Xeltek Inc.
1296 Kifer Rd. Suite #605
Sunnyvale, CA 94086
United States

General Consultation	001-408-530-8080
Order/Sales	001-408-530-8080 ext # 600 Email: sales@xeltek.com
Domestic Sales	001-408-530-8080 ext # 605
International Sales	001-408-530-8080 ext # 603
Technical Support	001-408-530-8080 ext # 604 Email: techsupport@xeltek.com
Fax	001-408-530-0096

Website: http://www.xeltek.com

License Agreement

The copyright of the program and the User's Guide remain the property of XELTEK.

You may:

1. Copy the program for back-up purposes ONLY in support of its use on a single computer.
2. Transfer the program and license to another party if the other party agrees to accept the terms and conditions of this agreement.

You may not:

1. Use this product on a computer system or network that allows the program to be operated by more than one user at the same time.
2. Modify, copy, or transfer the User's Guide or other documentation or any copy.
3. Decompile or disassemble any program modules or encrypted devices.

Warranty

XELTEK has a strict quality assurance system. If the software or hardware has any defects, it can be repaired or replaced free according to the specific situation within one-year period after the user buys the product.

The warranty is based on proper installation of the software and usage in the specified working environment.

XELTEK is not responsible for the following cases:

- The product has been repaired, used, or changed by any organizations or persons who are not authorized by XELTEK.
- The product has been damaged because of misuse or careless use.
- The product serial number has been changed by the user or some other person.
- The program errors caused by the physical damage of the disk.