

Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

Xeltek, Inc.

Note: PLEASE INSTALL THE SOFTWARE FIRST BEFORE CONNECTING THE PROGRAMMER TO A PC.

Copyrights

Software Copyright 2004-2014 XELTEK

User's Guide Copyright 2004-2014 XELTEK

Copyrights © 2004-2014 XELTEK

The distribution and sales of the 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.

SUPERPRO<sup>®</sup> is the trademark of XELTEK.

# 1. General Description

## 1.1 INTRODUCTION

### 1.1.1 What is Superpro9000?

The SUPERPRO 9000 is a cost-effective, reliable, and high-speed universal programmer for mass-production, with two modes of operation.

**PC mode:** Programmer connects to the PC through a standard USB cable. A menu-driven software makes chip programming easy.

**Standalone mode:** In this mode, programmer is not connected to the PC. A keypad and an LCD display on the programmer are used to operate the programmer. Chip algorithm(s), data, and option settings are stored in a CF card. Users must first operate in PC mode to transfer this information into the CF card.

#### General features

- 4 complete independent modules, asynchronous and high efficiency programming mode
- High programming speed
- PC and stand-alone modes in one unit. 6 operation keys, 2.5 in x 1 in LCD display, 4 CF card slots
- 48 pins universal pin-drivers design. Supports 14,000+ devices and growing.
- Chip insertion test
- Two programming modes: Synchronous mode, under which four chips are programmed at the same time. Asynchronous mode is where each module works independently with respect to other modules. Operation starts as soon as the chip is inserted in the socket.

#### **Software features:**

- supports Windows 95/NT/2000/XP
- Support multiple file formats including Binary, Intel (linear & segmented) Hex, Motorola S, Tektronix (linear & segmented), Jed, POF etc.
- Device insertion test (48 pins or less) to detect defective chip, improper inserted device and poor contacted pin (chip part dependent)
- Integrated full screen buffer editor with commands such as fill, copy, move, swap, and more.
- Project and history records.

## 1.1.2 Manual Organization

This booklet consists of five main chapters:

Chapter 1: Introduction to the SUPERPRO series, including system requirements

Chapter 2: System set up information, such as set-up of the hardware and software as well as solutions for communication error if encountered

Chapter 3: Pre-programming guidance and quick start

Chapter 4: Explanation for software commands and functions

Chapter 5: Guidance for trouble shooting and error messages

## 1.1.3 Manual convention

### **The following conventions are used in this manual :**

- The names of all keyboard keys are enclosed in angle brackets, For example the Enter (or Return) key is shown as <Enter> and the Page Up key is shown as <PgUp>
- The cursor keys are shown as follows:  
Left arrow key = <Left arrow>  
Right arrow key = <Right arrow>  
Up arrow key = <Up arrow>  
Down arrow key = <Down arrow>
- Unless stated otherwise, keystrokes are not case-sensitive. e.g.: Both 'A' and 'a' are acceptable.

## 1.1.4 System Requirements

### **The minimum requirements are as follows:**

- A Personal Computer, IBM or compatible, desktop or laptop, one USB port (USB 1.0 and above)
- Windows 98/NT/2000/XP operating system
- CD-ROM driver
- Hard disk of at least 20MB of spare capacity

## 1.1.5 Programmer Package

### **Standard package contains following:**

- A programmer main unit
- 4 Compact Flash card
- An USB cable
- An external power supply
- The installation software CD
- User's Guide
- Registration card

# 2. System Setup

The software installation is straightforward and self-explanatory. If it is your first time to use the USB-based programmer of XELTEK, this chapter will help you to properly install the software and connect the hardware. USB

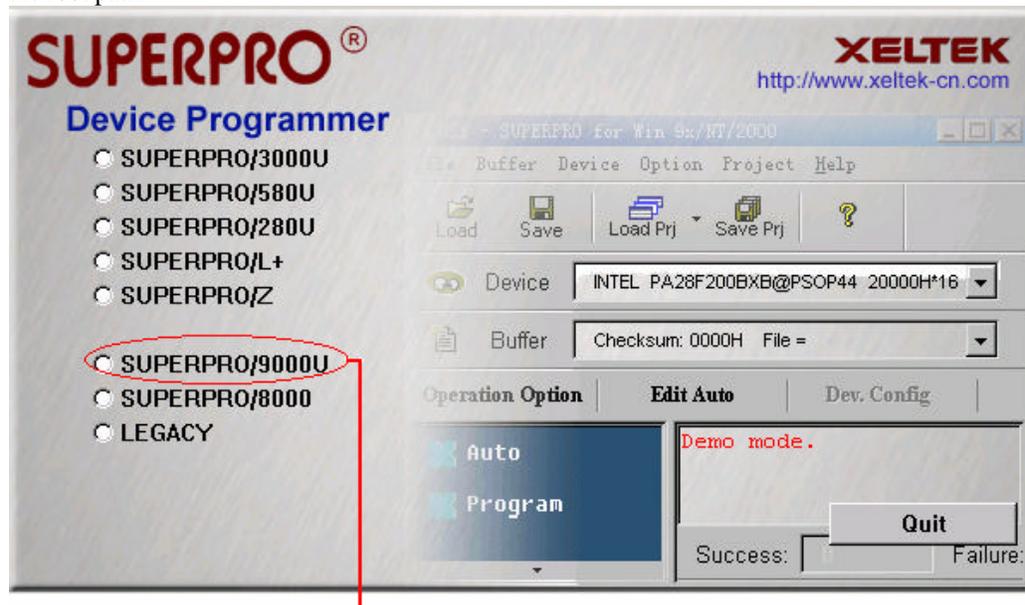
Xeltek, Inc.

devices are PnP devices. If the programmer is connected to the PC prior to software installation, Windows will start with “new hardware wizard” to scan all available INF files, and find out the appropriate driver. In order to avoid unwanted complication during installation of USB device, we strongly suggest that you install the software first. The setup program will handle the needed INF files and drivers automatically.

## 2.1 Software Setup

### 2.1.1 Setup from CD-ROM

Insert the CD into a CD-ROM drive and the setup program will start automatically, if not, please run SETUP.EXE in the root path.



Select the module you buy, then click “Setup” button

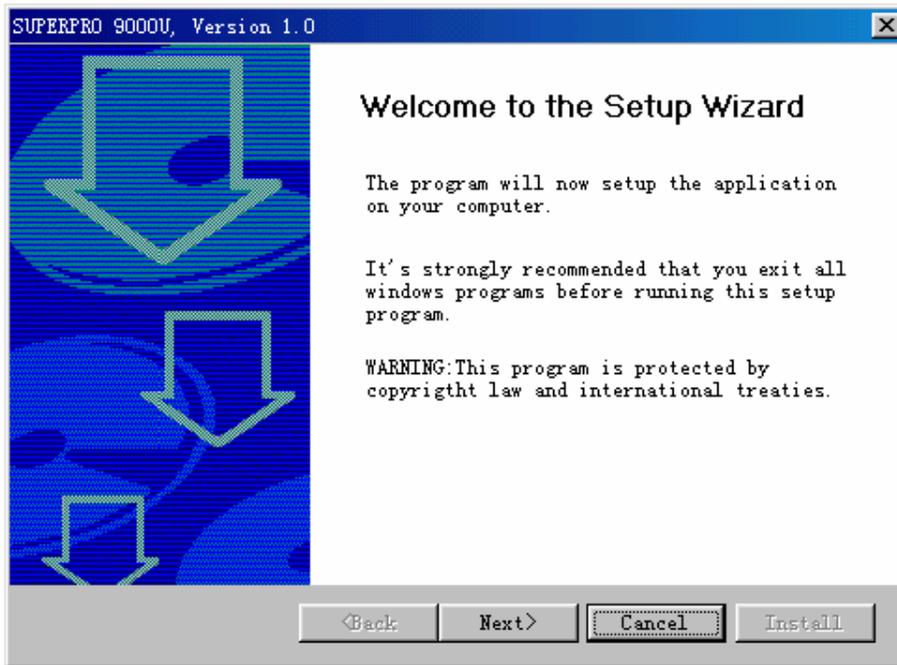
Caution: Each model has its own software, please do not mismatch.

### 2.1.2 Download software from the Internet

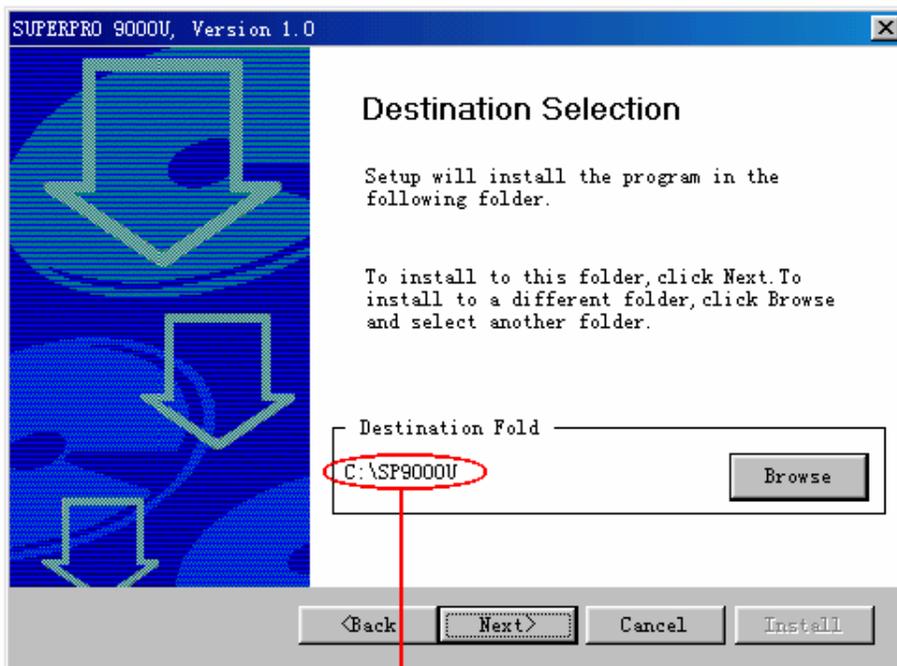
You can download the software for your programmer at [www.xeltek.com](http://www.xeltek.com). It is usually a self-extracting file. You only need to run this file to setup the software.

### 2.1.3 Setup Process

The setup procedure goes step by step, you can change the default settings as you needed during setup process. First, setup wizard, please read the text carefully.

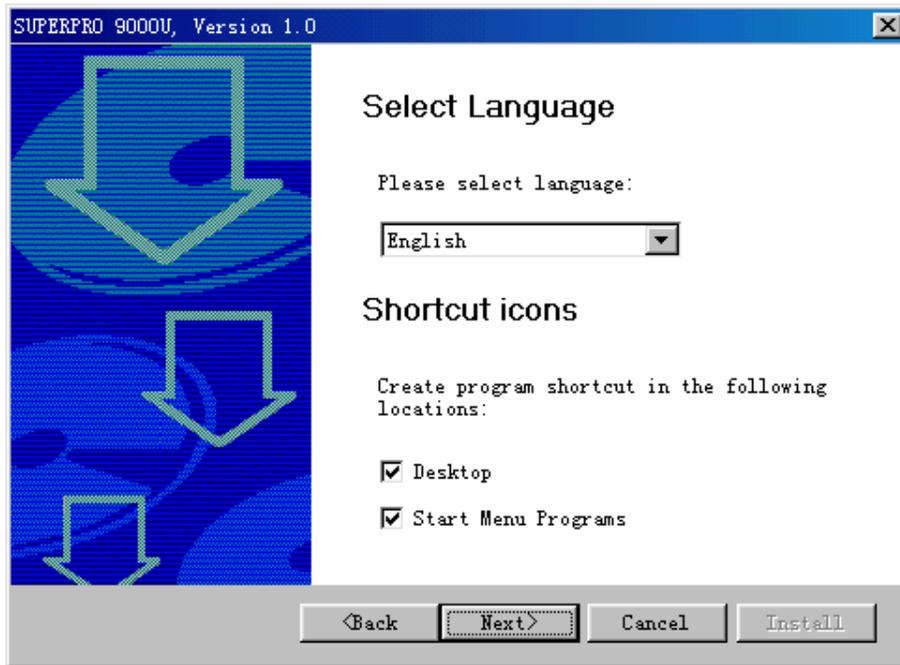


Click "next" button  
Second, select destination fold



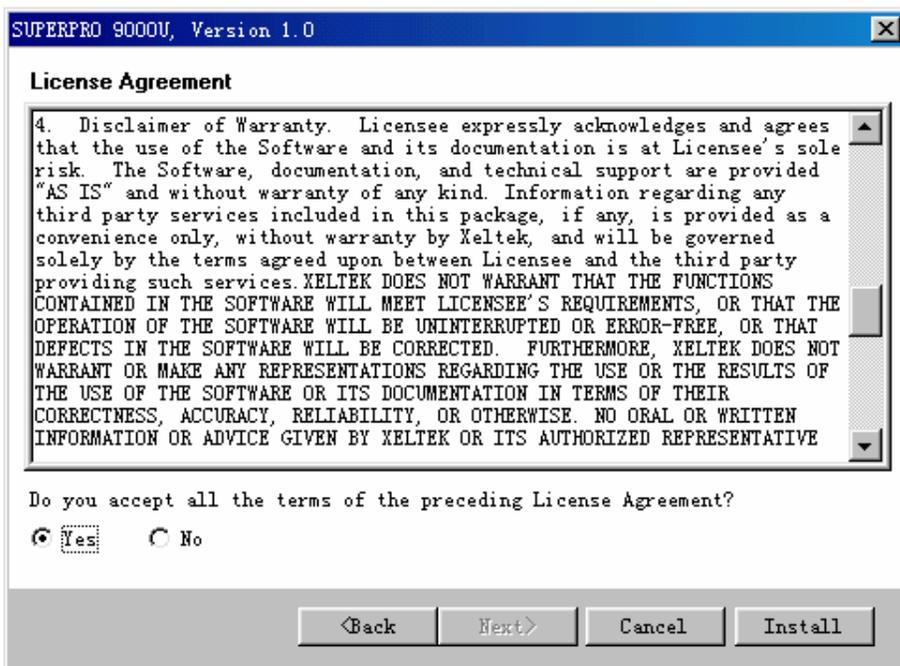
Select the folder in which you install the software, click "Browser" button to change the default folder. Then click "Next" button to go on.

Change the checkbox if you want.



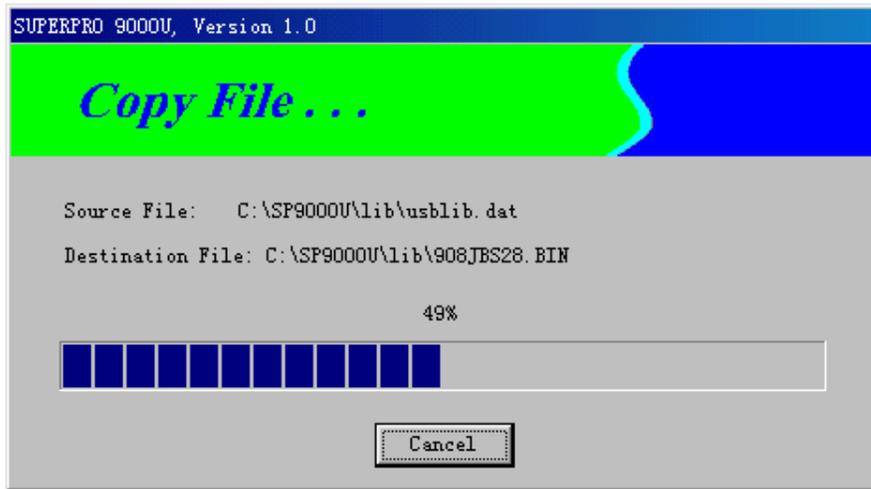
Click "Next" button

Then comes the License Agreement

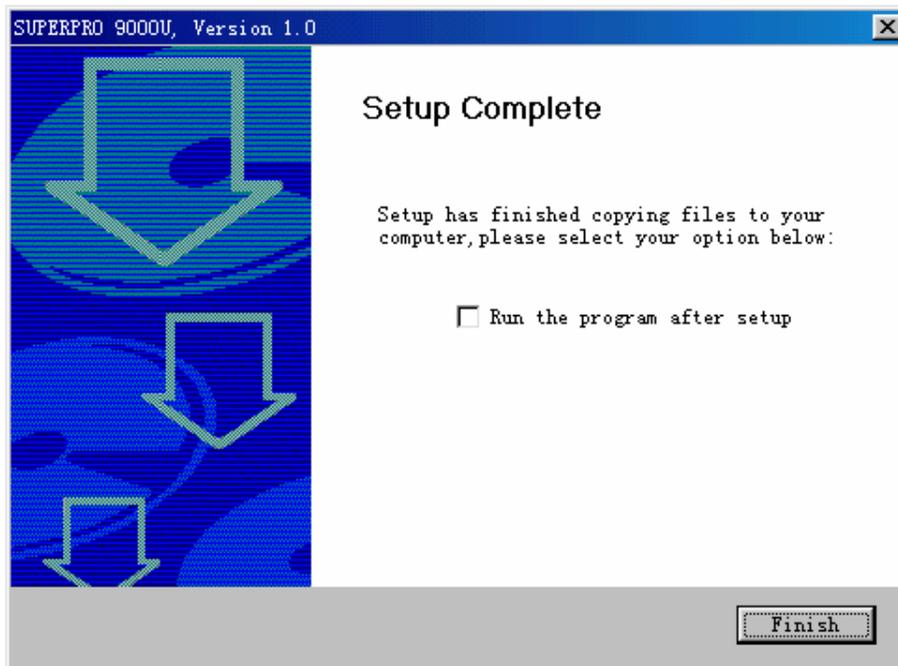


The installation process will not continue unless you accept the terms. The License Agreement can also be found in the appendices.

Xeltek, Inc.



Click "Finish" button to finish the setup.



## 2.1.4 Hardware Setup

Connect the programmer to the computer's USB port. A minimum of one CF card must be inserted in the programmer before turning on the programmer. If you are using one CF card, insert the card into the slot for module 1. Having no CF cards installed into the programmer may result in a communication error when turning on the software. The CF cards are used as memory storage for each module, thus a module without a CF card is disabled. Turn on the programmer power switch after inserting the CF card(s). The driver installation is very simple; just wait for installation to finish of the new hardware wizard.

## 2.1.5 Running the Program

The software for the programmer has a standard Windows interface, including pull down menus and buttons, and etc. Double click on the software icon located on your desktop to run the software. As the program starts, it will

Xeltek, Inc.

communicate with the programmer immediately and begin initialization.



If communication error occur,



Please try the following:

- Click on the Retry button.
- Shutdown software and turn off programmer. Afterwards, turn on the programmer and wait for a few seconds

Xeltek, Inc.

than turn on the software.

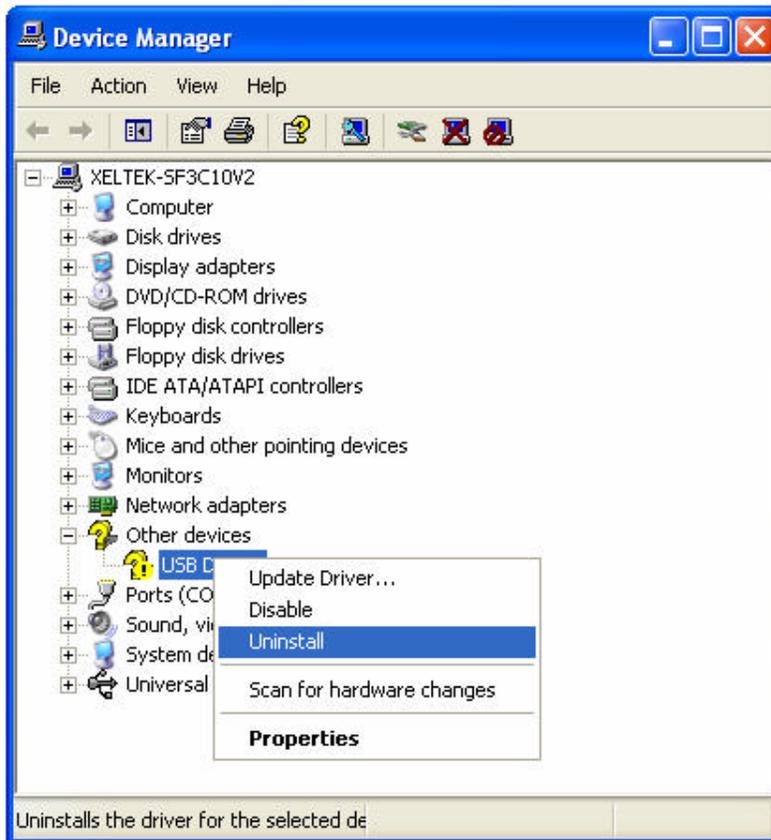
- Go to section 2.2

## 2.2 How to resolve the communication error

If incorrect installation makes the PC fail to communicate with the programmer, please follow the following steps.

Windows 98/ME: reinstall the software, turn off the power switch of the programmer, wait a few seconds, turn on the switch, wait until the drivers are installed, run the program again.

Windows 2000/XP: With the programmer's power switch turning on, open the device management (in 'Control Panel', switch to classic View, click on the System icon, click on Hardware tab, and finally click on Device Manager button.), find the USB device with "?", right click it and select uninstall. Then reinstall the software, turn off the switch of programmer, and turn on after a few seconds.



Caution: Under Windows XP, there are some options during the driver installation, please don't change the default settings.

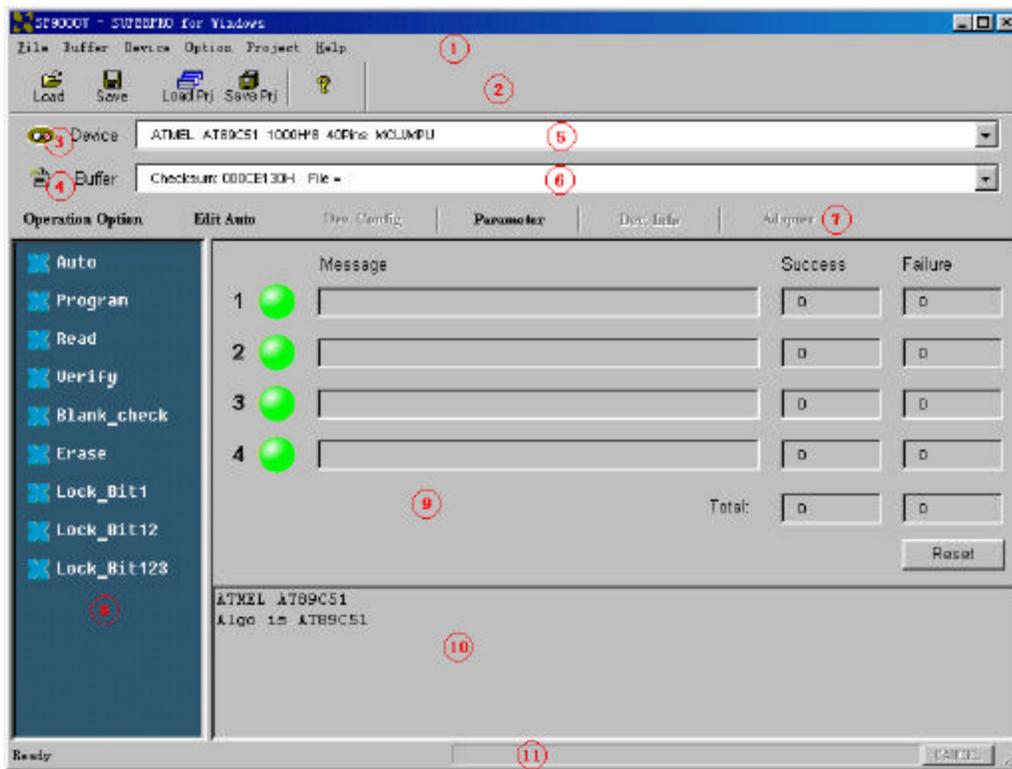
## 3 Quick Start

This chapter helps users understand the whole process of IC programming. The content includes:

- A description of the software.
- The steps of device programming

## 3.1 Description of Interface

The interface is as follows:



1. Main Menu
2. Tool bar
3. Select Device
4. Edit Buffer
5. Device Messages
6. File Message
7. Operation Tool bar
8. Device Function
9. Operation Message1
10. Operation Message2
11. Progress Bar

## 3.2 The process of the programming

### 3.2.0 Hardware Preparation

First, make sure the programmer is properly connected to the PC, and communication is established successfully with the PC. Some SMD devices need adapters, please contact Xeltek or its dealers for your adapter needs. Second, insert the chip or adapter correctly on the ZIF-48 socket of the programmer. Instructions for SMD and non-standard insertion will appear on the screen when the selection of the device is made on the software. If no special adapter/instruction information appears, than most likely you have DIP IC package. This IC package is always inserted bottom justified on the ZIP-48 socket of the programmer.

#### 3.2.1 Select device

Click on either the Device pull down menu or Device button to select your chip for programming. A dialog box will appear where you can select your chip. You can manually enter your part number into the search box to

Xeltek, Inc.

directly go to your part or you can search for your part by first selecting the Device Type (E/EPROM, BROM, SRAM, PLD, or MCU) followed by manufacturer and device part number from the screen. Click OK button or double-click the device to confirm your selection.

## 3.2.2 Load data into Buffer

The programmer accesses data in the buffer. User file should be loaded into the buffer before programming. Another buffer data source is the master chip.

### Loading a data file

You may load a data file into the memory buffer by either clicking on the File menu and selecting Load File to browse for your data file or by clicking on the Load button. After selecting your data file, a File Type box will appear where you will need to select the appropriate file type of your data file. At this point, the selected data will be loaded into the memory buffer. Please go to the Buffer Edit screen and check if the data is loaded properly.

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

### Loading data from a master chip

Loading data from a master chip begins by reading data from a chip. Insert the master chip/adaptor with chip inserted on the programmer and select the chip in the software. Click on the Read function located in the device function area. The result of the Read function, will copy the data from the master chip into the memory buffer. At this point you may go to the memory buffer edit screen and check if the data is loaded correctly. The data may be saved to a disk for later use.

Note 1: Only the chip in the master module (#1) is readable.

Note 2: Some devices have no read function or are encrypted. Data cannot be read from under this condition

## 3.2.3 Options

1) Operation Option including:

- Insertion Test, checks the pin contact before programming.
- ID Check, check device's ID before programming.
- Beeper On, beeper sound off when the operation succeed or fail.
- Verify Mode, gives the user the option to choose specific VCC voltage values during verification in order to check if the chip is correctly being programmed. Use appropriate VCC values according to your chip.

2) Edit Auto, gives the user the option to create batch commands to automatically perform multiple device functions. Click on the Edit Auto button located on the Operation Toolbar or under Options pull down menu and add/delete functions you want listed under the Auto list. Clicking the Auto function in the device function area of the software will run these batch commands. Normally, the following programming steps are included under Auto list,

- Erase
- Blank\_check
- Program
- Verify
- Security or Protect

3) Dev. Config, gives the user the capability to enable/disable available options specific for a chip. These options must be properly set before programming to ensure the chip can be used on the targeted system

4) Dev. Information, some devices have special requests or conventions of programming so user should adjust operation steps or buffer data after carefully reading the information in the Dev. Information box

5) Production Mode, puts the software in a mode where programming a large number of chips is made easy. With the help of insertion test and batch Auto commands, users simply insert and plucking out the chip under the Production Mode. You must first edit the Auto list to suit your needs and then press the Auto function to start operating in production mode.

## 3.2.4 Program the Data in Memory Buffer into a Chip

If the chip package is not DIP encapsulation, user may need to buy the adapter according to the info in the Adapter

Xeltek, Inc.

box. (Contact Xeltek for selection and purchase the adapters.)

Insert the chip properly into the socket; followed by these steps:

- Blank Check, this step can be skipped if the chip is brand new
- Program
- Verify, this step is necessary. The programming cannot be implemented unless passing the Verification
- If the chip to program is not blank, you should add Erase before the Blank Check.
- If encryption is required, add Security or Protect after Verify
- User can choose Auto to perform multiple functions.

## 4 Description of Function

After reading through the Quick Start section, you may want to know more about the process of programming and features available on the software. This chapter encompasses the following,

- menu and tool bar
- select device and device info box
- edit buffer and file info box
- device function screen
- operation info screen
- status bar

### 4.1 menu and tool bar

#### 4.1.1 File

##### 4.1.1.1 Load

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

For most EPROM and SCM, the data type is Data (HEX/ASCII) and for PLD devices it is Fuse.

After device selection, the software will identify the data type automatically. To view the data type for your chip, go to the buffer edit screen. Files are loaded into one of two buffer types, HEX/ASCII buffer (EPROM, MCU etc.) and JEDEC buffer (PLD/PAL).

- For data type Data (HEX/ASCII), select Load menu, the following Load File box will popup.

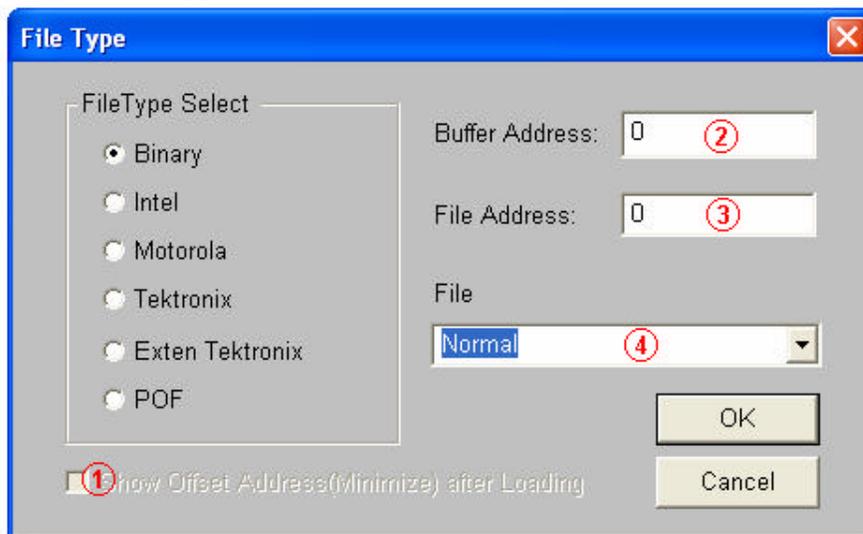


To select a data file to be loaded, enter the path and file name into the name field. If the full path or the exact name of

Xeltek, Inc.

the file is unknown, then a partial path may be entered using wild cards, e.g. '\*.\*' or '\*.bin'.

You need to select the appropriate data type of your data file when the software asks for the file type so that the data can be loaded correctly. The file types include Binary (or POF), Intel HEX, Motorola S record and Tektronix Hex. Refer to the File Type box below (no such box under JEDEC files)



1. If the start address is unknown, choose this option (at Left, Lower screen corner) – Show Offset Address after Loading.
2. Data are filled into buffer from this address.
3. Data loaded into data buffer from this address of the data file.
4. Loading Mode, some files have non-zero file start or file offset address. This should be entered in the file start address for proper data loading. Uncorrected file offset address will cause FF to be stored in the beginning part of the buffer. Uncorrected large offset address may cause data overflow in the buffer and system failure may occur.

Loading Mode:

Normal: all file be loaded

Even: keep the first byte within every two bytes ; drop the second.

Odd: keep the second byte within every two bytes ; drop the first.

Others by analogy.

· As for Fuse, the file type is JEDEC whose extension name is \*.JED. After file selection, it is unnecessary to select the file types. As for Altera chips, these chips use POF files. If user is programming Altera chips, please see Q&A section.

#### 4.1.1.2 Save

This selection will save the current data in the buffer to disk.

For E/EPROM, BPROM or MCU device types, Save File window will popup. Give a location and file name for your data file. Next, the File Type box will popup where you can give your data file a file type. For PLD devices, the Save JED File dialogue box will popup for you to enter a file name.

#### 4.1.1.3 Recent Project

By saving the file path of recent project, it is convenient for user to reopen the projects.

#### 4.1.1.4 Exit

This command closes the programmer software.

## 4.1.2 Buffer

The following selections under the Buffer pull down menu, manages data in the buffer.

#### 4.1.2.1 Edit

The selection brings up Fuse Buffer Edit window if the device buffer type is Fuse. For all other cases, it brings up Data Buffer (HEX/ASCII) Edit window. You may edit the buffer data on the screen with the following keys:

<PageUp> Page up

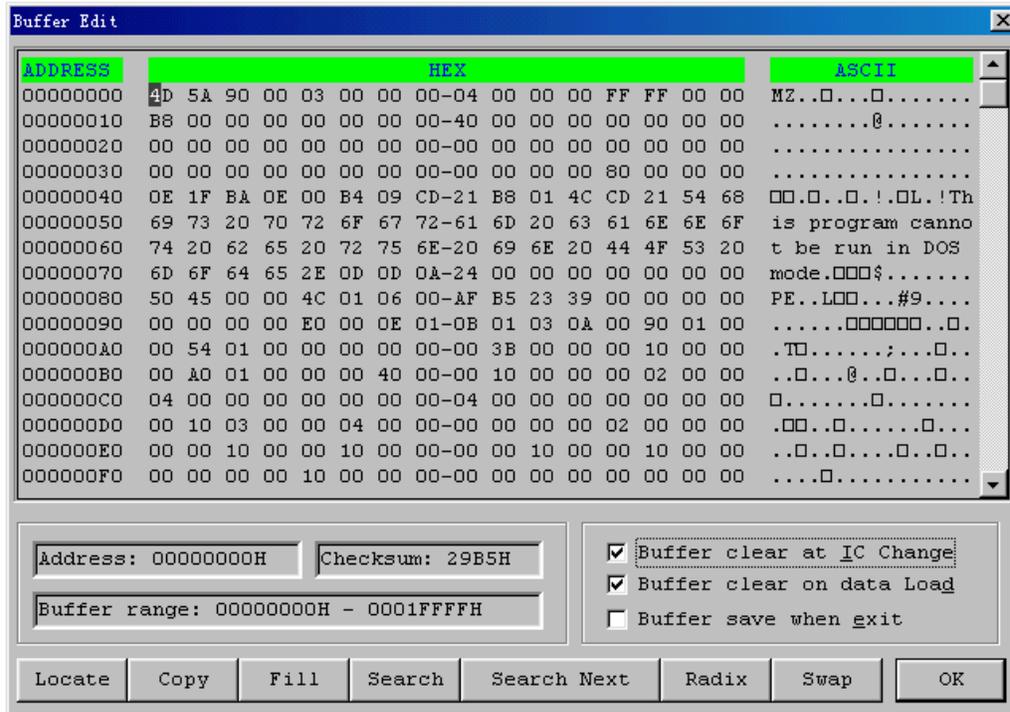
<PageDown> Page down

Xeltek, Inc.

- <Ctrl-PageUp> Moves cursor to the beginning of the buffer
- <Ctrl-PageDown> Moves cursor to the end of the buffer
- <Home> Moves cursor to the beginning of the line
- <End> Moves cursor to the end of the line

#### 4.1.2.1.1 HEX/ASCII data buffer

The HEX/ASCII data buffer is 8-bit wide. TAB key may be used to switch between HEX and ASCII data forediting . According to the rule, if required to enter start and end addresses, the value of start address must be less than that of the end address.



- Locate:  
In Locate Buffer dialog box, enter the address you wish to see displayed and press OK. The cursor will blink at address.
- Fill  
The function will bring up the Fill Data Into Buffer dialogue box. Input the appropriate start address, end Address, and fill data value. When finish press the ok button. For the Fuse Buffer Edit window, fill data values are in binary. For the Data Buffer Edit window, fill data values are in two character HEX code, such as AA, 55, E4
- Copy  
The function displays Copy Buffer dialogue box. It consists of Start Address, End Address, Copy to Address input lines, OK and Cancel buttons. Data between start address and end address will be copied to the buffer beginning with new address.
- Swap  
Swap MSB and LSB byte order for the specified word width in the address range.  
For example, assuming data buffer addresses 0-F (HEX) is:  
12 34 56 78 90 AA BB CC – DD EE FF 11 22 33 44 55  
The word width for selecting is:  
16 Bits (2 bytes), after swapping the data is:  
34 12 78 56 AA 90 CC BB – EE DD 11 FF 33 22 55 44  
32 Bits (4 bytes), after swapping the data is:  
78 56 34 12 CC BB AA 90– 11 FF EE DD 55 44 33 22  
64 Bits (8 bytes), after swapping the data is:  
CC BB AA 90 78 56 34 12 – 55 44 33 22 11 FF EE DD
- Radix  
Toggles between HEX and DEC memory address display.
- Search

Xeltek, Inc.

Searches for a combination of HEX/ASCII codes

- Next

Performs the next search for the search string in Search

#### 4.1.2.1.2 Fuse Buffer

The data for editing is either 0 or 1, which has different meanings depending on the device.

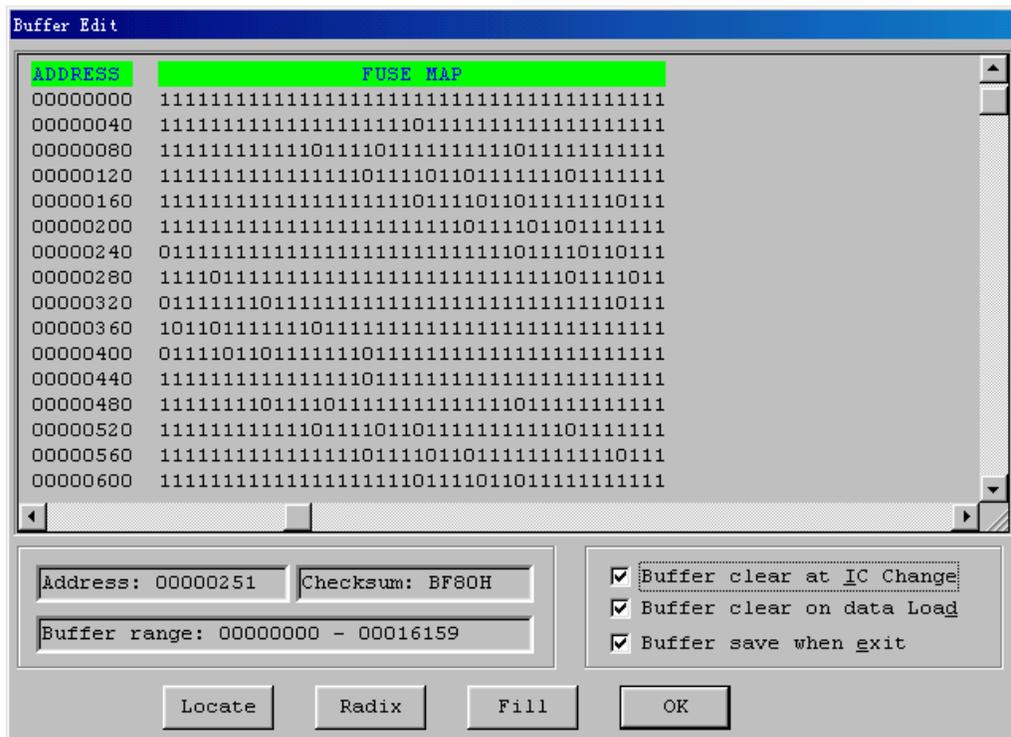
For some devices 1 and 0 can mean,

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

For others,

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

User can define the significations of 0 or 1 according to the devices' manual and JEDEC files.



Note the three important options in the Buffer Edit dialogue box:

- Buffer clear at IC change: clear the buffer after device being selected
- Buffer clear on data load: clear the buffer before load file data
- Buffer save when exit: save buffer data before exiting the software. When entering the software again, the most previous saved buffer data will be loaded automatically.

#### 4.1.2.2 Save Buffer

When the buffer data is saved to a file, the format looks like the following:

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 .....@.....
00000020 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 80 00 00 00 .....
```

Fuse data buffer

```
00000768 10011010101010101010101010101010
00000800 01101010101010101010101010101010
00000832 10100110101010101010101010101010
00000864 10101010011010101010101010101010
```

Xeltek, Inc.

00000896 10101010101001101010101010101010  
00000928 10101010101010100110101010101010  
00000960 10101010101010101001101010101010  
00000992 10101010101010101010100110101010

#### **4.1.2.3 Encryption Table**

The Load Encryption Table, with its two sub-menus, manages an encryption array. The two sub-menus will appear only if the chip selected is equipped with an encryption array.

Load: Brings up the Load Encryption Table dialogue box. Enter the name of the file to be loaded in the input line provided.

Edit: Opens the Encryption Buffer Edit window for viewing and editing.

#### **4.1.2.4 E-Fuse**

If the data type of the device is Fuse and the JEDEC file has an E-field, bring up the E-Field Fuse Data Edit dialogue box, located under the Buffer pull down menu. Please refer to the chapter about the Fuse Buffer.

### **4.1.3 Device**

Before any operation, please select the device for programming, so that the programmer could select the proper algorithm according to the name and manufacturer of your chip.

#### **4.1.3.1 Select Device**

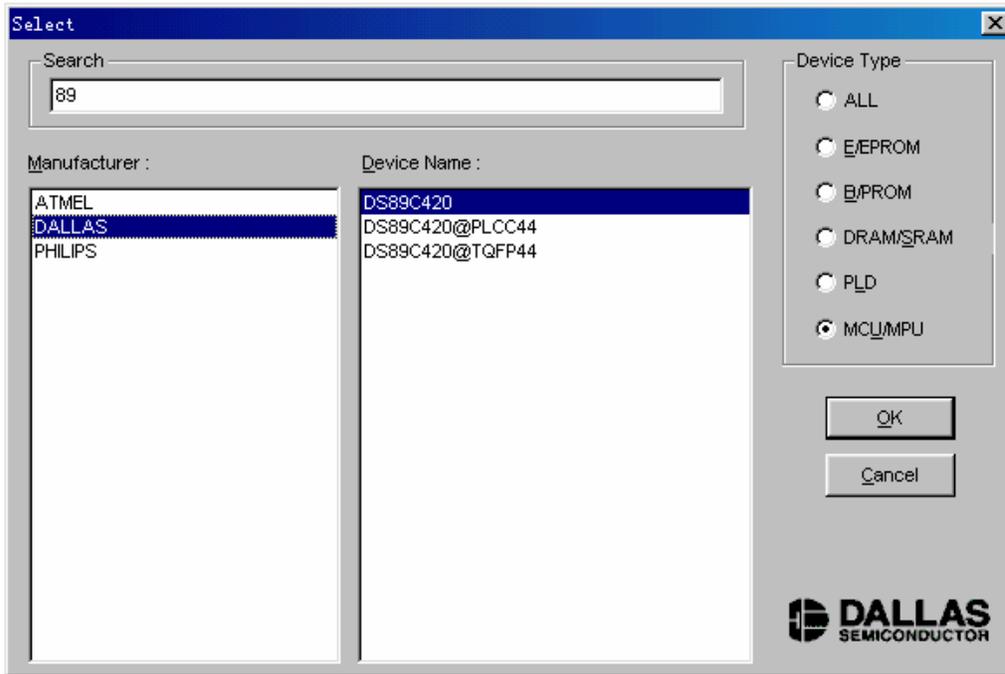
The function brings up Select Device dialogue box. It consists of Manufacturer list viewer, Device Name list viewer, Device Type button and OK, Cancel button, and Search Edit box.

How to select a device?

- Select the device type through the Type Selection button, total five groups: E/EPROM (including EPROM, EEPROM, FLASH), PLD, B/PROM, DRAM/SRAM, MCU. If not sure, please select All.
- Select the manufacture in the Manufacture r column.
- Select the device name through the Device Name column, click OK button.

The part number written on a chip may include letters and numbers to indicate speed, temperature, and encapsulation parameter. These extra letters can be ignored if what is written on the chip is different compared to the Device Name given by the software.

Use of the Search Edit box: Since the large quantity of the devices and difference of the chips' name, there may be some trouble in selecting the device directly. The Search Edit box helps to select. By entering some characters of your chip, will decrease the number chips the software has to search. For example, entering '89' in the Search Edit box all the devices, whether its name or manufacturer, containing '89' will be listed as a result. The search function is sensitive to the sequence of the character string. The character strings such as '819', 'DA8S9' will be listed, but '98' will be ignored.



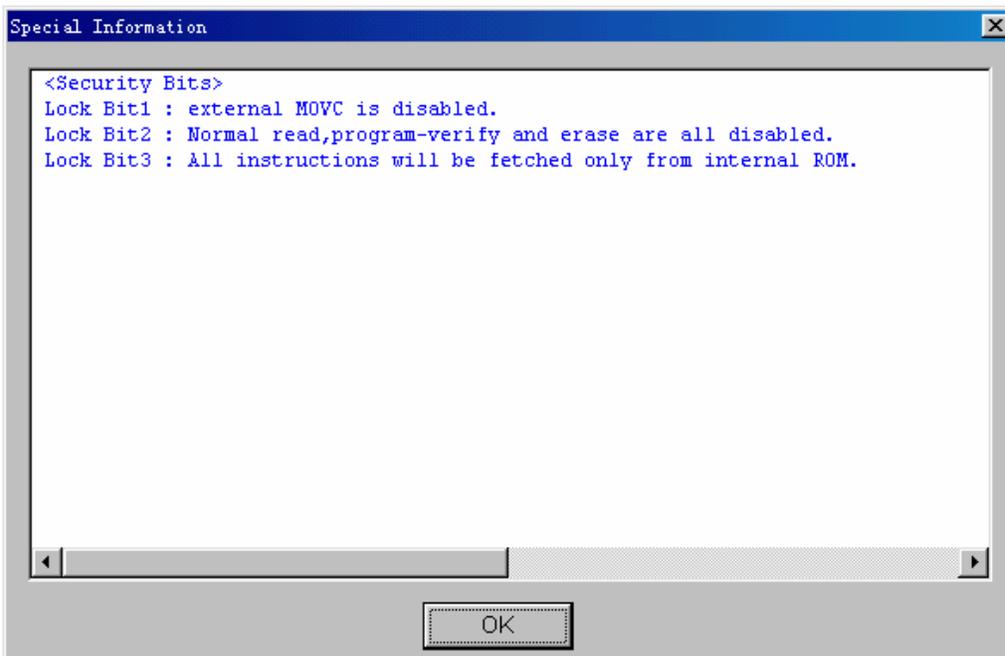
Using Search Edit box reduces the number of devices listed. Users also need to select the encapsulation of the device. For example, as the picture above, DS89C420 represents a DIP chip, DS89C420@PLCC44 represents a PLCC 44-pin chip, and DS89C420@TQFP44 represents a TQFP 44-pin chip.

#### 4.1.3.2 Dev. Information

Some devices have special instructions and conventions, which users need to take in consideration when programming their chip. These instructions may include,

- arrangement of the data buffer content
- explanations of the devices' special operation

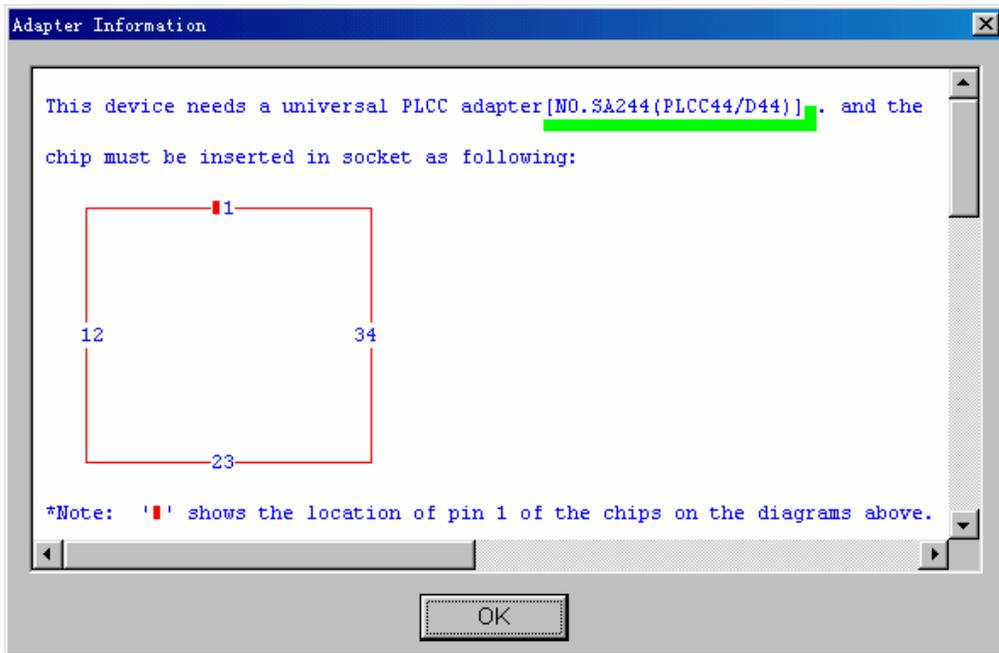
In the popup Special Information dialogue box, user should read the info carefully, and configure properly according to the requests to ensure programming successfully. Refer to the device's manual to find out more about the chip.



#### 4.1.3.3 Adapter Information

Xeltek, Inc.

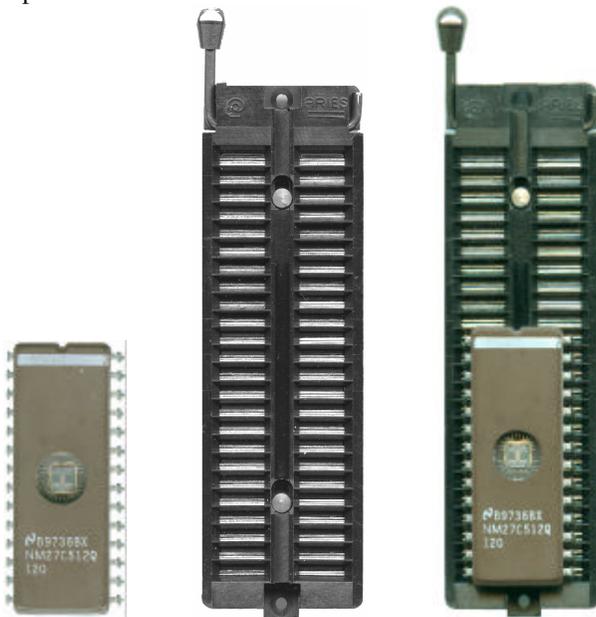
The programmer is equipped with a standard DIP socket. For some device types, users need to use appropriate adapters. The Adapter Information dialogue box including:



- Adapter type such as SA244
- A PEP module may be necessary to program devices which have larger pin count.
- Instruction for insertion
- Applicable chip's size
- Routing layout of the adapter and DIP socket

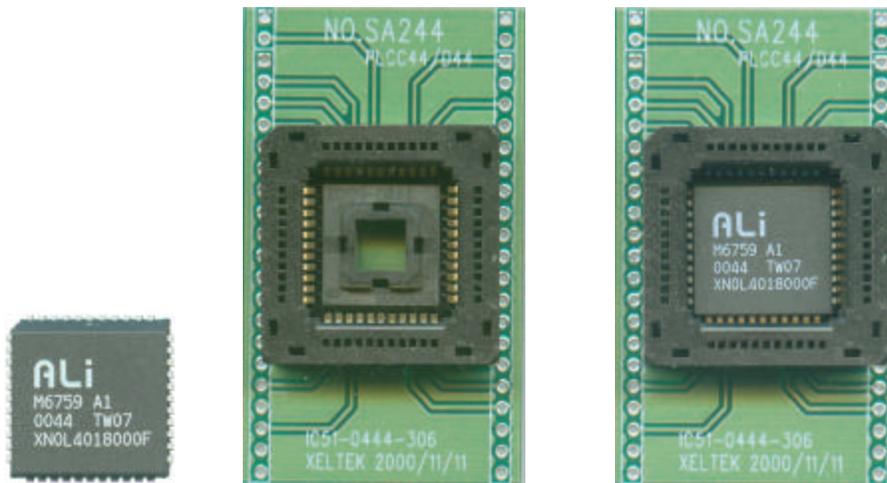
#### Chip Insertion

- DIP: Are inserted bottom justified of the socket with the nick upward. The dia gram below is the process of inserting a 28-pin chip.



- Non-DIP: Socket adapters are available for PLCC, QFP, SOIC etc. Insert the chip according to the instructions in the Adapter Information box. The insertion follows the position of pin 1. It is necessary to identify the pin 1 of the chip. In the Adapter Insertion dialogue box, there is a solid square, which indicates the position of pin 1.

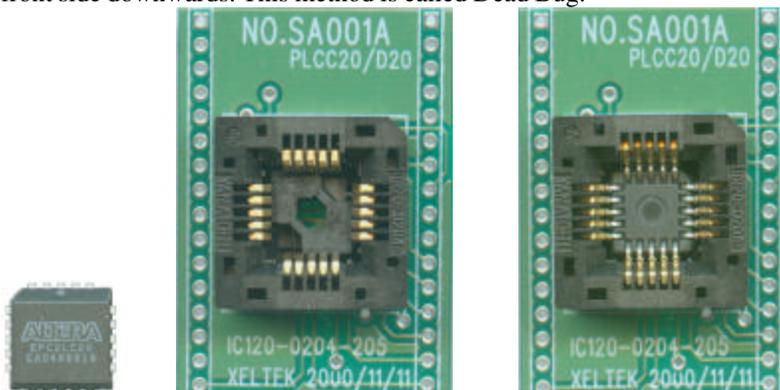
Xeltek, Inc.



· Adapters are inserted in the same way as DIP devices.



Note: usually the front side of the chip is placed upwards; but for PLCC 20-pin chip, it needs to be placed with the front side downwards. This method is called Dead Bug.

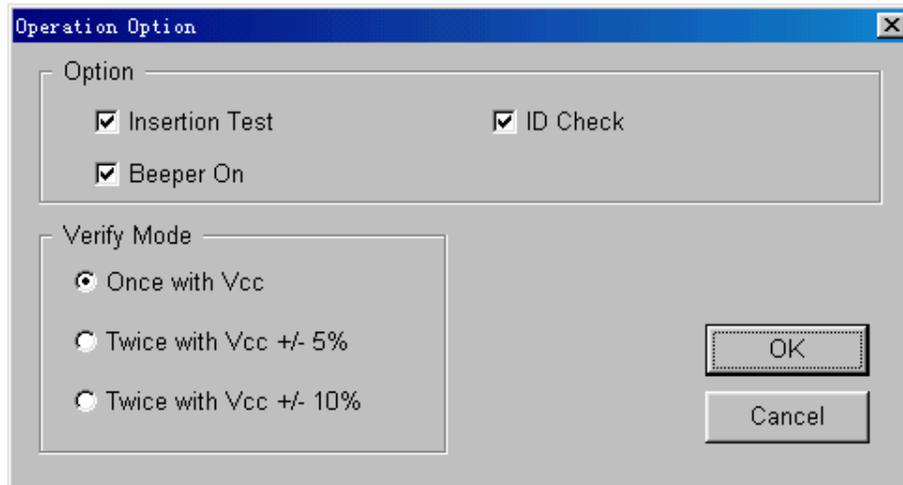


## 4.1.4 Options

This function helps users to program in a specific mode to meet the targeted system requirements.

### 4.1.4.1 Operation Option

This option includes:



- Insertion Test

Only for chips with 48 pins or less (including the chips with more than 48 pins, which are transformed to 48 pins or less by an adapter)

By selecting this option, the programmer will check the pin contact status before programming. The check process includes a check for loose contact, insertion orientation correctness, and more.

An error message will be displayed if there is a failure during the insertion test.

- ID Check

Device ID is the code read from the chip to identify the manufacturer and the device type. This means the device ID will be checked before programming.

An error message will be displayed if there is a failure during the id test.

- Beeper On

During the operation, the beeper in the programmer will sound-off to indicate the result of operation, such as Insertion Test error, ID unmatched, programming successfully or failed etc. This option allows users to turn on or off the beeper.

- Verify Mode

It is necessary to verify after the programming of a chip. VCC voltage for verification is different for each chip.

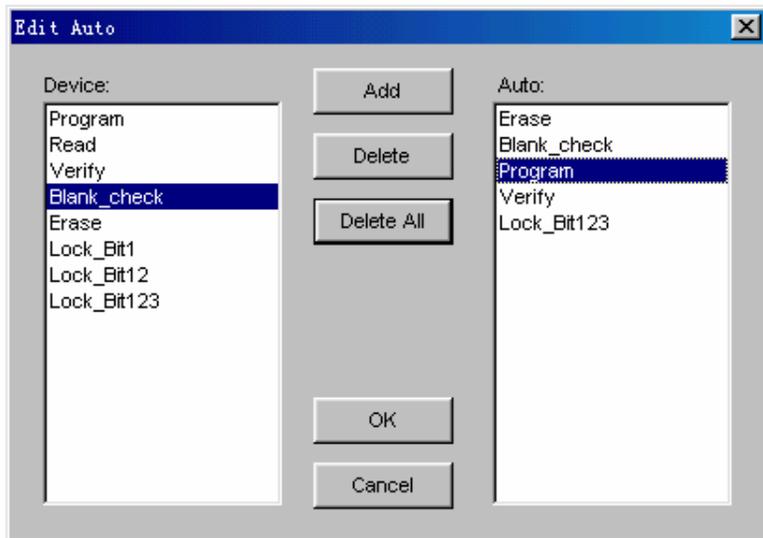
1. Use VCC (+/- 5%) or (+/- 10%) to verify. For example, VCC=5.00V. Use VCC=5.00V to verify once, or use VCC=4.74V and VCC=5.25V to verify twice (+/- 5%), or use VCC=4.50V and VCC=5.50V to verify twice (+/- 10%).

2. Use MinVcc and MaxVcc to verify.

Note: The Operation Option will vary with the various devices. i.e. Most SCMs don't allow user to set the start and end address of the device.

### 4.1.4.2 Edit Auto

In the Edit Auto dialogue box, all devices have an Auto option, which is like a batch command. This function allows the user to perform multiple device functions at once. For example, select ATMEL AT89C51, open Edit Auto dialogue box. On the left of the box is Device Functions column in which all device operations are displayed. On the right of the box is Auto column in which operations needed and operation sequence are displayed.



In the diagram above, the operation sequence is Erase, Blank\_check, Program, Verify, Lock\_Bit124.

Add button: add the functions in the left column to the right Auto column

Delete button: remove the functions from the Auto column

Delete All button: remove all the functions from the Auto column

Double click on the functions listed in the left column to add them to the Auto sequence.

Double click on the functions listed in the Auto sequence to remove them.

Once the Auto menu has been edited, the auto sequence will be saved in the hard disk.

This option helps to improve the efficiency of programming.

#### 4.1.4.3 Device Config

Some SCMs allow users to configure special modes of the chip. Such as memory block mapping, watchdog timer, clock, and encryption. In the picture below, is the ConfigWord of DALLAS DS89C420, which has watchdog and three encryption modes. Some SCMs have some ConfigWord that its option may be divided into 2 or more pages. Users need to configure every page before the programming. The configuration of the ConfigWord includes 8-bit edit line, 16-bit edit line, radio box and check box.

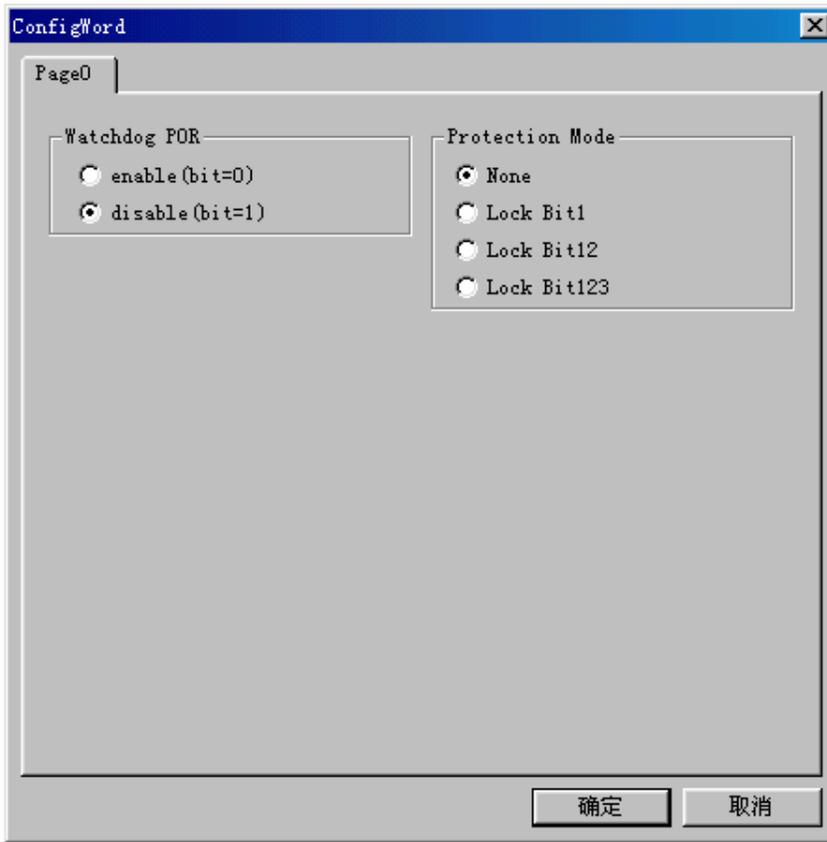
The ConfigWord program includes,

1. If there is encryption option in the ConfigWord, the verify operation cannot be executed; therefore the software will disable encryption automatically and write the other options in the ConfigWord into the chip during the programming. Users need to use Operation Option Security and Protect to write the encryption option. If you select None in the encryption option, the Security and Protect is unavailable.

2. Since some ConfigWord cannot be written separately, they will not be written during the programming. At this point the relevant Operation Option is required, such as Write\_Option.

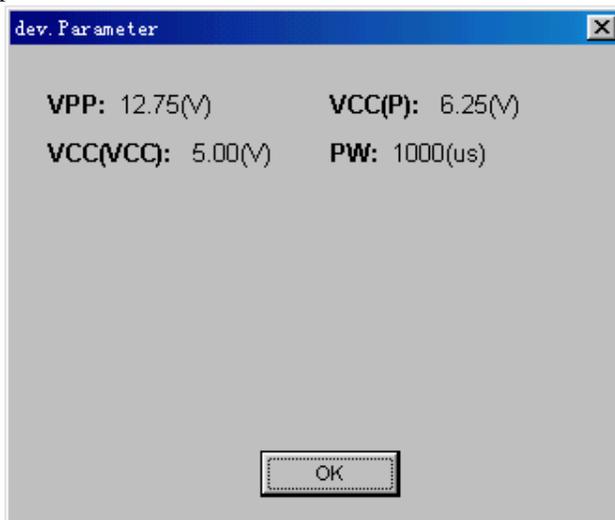
Different devices require different ConfigWord. Reference is in the devices' manual.

Some FLASH devices also use ConfigWord to execute Protect function and displaysegment protect.



#### 4.1.4.4 Parameter

This function displays vital parameter of the chip, including VCC, time and retry times after failure. These parameters cannot be revised.



#### 4.1.4.5 Production Mode

This function is used to program devices in large numbers. With the Insertion Test, the programmer detects the chip's insertion and carries out automatically the execution of the auto operation sequence.

If a nonsupport window pops up, that means the device does not support production mode.

Advice: First select the device and edit the auto menu. After tests, select production mode, and execute auto program, according to the info displayed in the "Operation Message".

	Message	Success	Failure
1	VERIFY OK! <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">1</span> Get chip off Pls	1	0
2	Waiting... <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">2</span> Put chip in Pls	0	0
3	PROGRAMMING... <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">3</span> 0600	1	0
4	Waiting... Put chip in Pls	0	0
Total:		2	0
<input type="button" value="Reset"/>			

1. **Get chip off Pls**: program finished and wait for chip off, please take the chip off. At this point, check the result of programming, whether success or failure, according to the info in “Message”.
2. **Put chip in Pls**: wait for chip insertion, insert the chip into the socket properly.
3. **PROGRAMMING...**: wait finishing.

Asynchronous programming mode: every module programs the chip independently, user can get off the chip and put in another chip when programming finishes in this module, is independent of the other modules.  
To end the production mode, please click Cancel button.

## 4.1.5 Project

A Project file encompasses the following:

- current device’s info, such as manufacturer name, type, pin info, ConfigWord, current buffer data and data file name.(Since the buffer data can be revised, the buffer data may differ from the data file therefore project file must be resaved to include changes made on buffer data.)
- all Operation Option settings

Auto program sequence

Project records all the preparation works before programming. Users can return to the previous environment by opening the project. In this way, users can avoid error during the work,

Note: The software update and reinstallation to another folder will make the project saved previously become unavailable.

### 4.1.5.1 Load Project

Load project by selecting the file in the File dialogue box. According to the file, change device, buffer data and operation option etc.

### 4.1.5.2 Save Project

Save the current work environment to specified project.

### 4.1.5.3 Download Project

Load project and download to CF card. Information stored in CF card enables users to operate in stand-alone mode. In stand-alone users can scroll through a list of saved project files to program their chip.

## 4.1.6 Tool Bar

The tool bar provides the buttons of functions in common use.

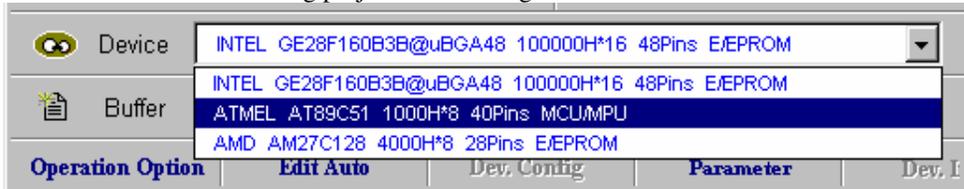


## 4.2 Device Selection Bar

The recent selected device names are saved in the bar. (Also including manufacturers, size, pin info, and types)



By using this bar, users can change the device selection, including manufacturer, device name and buffer capacity, etc. It's different from Loading project file to change the work status.



The following box will popup after selection.  
Select ATMEL AT89C51 Continue?

## 4.3 Edit buffer and File Info

This bar quickly launches the Buffer Edit and displays the file loaded recently.



The file info displayed includes history, which allows user to select the file again.  
Since the device selected may be different, the event of data lost, file format unmatched may occur.



The following box will popup after selection.

Reload the file E:\test.bin in the buffer. Continue?

## 4.4 Special Device Info and Operation Option Bar

Before the programming, users should obtain the special device info and set the operation option properly according to the targeted system's request. The toolbar below will facilitate user to finish this work.

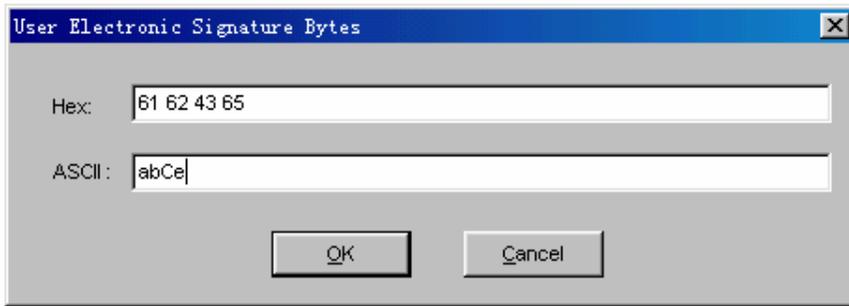


For detailed info, please refer to 4.1.3 and 4.1.4

If the device selected has a U-Field, the Operation Option Bar will become as below

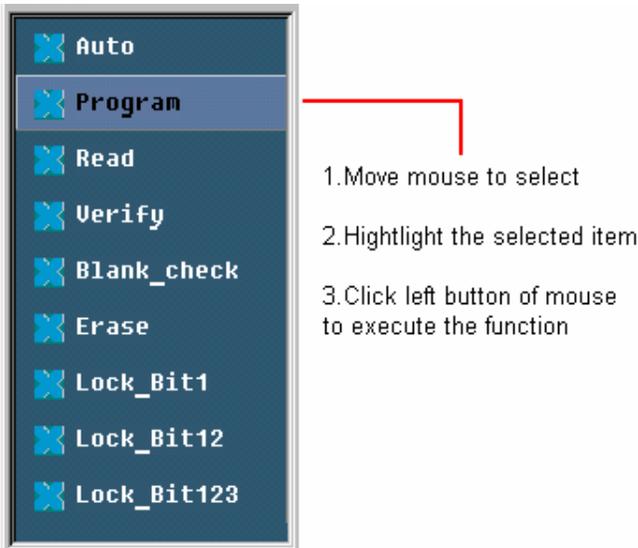


U.E.S.: Edit U.E.S. This content may be changed during reading device or Loading files. Users can view and edit U.E.S. (two display methods, Hex and ASCII)



## 4.5 Device Operation Window

Each chip has different available functions for programming. Listed programming functions are based on the chips features. Users can execute the operation by click the icon in the window. Following is the window for Atmel At89c51.



Frequently used operations are explained below.

**Auto:** The function executes a series of operation in sequence. Please refer to 4.1.4.2

**Program:** The function writes the data from the buffer into the chip. The Verify function is performed in two modes. One mode is verifying while programming. This method ensures the correctness of programming. If an error occurs, the error message will be displayed and the process will be ceased. The other method is not verifying during the programming. Even if an error occurs, the process will not be ceased. When the last address is written, the program successful message will be displayed. Anyway, the program successful message only means that the program process is finished, the correctness is dependent on the result of verification.

**Read:** The function reads the content of the chip into the buffer. To view the content read, please open the Buffer Edit dialogue box.

**Note:**

- 1) For the encrypted chips, their content cannot be read
- 2) For the chips with ConfigWord, the ConfigWord will be read and save to ConfigWord buffer.
- 3) For PLD devices, their test vector is not in the chip, thus reading it is unavailable.

**Verify:** The function compares the content of the buffer to that of the chip. If there is any discrepancy, verification will cease and failure message will be displayed. For most memory chips and SCM chips, the failure message includes address and data. For PLD chip, the message displayed is dependent on the programming reference. Only the chips that pass verification can be recognized as chips being programmed correctly.

**Blank check:**

This function checks if the device is in blank state, which is dependent on the data manual. If selecting fill data

Xeltek, Inc.

buffer blank as devices changed, the programmer will fill all the HEX/ASCII buffer data with FF or 00 (Hex), and Fuse buffer data with 1 or 0 after the devices selected. An encrypted chip is displayed as a blank chip. EEPROM does not need Bank\_check.

How to write over a chip into blank?

- 1) For chips that can be erased electrically, execute Erase
- 2) For chips that can be erased with ultraviolet ray, use the ultraviolet ray eraser for 3-30 minutes.
- 3) For OTP, once written it is unable to be written over back to blank again.

Erase:

Erase the chip to blank state. This operation can only be executed with the chip that can be erased electrically. EEPROM doesn't need to execute this operation. If you need a chip of blank state you can directly program the blank data to the chip. Some chips that can be erased electrically are able to become an OTP by configure the ConfigWord. Please refer to the device data manual.

Protect:

- 1) Security the data to make them unreadable. Whether the protect state is readable or not, affect the editing of the operation function. When the protect state is unreadable, the Protect operation will be added to the operation function window. As long as the Protect function is executed, the chip will be protected. If the protect state is readable, the state None and all the protect options will be listed out for user to select. The protect operation will be executed according to the ConfigWord. If user select None, the protect operation is non-functional.

Note: Some manufacturers' chips with window will be used again after full-protect, and ultraviolet ray eraser is also unable to erase.

- 2) For FLASH chips, this function is for protecting the data in the chips. The protected chip is not re-writable before the protection removed. The Erase operation would remove the protection.

This operation is required to be used with Dev. Config. Users need to identify the Sector to protect. There are two ways to select the Sector,

1. Users enter the address of several (max 8) sectors to decide the sectors to protect. After the protect operation, the sectors whose addresses are entered will be protected.

	From:	To:
1.	00000000	00000000
2.	00000000	00000000
3.	00000000	00000000
4.	00000000	00000000
5.	00000000	00000000
6.	00000000	00000000
7.	00000000	00000000
8.	00000000	00000000

2. Divide the chip into several sector groups, users can only choose these groups to protect.

Security: equal to Protect,

Erase\_All: equal to Erase

Protect\_All: equal to Protect

Lock\_Bit: the notion for protects. Please refer to the manual.

Above are commonly used functions. For some devices, which have special requirements and functions, the operation function window may be different.

For detailed information, user could refer to:

1. The manual of the device
2. Dev. Information