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

Note: In order to connect the programmer easily and quickly, please connect the hardware after installing the software.

1. General Description

1.1 INTRODUCTION

1.1.1 What is Superpro?

The SUPERPRO is a series of cost-effective, reliable, and high-speed universal programmers. They are designed to communicate through a USB or parallel port (model dependent) and to operate with most IBM-compatible desktop computers and notebook computers. Their menu-driven software interface makes them easy to operate.

Programming hardware includes the following items:

·A programming module (including a 40 or 48 pin ZIF)

·An AC adapter w/ dual range switching power supply (model dependent)

·A USB or parallel connecting cable

·Optional socket adapters to accommodate PLCC, TSOP, SOIC, SOP, QFP, TSSOP and BGA package

types are available.

·Software (CD)

Software includes the following features:

·Supports Windows 95/NT/2000/XP

·Programming support for a large number of devices (1500-11000) including PROMs, E/EPROMs, PLDs and MCU from more than 100 manufacturers (model dependent).

·Support formats in 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 loose contacted pin (model dependent)

·Integrated full screen buffer editors with commands for fill, copy, move, swap, etc.

·Auto-generation of electronic serial numbers (model dependent)

·Project and selecting history

1.1.2 Manual Organization

This booklet consists of five main chapters:

Chapter 1: Provides an introduction to the SUPERPRO series, including system requirements.

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

Chapter 3: Provides pre-programming guidance and device selection, loading and reading data.

Chapter 4: Provides explanation for each software command and its functions.

Chapter 5: Provides 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>

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) or one parallel port (model dependent)

·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 the following:

·A programmer module

·A parallel or USB connecting cable (model dependent)

·An AC adapter or a dual range switching power supply (model dependent)

·The installation software CD

·User’s Guide

·Registration card

2. System Setup

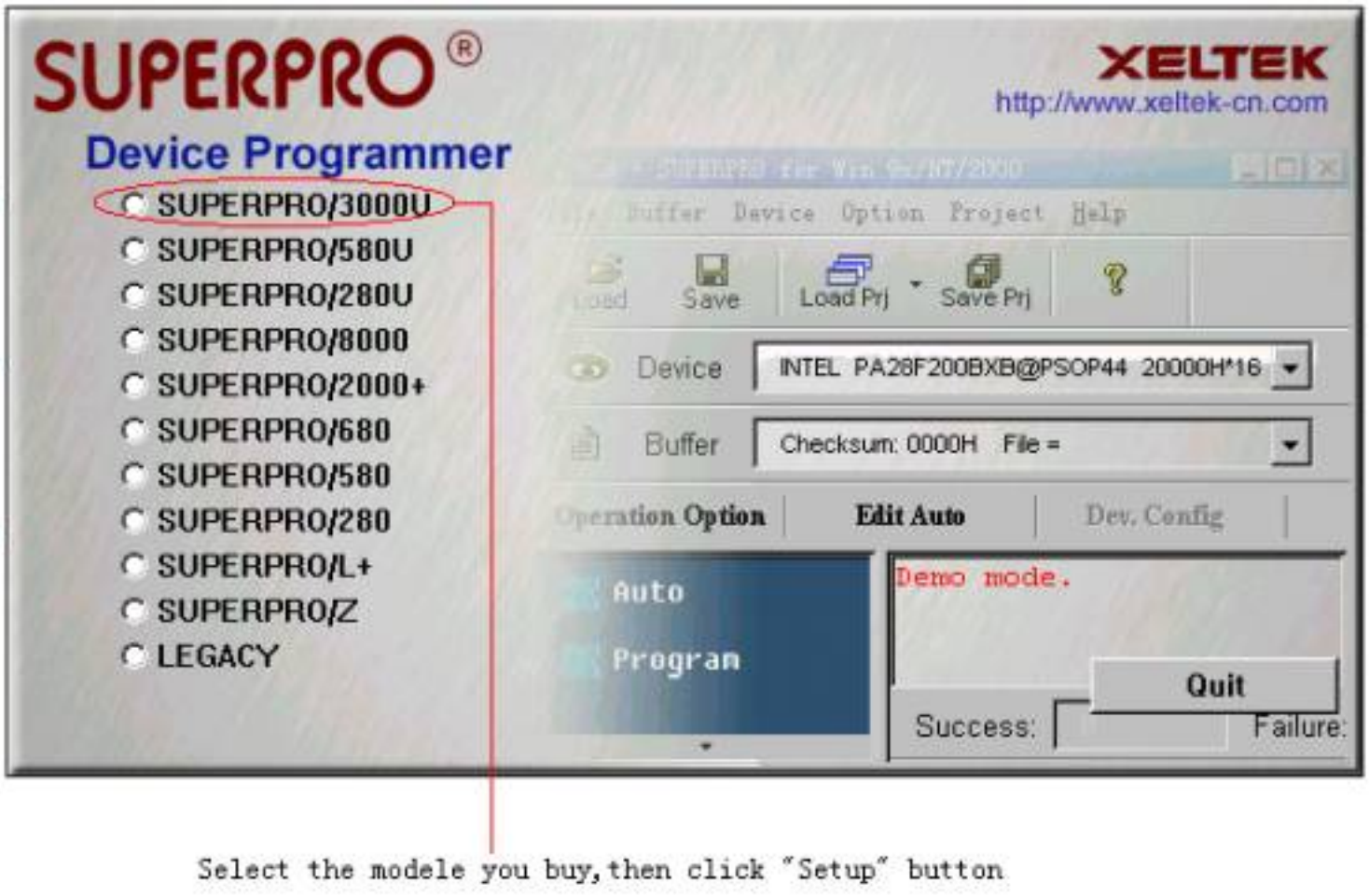
Parallel-based programmer’s software installation is straightforward and self-explanatory. If it’s your first time to use the USB-based programmer of XELTEK, this chapter will help you to properly install the software and connect the hardware. USB devices are PnP devices. At first installation, Windows 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

The content in 2.1.1, 2.1.2, 2.1.3, and 2.1.5 is applicable for both USB-based and Parallel-based programmers.

2.1.1 Setup from CD-ROM

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



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

2.1.2 Download software from the Internet

You can download the specific software for a certain model at Xeltek website. It is usually a self-extracting file. You only need to run this file to setup the software.

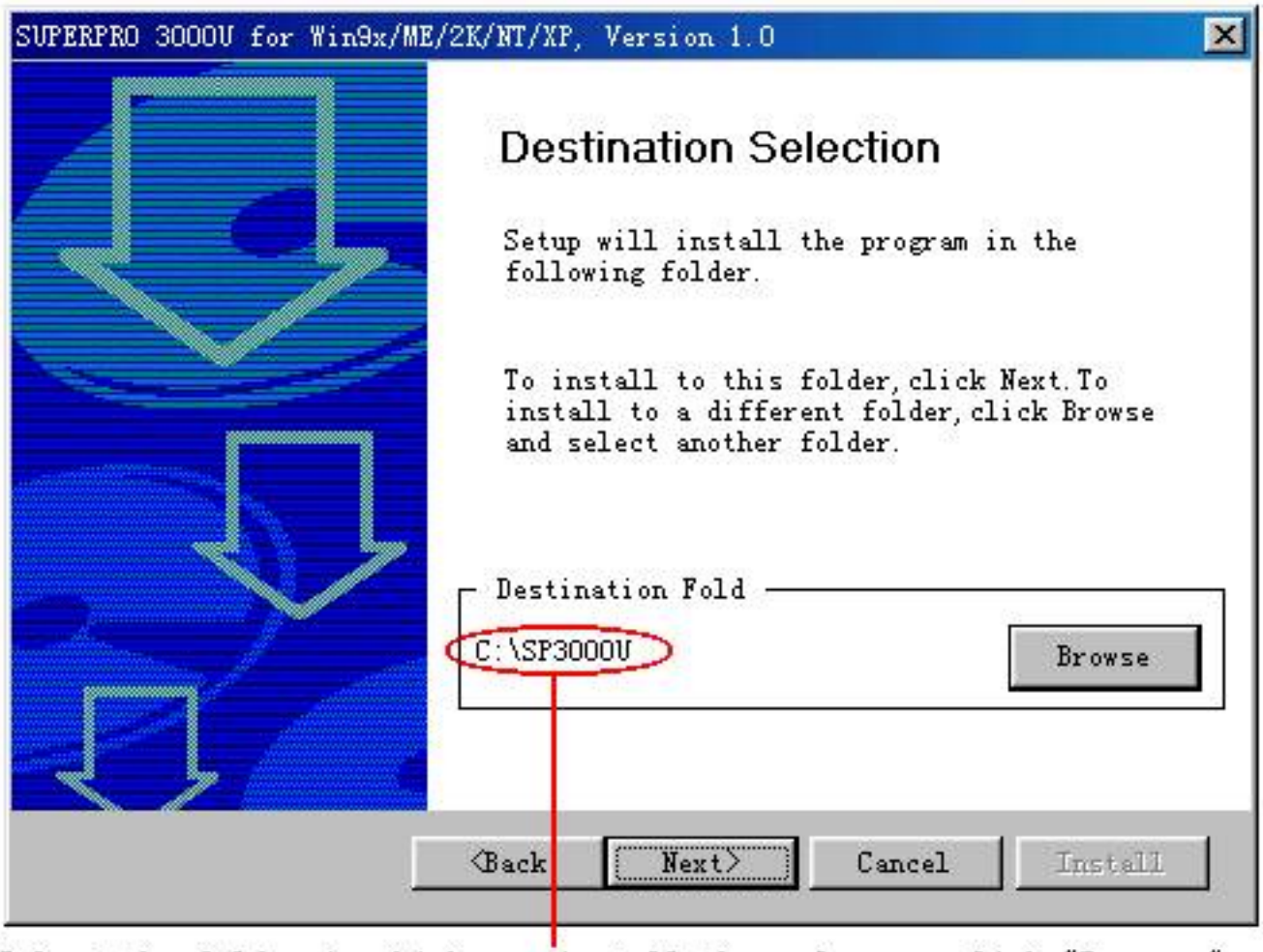
2.1.3 Setup Process

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.

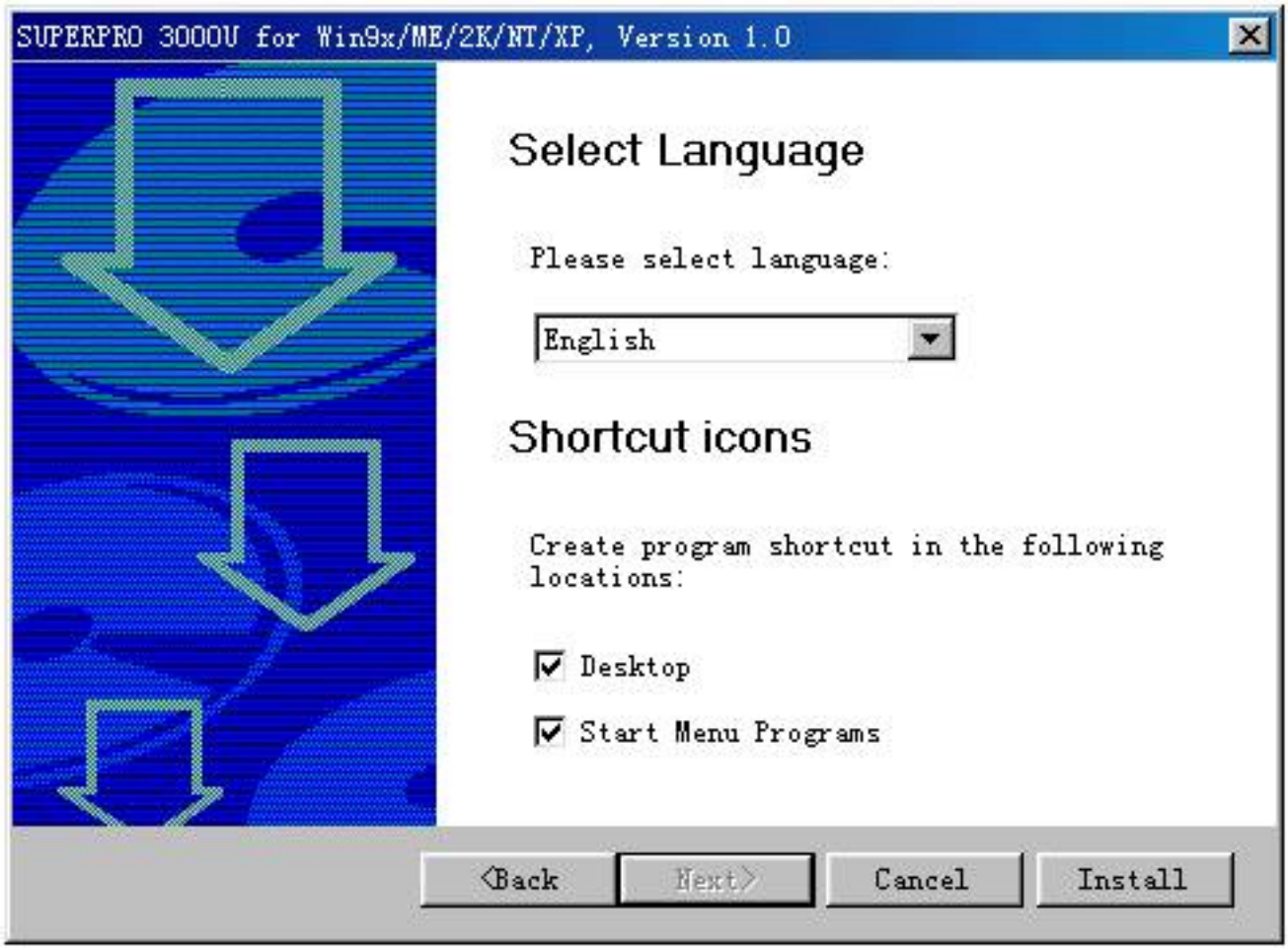
Click “next” button

Second, select destination fold



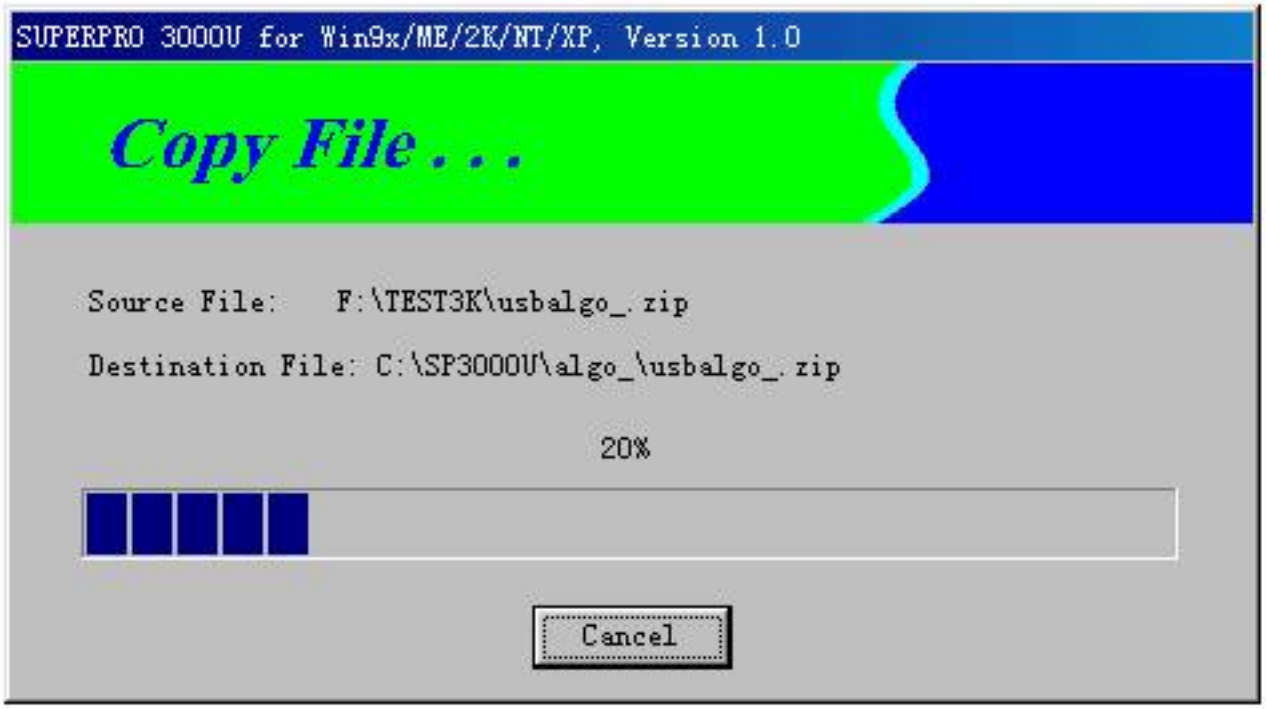
Select the folder in which you install the software, click "Browser" button to change the default folder. Then click "Next" button to go on.

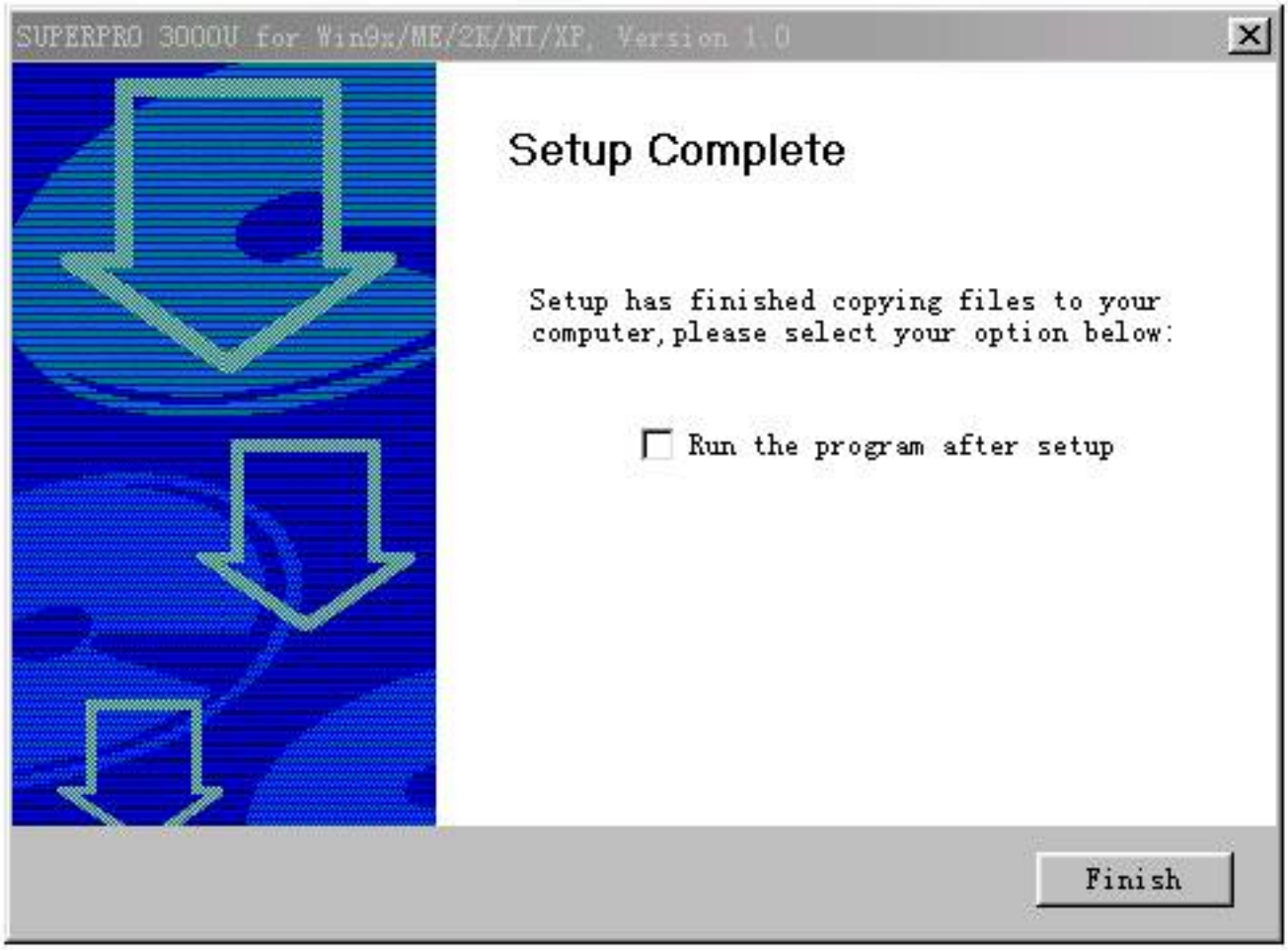
Third, as following



Click “Install” button

Fourth, copying files please wait





Fifth, click “Finish” button to finish the setup.

2.1.4 Hardware Setup

Connect the programmer module to the computer's USB port or parallel port. Turn on the programmer power switch.

The driver installation is very simple, only need to wait until new hardware wizard finish installation during the installation, please don't run any other program.

2.1.5 Running the Program

The program for the programmer has a standard Windows UI, including pull down menus and buttons, etc. User should master MS Windows OS know-how and use of pointing device such as a mouse or trackball.

As the program started, it will communicate with the programmer immediately and begin initialization.

If communication error occurred,

Please check the steps below:

·Connect the programmer to PC, and turn on the power switch

·Correct installation. If the programmer is connected before the software installation, PC will detect new hardware. The window appears as follows:

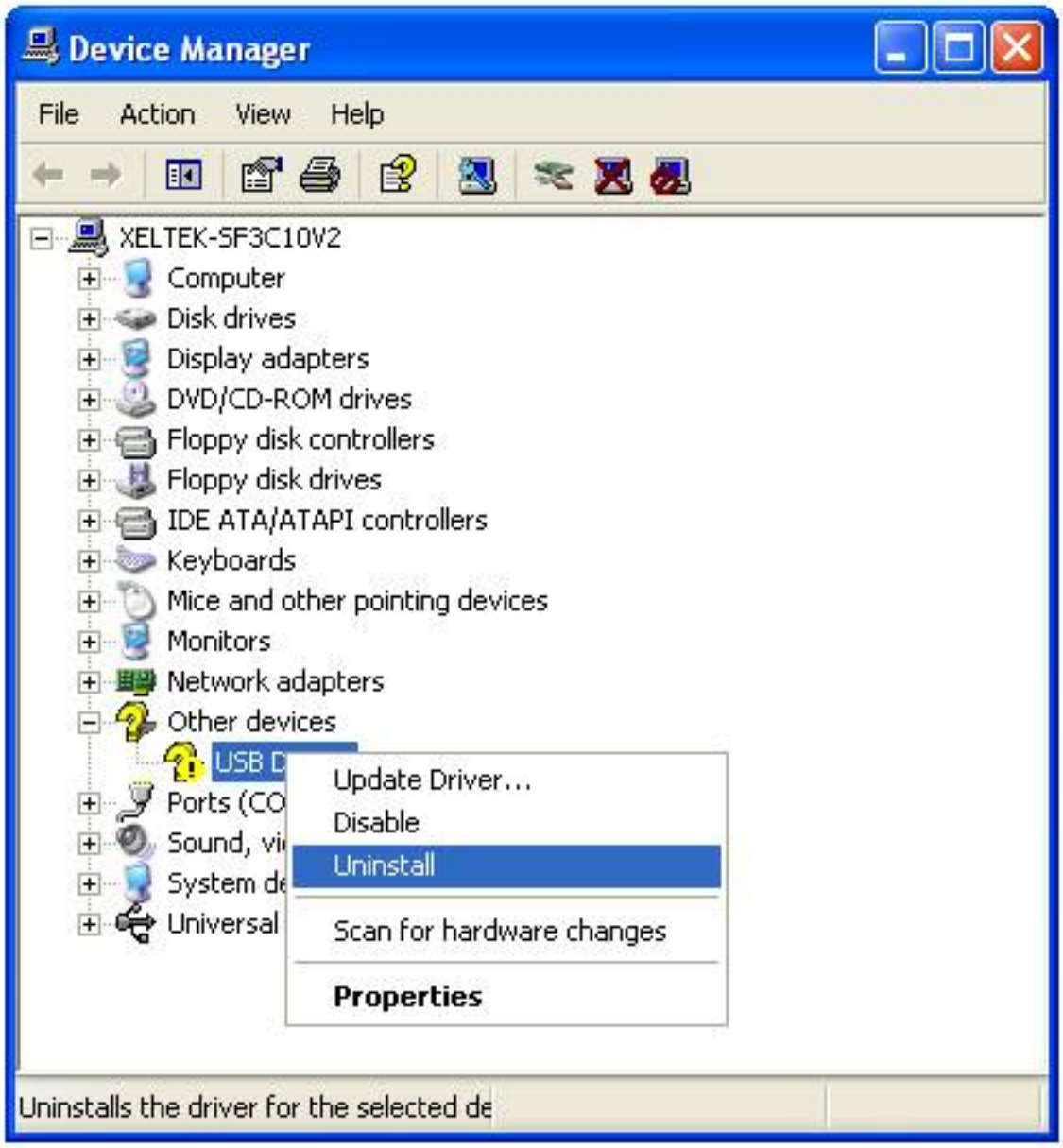
Please click “Cancel” to avoid troubles.

2.2 How to resolve the communication error

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

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, open ‘System’,click ‘Hardware’ property), 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 Guide

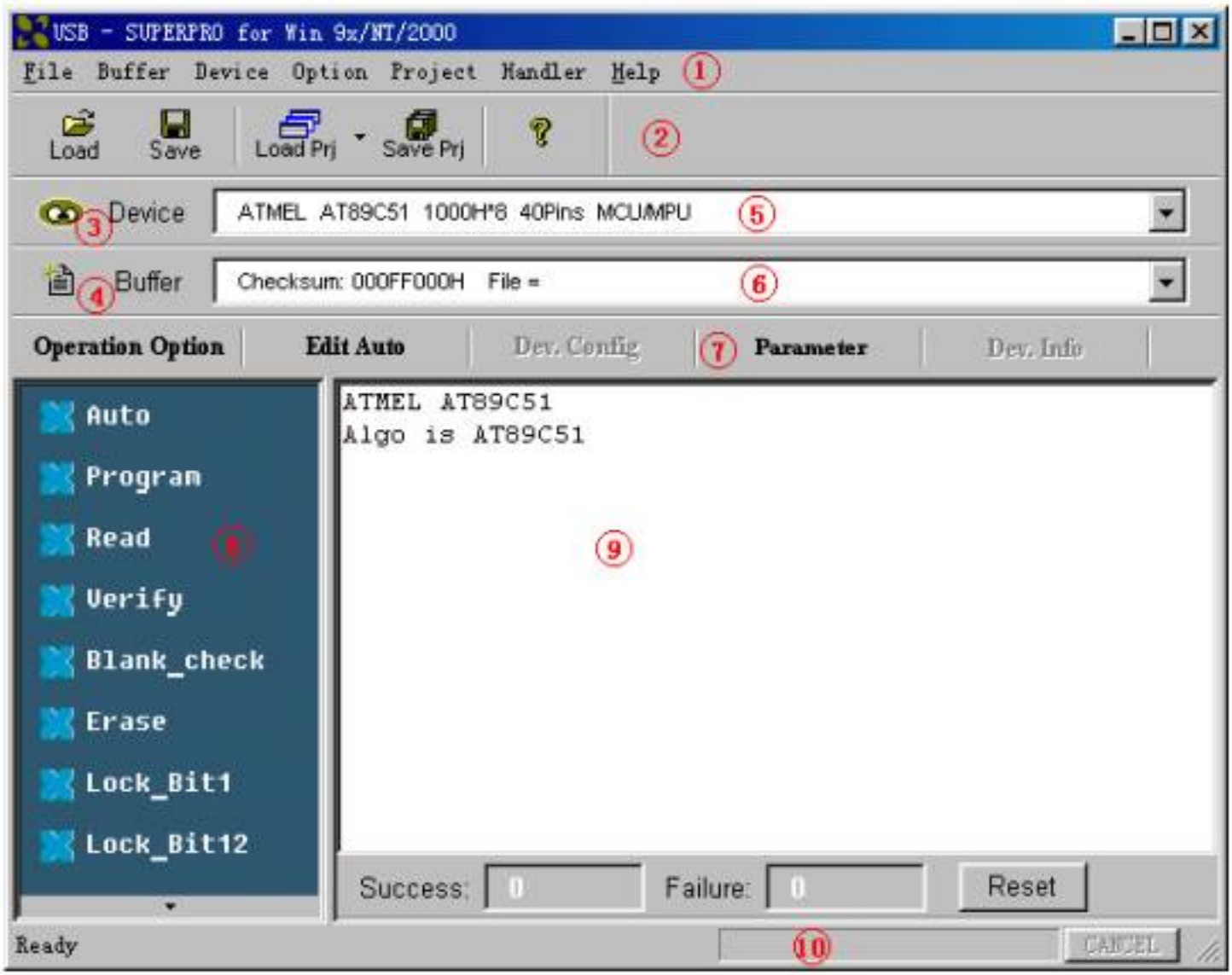
This chapter helps users understand the whole process of IC program. The content includes:

·The UI of the software

·The steps of device programming

3.1 Description of Interface

The interface is as following:



1. Main Menu 2. Toolbar 3. Select Device
4. Edit Buffer 5. Device Messages 6. File Message
7. Operation Toolbar 8. Device Function
9. Operation Message 10. Process Bar

3.2 The process of the programming

3.2.0 Hardware Preparation

First, make sure the programmer is properly connected with the PC, and communication established successfully with the PC. (Some SMD devices need adapters, please contact Xeltek or its dealers for your adapter needs.)

Second, insert the chip correctly. (Instruction for SMD and non-standard insertion will appear on the screen when the selection of the device is made, if no instruction appears, chip insertion orientation should follow the guideline of the chips marking near the pin-driver socket.)

3.2.1 Select the device

Click on “Select Device” button or click on “Device”, select the device for programming from the main menu. The Select windows will popup. First select the Device Type (E/EPROM, BPROM, 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. Typing in the device part number in the Search box may also be used to search chips.

3.2.2 Load data into Buffer

Writing the data into memory buffer of the chip basically does chip programming. Therefore, to load the data, you may load it from disk or copy master data by reading data from the master chip.

Loading a file

You may load a file into the memory buffer by clicking File menu in the main screen and select Load File. In the Look In dialog box, select the folder and file name you wish to load. In the File Type box, select the relevant data type. 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.

Reading data from master chip:

Insert the master chip in the socket and make the selection of the device to read from. In the Function screen, click on Read, which copies 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: some devices have no read function or are encrypted, under this condition data cannot be read from.

3.2.3 Options

1) Operation Option, including:

·Insertion Test, check the pin contact before programming

·ID Check, check device's ID before programming

·Beeper On, beeper sound off when the operation succeed or fail

· Auto Increment, when programming add a number label to the designated place so that each chip has a different label

·change the start and end address of programming zone of the device

·Verify Mode, according to the requests, select specific VCC voltage verification in order to check the programming chip correctly

2) Edit Auto, in the Function Screen, every device has a basic Auto operation, which automatically execute the operation of device step by step. Normally select the following programming steps,

·Erase

·Blank-_check

·Program

·Verify

·Security or Protect

3) Dev. Config, as for the chip requires configuration, it must be properly configured 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 read the text in the Dev. Information box

5) Production Mode, in order to program chip in large number, with the help of insertion test, users do not need to use keyboard and mouse repeatedly, only simply inserting and plucking out the chip under the Production Mode.

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

Insert the chip properly into the socket; following 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 finish all operation by one step.

4 Description of Function

After reading the Quick Guide, you may want to know the specific process of programming. This chapter will tell you all the function of the programmer.

·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

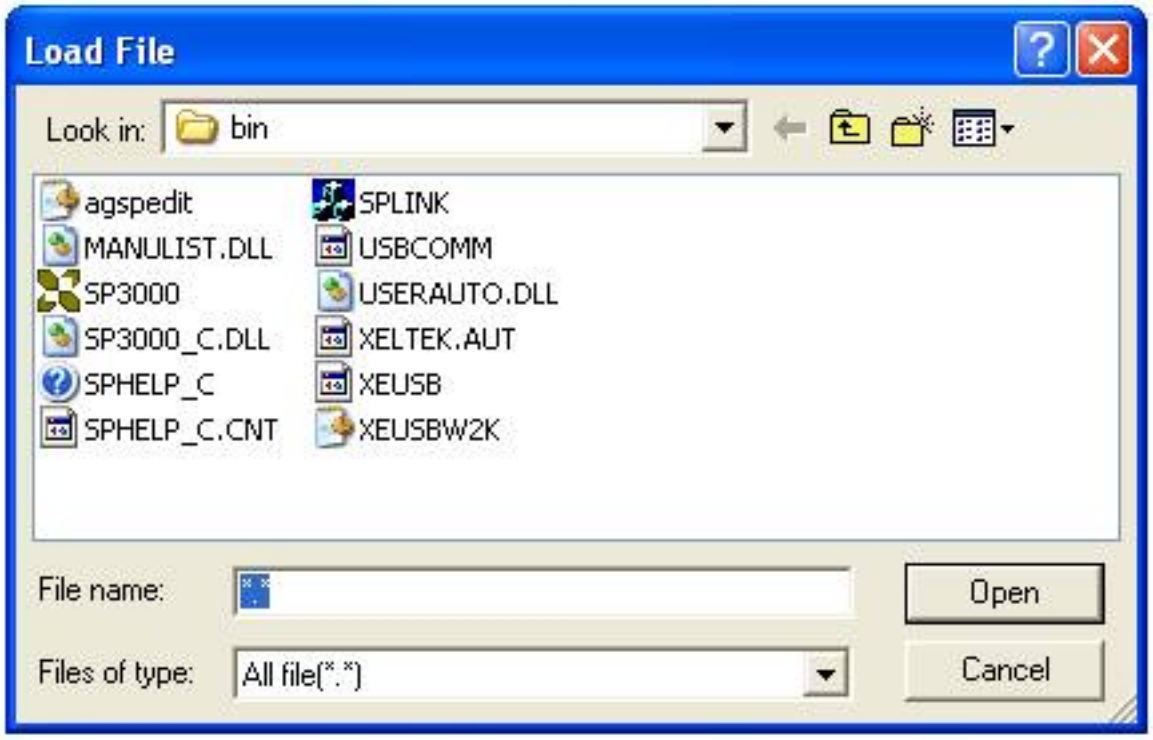
The two data types of device are Data (HEX/ASCII) and Fuse

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

After device selection, the software will identify the data type automatically, to see the data type by opening the buffer edit box

Files are loaded into one of two buffer types, HEX/ASCII buffer (EPROM, MCU etc.) and JEDEC buffer (PLD/PAL).

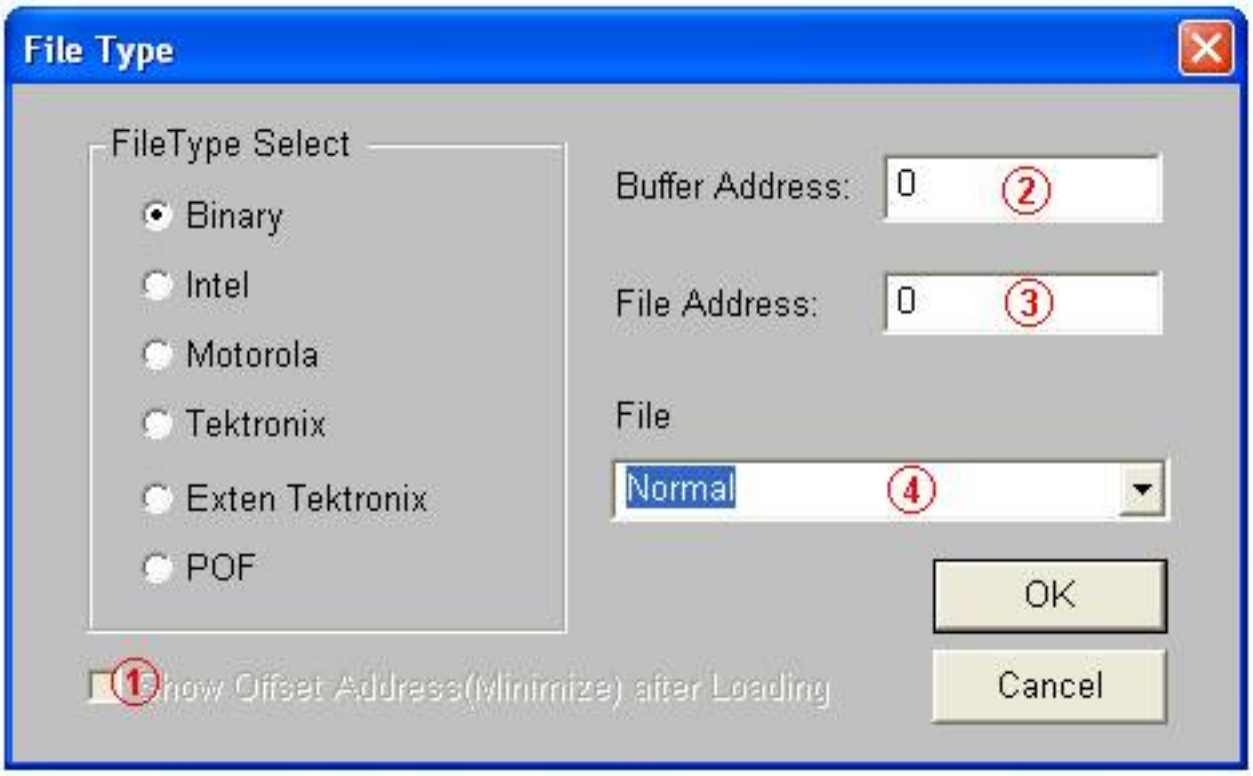
· For data type Data (HEX/ASCII), select Load menu, the Load File box popup.



To select a data file to be loaded, the path and the file name should be entered into the name field. If the full path or the exact name of the file is unknown, then a partial path may be entered using wild cards, e.g. '*.*' or '*.bin'.

Based on the saving mode, the files are divided into various formats. You need to select the relevant data types after selecting the file so that the data can be loaded correctly. The file types include Binary (or POF), Intel HEX, Motorola S record and Tektronix Hex.

File Type box as following (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

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 FFs 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 Altera Company uses POF files, if user is programming the chips from Altera Company, please see Q&A section in the chapter.

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. Select the folder and filename to be saved under. Next, File Type dialog box will popup for selecting file type to be saved under.

For PLD devices, the Save JED File dialogue box will popup for entering the 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 and returns you to the control of the operating system.

4.1.2 Buffer

The 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; otherwise, 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

<Ctrl-PageUp> Move cursor to the beginning of the buffer

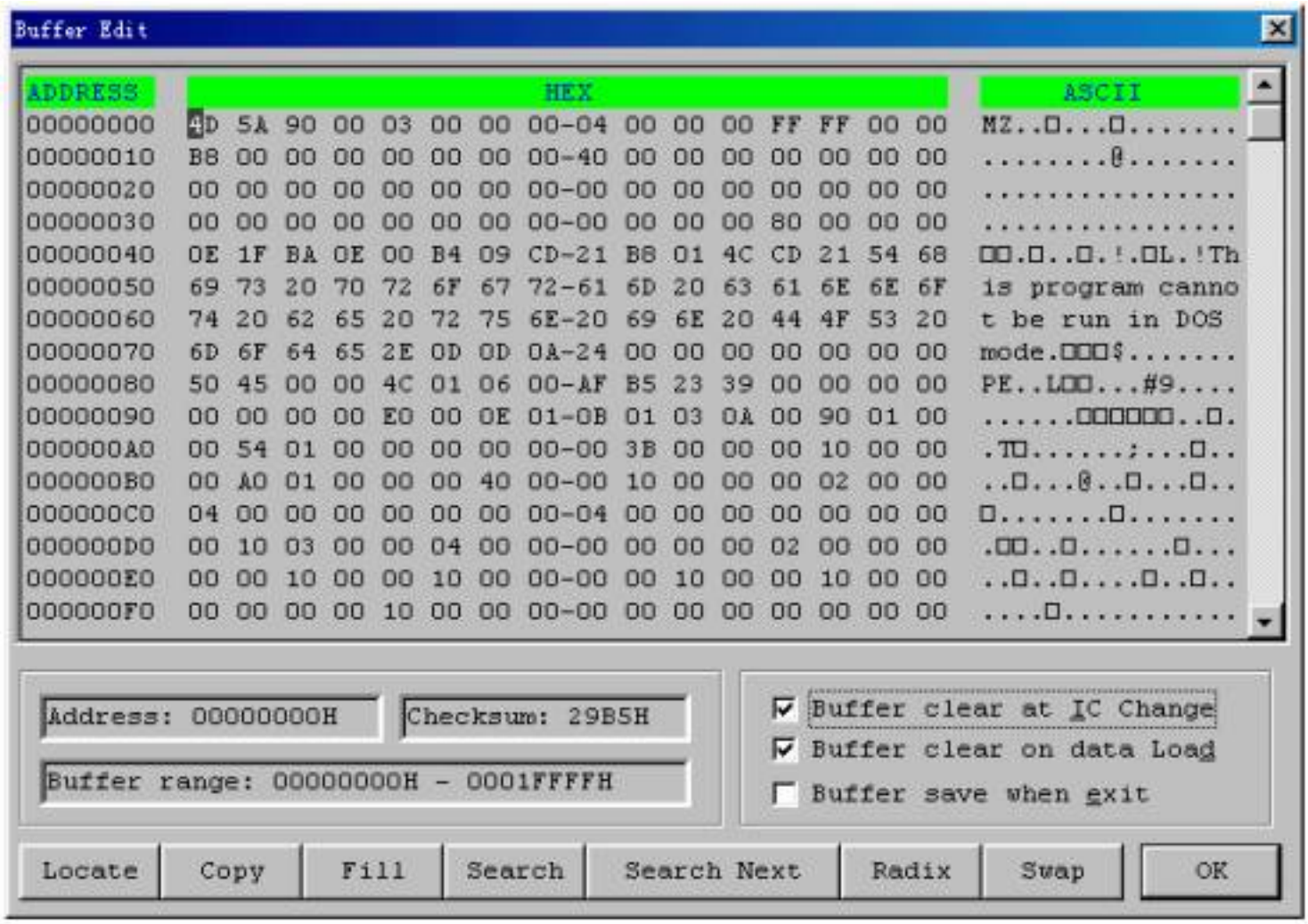
<Ctrl-PageDown> Move cursor to the end of the buffer

<Home> Move cursor to the beginning of the line

<End> Move 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 for editing. 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. It consists of the Start Address, End Address, Fill Data input lines, OK and Cancel buttons. Input desired data to be filled into the Fill Data input line, and specify the range by indicating the beginning and ending addresses. For the Fuse Buffer Edit window, the data will be either 1 or 0. For the Data Buffer Edit window, it will be a two character HEX code, such as AA, 55, E4

·Copy

The function displays Copy Buffer dialogue box. It consists of Start Address, End Address, New 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 0-10(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

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 two explanations according to the different devices.

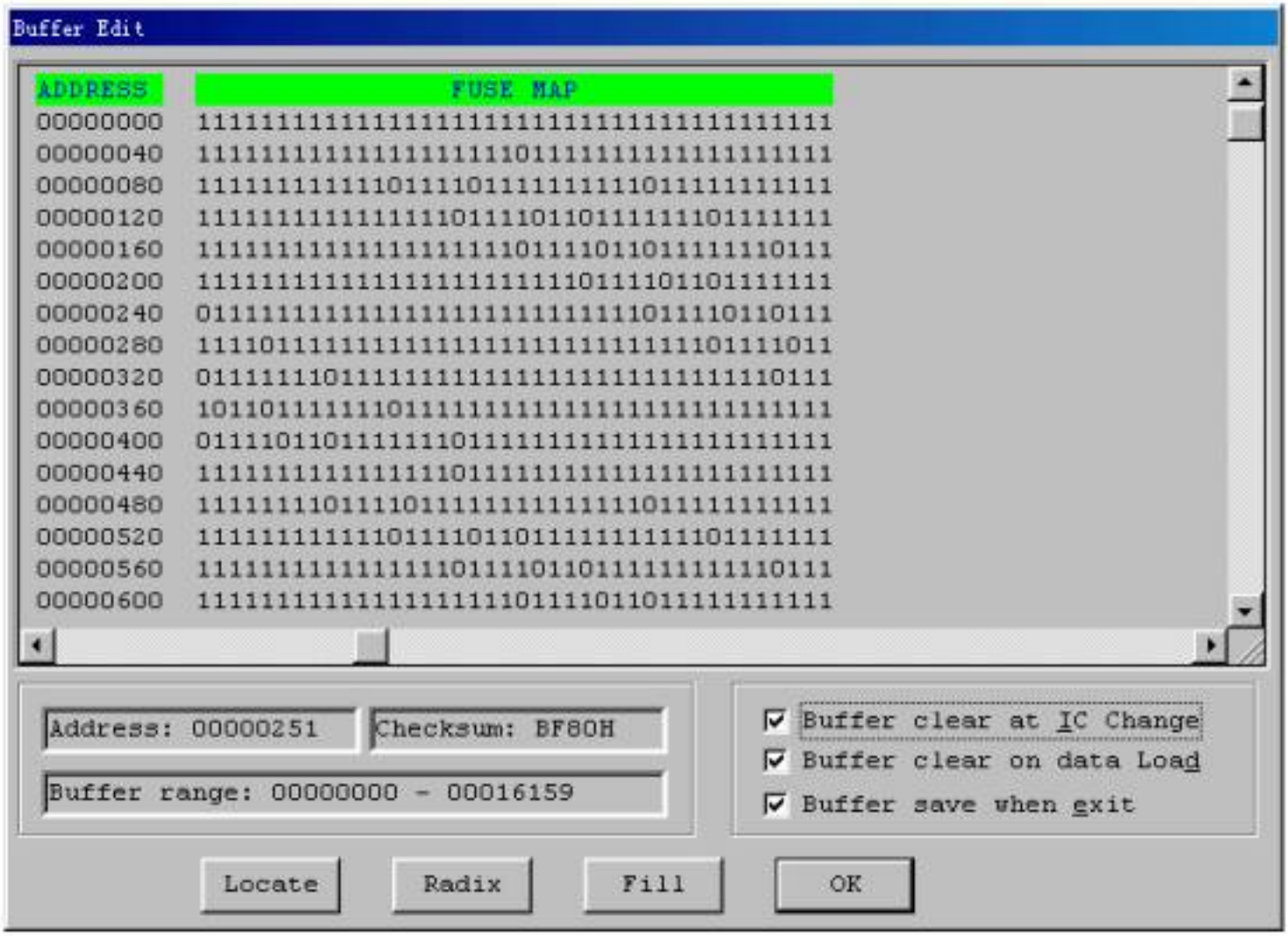
·1 represents an intact fuse

  0 represents a blown fuse

·1 represents an 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 when exit. When entering the system again, load the saved data automatically.

4.1.2.2 Save Buffer

User can save the data in the text file format to the specified file, the way is as 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 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 10011010101010101010101010101010

00000800 01101010101010101010101010101010

00000832 10100110101010101010101010101010

00000864 10101010011010101010101010101010

00000896 10101010101001101010101010101010

00000928 10101010101010100110101010101010

00000960 10101010101010101010011010101010

00000992 10101010101010101010101001101010

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.

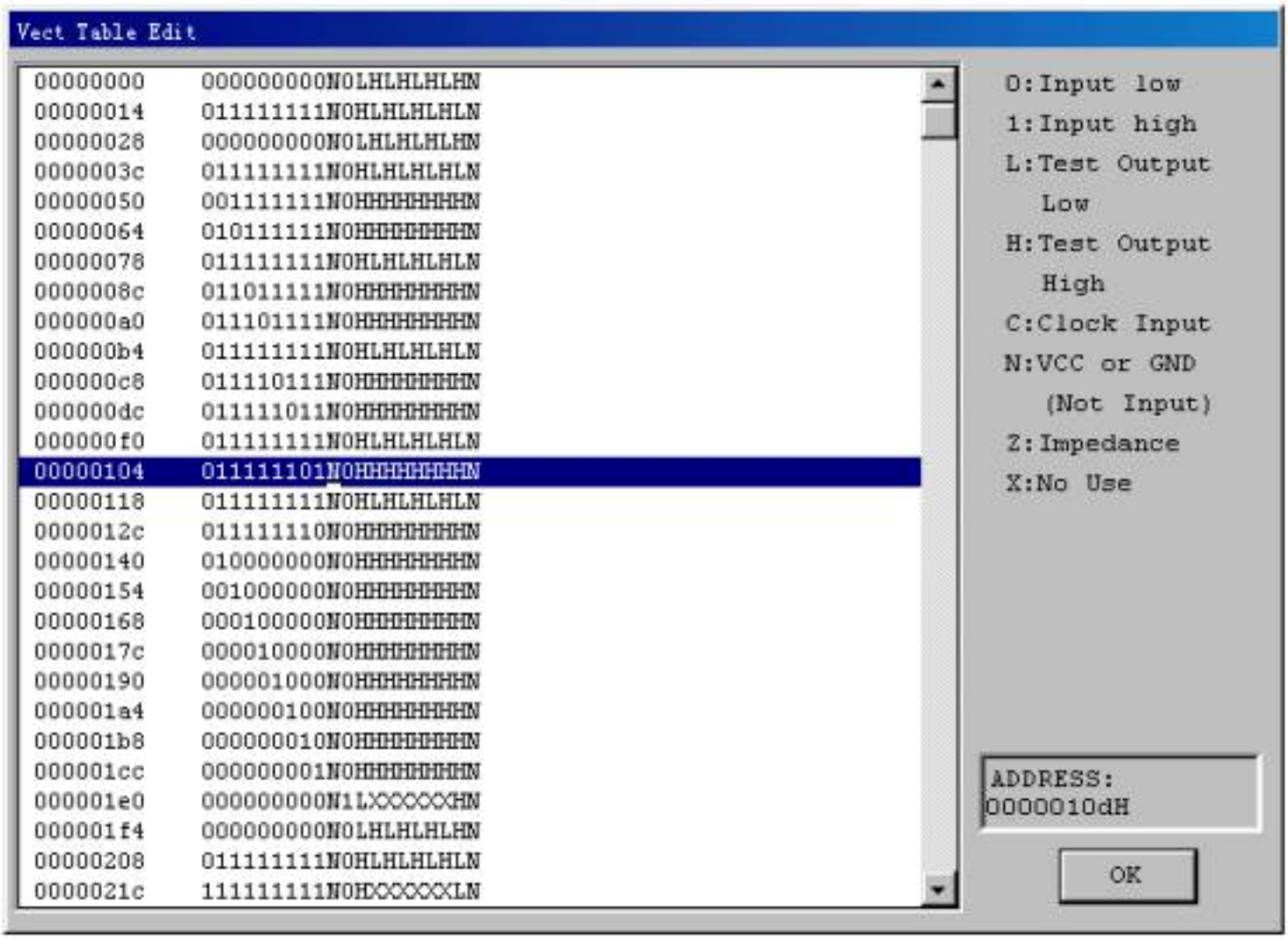
Edit: Opens the Encryption Buffer Edit window for viewing and editing.

4.1.2.4 E-Fuse

If the data type of the device is Fuse and the JEDEC file has E field, bring up the E Field Fuse Data Edit dialogue box. Please refer to the chapter of Fuse Buffer.

4.1.2.5 Vector Table

This opens the Vector Buffer Edit window. If a test vector table is included in the JEDEC file, the software will load the test vector table to the buffer when the JEDEC file is loaded. Each line display one vector table, each bit represents one pin's test character, from left to right related to the first to the last pin. Following is the test vector of a 14-pin device.



Z: High impedance state

X: Don’t care state

N: Vcc and Ground (output pins are not tested)

H: Output logic High (VOH)

L: Output logic Low (VOL)

C: Clock pin

1: Input logic High (VIH)

0: Input logic Low (VIL)

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 the devices.

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 Manufacturer column, at this point user can check whether it's correct or not by the icon shown at the lower right of the box.

·Select the device name through the Device Name column, click OK button.

Since some printed name on the chips may include speed, temperature, encapsulation parameter, when they are different with the names in Device Selection box, user should ignore them.

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. Entering a few vital characters the number of the devices will decrease greatly so that it is convenient for select the specific device. For example, as entering '89' in the Search Edit box all the devices, whether its name or manufacturer containing '89' will be listed in the list viewers. 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.