Advin Systems Inc.

SpeedPro-3000A
SpeedPro-580A
SpeedPro-280A

Universal Programmers

User' Guide

Rev 1.1

www.Advin.com

Advin Systems Inc., 556 Weddell Drive, #8,
Sunnyvale, CA 94089

(408) 243-7000 FAX: (408) 541-9006

Contents

1. General Description

1.1 Supplied Equipment

The supplied equipment are:

a. One SpeedPro Programmer
   (Model 3000A, 580A or 280A, depends on purchase)
b. One USB interface cable.
c. One 12V AC adapter (for 85-230 volt AC supplies)
d. One CD with installation S/W and User's Guide.

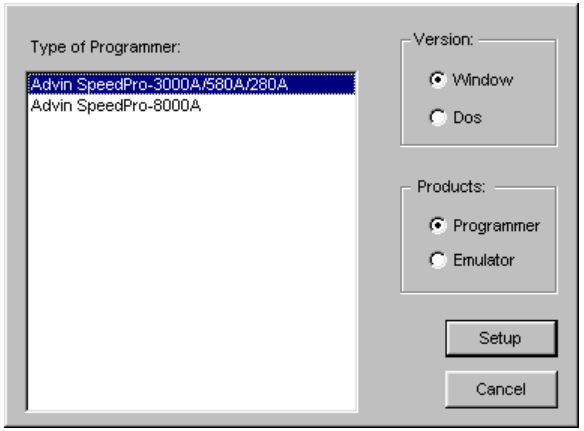
1.2 System Requirements

The system requirements are very simple. All SpeedPro programmers come with a DC power supply adapter, a USB cable. All you need to supply is a PC with a USB interface with a minimum of 128 MB RAM, running Windows 98, ME, 2000 or XP, and 10 MB of hard disk space.

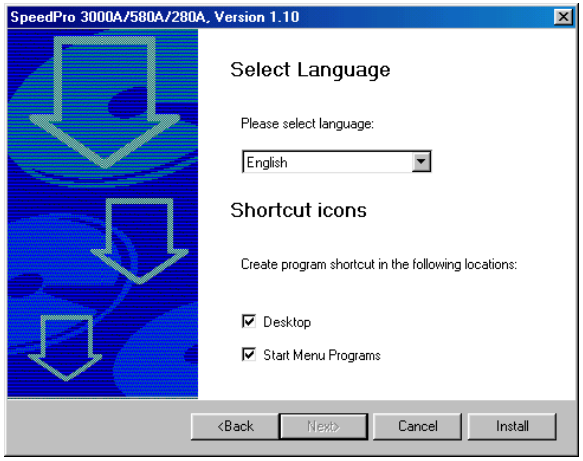
1.3 Installation Procedure

To install S/W:

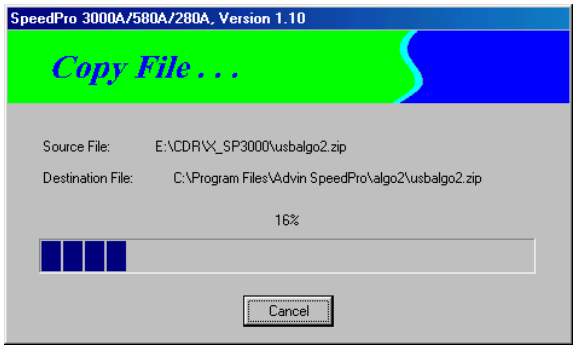
1. Insert CD into computer.
   Select type of programmer by clicking on "Advin SpeedPro-3000A/580A/280A", then click [Setup]:



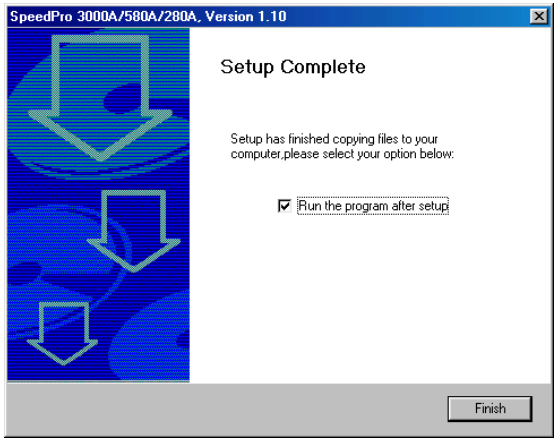
Follow prompts until this screen appears. Then click [Install].



Then this screen appears:

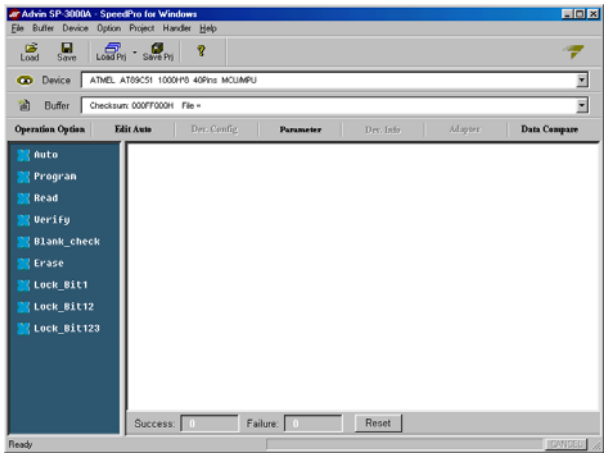


2. At the following screen, check the box:
   [ ] Run the program after setup.
   Click [Finish]



3. The Advin SpeedPro screen shall come up.

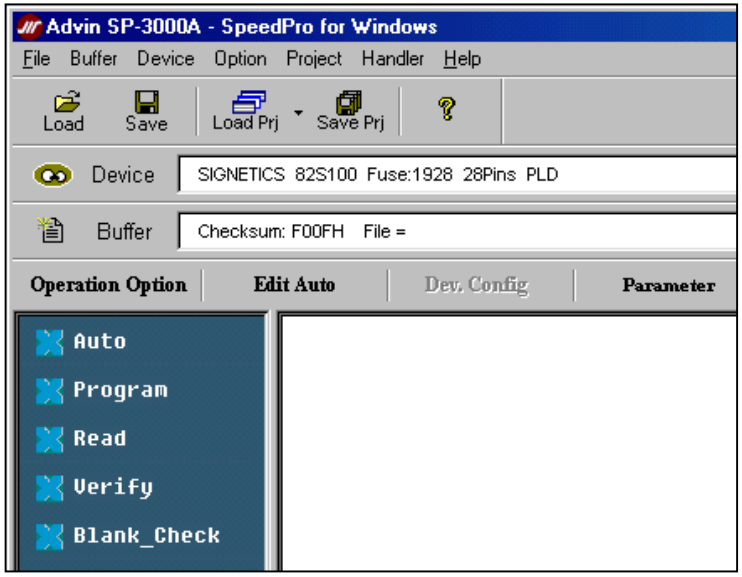
4. Connect programmer to computer. Turn on power switch on programmer. Click [Retry].
   The S/W shall now be able to recognize the programmer and the following screen shall come up:



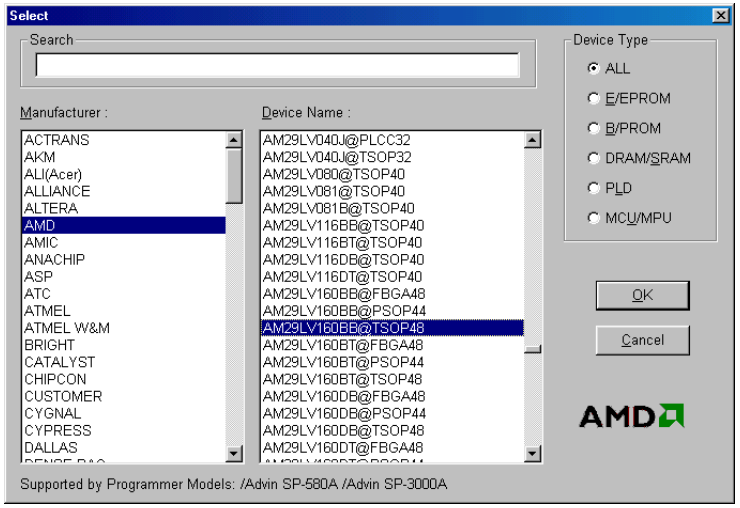
2. Quick Start

You are just a few minutes away to be able to program a device.

A. Select a device by clicking on the **Device** icon.



B. Select the manufacturer of the device and the device name:



C. Next, insert the device into the programming socket. Make sure the DIP device or the adapter is inserted “bottom-aligned”, with pin 1 away from you.

D. If you want to read from the device, just click the **Read** icon. The **Read** operation will read form the device into the memory buffer.

E. If you need to program the device from a data file, click on the **Load** icon and follow prompts. The **Load** operation loads data from a file into the memory buffer.

F. If you want to see the memory buffer contents, click on the **Buffer** icon.

G. The **Program** operation would (only) program the device, using data from the buffer. The **Verify** operation would verify the device.

H. The **Edit Auto** icon can be clicked to edit the **Auto** procedure. For example, for an EE erasable device, you can add **Erase, Program,** and **Verify** to the **Auto** procedure. Then when you click **Auto**, all three operations will be done for you.. With flexibility in setting the **Auto** procedure, you have control on what operations you need. For example, for programming brand new devices, if you want, you can skip the **Erase** operation and achieve higher thru-puts.

3. Detailed Description of Menus

For the most part the menus are self-explanatory. This section may help to explain certain menus that may not be as obvious.

Main Menu and Edit Menu
The Main menu contains all the current data relevant to the chip specified for programming. Across the top of the main screen is the menu bar, which consists of the menus: File , Buffer, Device, Test, Option, View, Window, and Help.
The Edit menu consists of the menus for editing buffer data, close buffer editing window to return
Main menu. Some functions only valid in Software for Windows Version

3.1 File: <Alt-F>

This menu deals with data file management and system interfacing.

3.1.1 Load <Ctrl+O>

Files loaded into the programmer are divided into two types. For programming memory and Microcontroller chips, data file types are generally HEX or Binary. For programming PAL or PLD chips, the data file type is .JED (JEDEC).

Loading files for Memory or Microcontroller chips
Data types supported are Binary (also .POF), Intel HEX (also extended HEX), Motorola S record, and Tektronix HEX types. Select the appropriate type for your data file. For Binary type data, it is possible to set buffer offset and file offset addresses.
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 not known, then a partial path may be entered using wild cards, e.g. ‘*.*’ or ‘*.BIN’.

Loading JEDEC (.JED) files
If the selected device type is PLD, the Load JED File dialogue box will pop up. Type in the file name and press return.

(You can choose PLD in the sub-menu Manufacturer' Default device format.)

3.1.2 Save

This selection will save the current data from the buffer into a file whose name is supplied by you.
For PLD devices, the Save JED file dialogue box will pop-up. It consists of Name (input line), Files (list viewer), OK and Cancel buttons. You can indicate the directory and the name of the file to be saved in the input line, labeled Name.
For E(E)PROM, BPROM or MCU device types, the box named Save File will be displayed. This dialogue box consists of an input line titled Name, a list viewer titled Files, an input line for Buffer offset, another input line for Save size, radio buttons for File format, and the OK and Cancel Buttons. You must input the path and the file name in the Name input line. The format of the file to be saved must also be selected in the File format area by pressing one of the radio buttons.

3.1.3 Exit

This command closes the programmer software and returns you to the control of the operating system.

3.2 Buffer: <Alt-B>

This menu manages the data file.

3.2.1 Edit

This selection brings up the Fuse Buffer edit window if the device type is a PLD, or the Data Buffer edit window for memory devices. The buffer is for 8-bit data, and the numbers in the left column are the addresses of the buffer. You can edit the buffer data on the screen with the cursor keys.

For the Fuse Buffer edit window, 1 is logic High and 0 is logic Low.

The Data Buffer edit window has two edit areas: one supports Hex code, and the other supports ASCII. You can switch the cursor between the two sections with <Tab>.

Locate

This selection is convenient for locating a set of data for editing. This selection will bring up the Locate buffer dialogue box, which consists of the New address input line, OK and Cancel buttons. Key into the input line the address you wish to be displayed on the current screen, and press OK. The cursor will blink at the address specified in the space marked New address.

Fill

This selection will bring up the Fill buffer dialogue box. It consists of the Start address input line, End address input line, Fill data input line, OK and Cancel buttons. Input any 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.

Copy

This selection displays the Copy buffer dialogue box. It consists of the Start address input line, End address input line, New address input line, OK and Cancel buttons. Data between Start address and End address will be copied to the buffer starting from New address.
Note that the source and destination areas must not overlap.

Swap

Swap byte order with specific word width and address range.

Radix

Toggles HEX/DEC memory address display.

Search

Search for a combination of hex/ASCII codes (search string).

Next

Performs the next search for the search string in 'Search'.

3.2.2 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 of interest is equipped with an encryption array, and the right selections have been made for the manufacturer and the part number of the chip.

Load: Brings up the Load Encryption Table dialogue box. It consists of the Name input line, File list viewer, OK and Cancel buttons. This selection loads an encryption table.
Edit: Opens the Encryption Buffer edit window for viewing and editing.

3.2.3 Vector Table

This opens the Vector Buffer edit window. If a test vector table is included in a JEDEC file, the software will load the test vector table to the buffer automatically when the JEDEC file is loaded. Refer to the following when editing the test vector.
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)

3.3 Device: <Alt-D>

This menu is used to choose the IC manufacturer, and the device name of the chip to be programmed.

To select a device, first click on the Select box. This allows the selection of the manufacturer followed by the device part number. (Make sure that you always select the Type of device you are going to program, before selecting either the manufacturer or the part number.)

3.4 Device Operations

3.4.1 Program

This 'burns' the data from the buffer into the chip. The Verify function is performed automatically after programming is complete. If an error occurs, the error message will be displayed along with the addresses and data of the chip and buffer where the error started. Any other result will be displayed in the Message section. The current address is incremented in the Current Address display window while the chip is being programmed or verified.
In the case of ROM, the range for the buffer and chip addresses, to be used for various functions, can be changed in the right half of the Address box.

3.4.2 Read

This reads the content of a chip into the buffer. After reading is complete, the checksum of the data will be displayed in the cobalt blue background screen (in the lower-left corner). If the chip selected is either a PAL of GAL, the blow count will be shown instead of the checksum. When a GAL is programmed, the device should be specified correctly (otherwise an error message will be displayed and the chip will not be programmed or read). If the security fuses are blown in a PAL of GAL, the data read from the chip will be all 1's or 0's instead of the real data, due to the data protection scheme.
If it is a ROM or Microcontroller, the data between the chip start address and the end address will be read into the buffer. The range of the addresses in downloading transactions, and the result of the transactions, will be displayed in the Message display window.

3.4.3 Verify

This function compares the contents of the buffer and the contents of the chip. If the data in the buffer is different from the data in the chip, the Verify Failure message and the address where it started failing will be displayed in the Message window. For a ROM or Microcontroller, verification of a range between a start and end address (specified by you) is possible.

3.4.3 Blank Check

This function checks that a memory device is correctly erased. For a ROM or Microcontroller, checking of a range between a start and end address (specified by you) is possible.

3.4.4 Security

This feature applies to PLDs and Microcontrollers equipped with a security function. If the security is set, the data programmed into a chip will not be read correctly.
Note: Lock, Protect, etc also mean Security dependent on Device.

3.4.5 Encryption

This applies only to some Microcontrollers. This will program the content of the encryption table onto the encryption array of a chip.

The content of the encryption table can be loaded, saved, and edited. Once encryption is completed, the data in the main buffer will be the Exclusive-NORed data with the encryption table.

3.4.6 Word Format

The Word Format window allows you to specify the data format loaded into the buffer. The default setting is Gang1–byte. Changing the setting is necessary only if the data loaded is not in normal byte (8-bit) mode; then it will be necessary to select a specified byte for programming purposes.
The Word Format selection brings up the Word Format dialogue box, which consists of the Word Format radio buttons, OK and Cancel buttons. The details of formatting are as follows.
Byte
This processes in a normal 8-bit data format.
Even Word
This processes in a 16-bit data format. When Even Word is indicated, the unit will program only the data bytes at the even addresses to an 8-bit wide device. For example:

Given:

buffer address 00 01 02 03 04 05 06 07
buffer data 01 23 45 67 89 AB CD =

Result after programming:

device address 00 01 02 03 …
device data 01 45 89 CD …

Odd Word
When Odd Word is indicated, the unit will program the data bytes at the odd addresses to an 8-bit wide device. For example:

Given:

buffer address 00 01 02 03 04 05 06 07
buffer data 01 23 45 67 89 AB CD EF

Result after programming:

device address 00 01 02 03 …
device data 23 67 AB EF …

Note: The next four topics are for the word format for double words (32-bits) as a unit of processing data. There are four kinds of formats in Double word. The following example will be used for explaining the four formats of Double word.

Given:

buffer address 00 01 02 03 04 05 06 07 08 09 0A 0B
buffer data 01 23 45 67 89 AB CD EF FE DC BA 98

Double Word 0
This will process the data at addresses 00, 04, 08_ for reading, verifying, and programming. (e.g: 01 89 FE)
Double Word 1
This will process the data at addresses 01, 05, 09_ for reading, verifying, and programming. (e.g: 23 AB DC)
Double Word 2
This will process the data at addresses 02, 06, 0A_ for reading, verifying, and programming. (e.g: 45 CD BA)
Double Word 3
This will process the data at addresses 03, 07, 0B_ for reading, verifying, and programming. (e.g: 67 EF 98)

3.4.7 Test: <Alt-T>

The SPEEDPRO' s can test logic ICs and RAM devices, as well as perform vector testing on PLDs. The user-friendly software provides an easy interface for loading files, editing test patterns, and downloading and uploading data between the buffer and the device. The SPEEDPRO* s provide a 74/54 and 4000/45000 TTL/CMOS logic device test library, and memory test algorithms. The following sections describe the functions available for testing logic ICs and RAMs.
Tests are performed by applying the input(s) specified in the test pattern (see below) and checking for the specified output(s). Up to 291 I/O combinations can be specified in a test pattern, allowing quite complex devices to be tested.

3.4.7.1 New Pattern

To test a new device that is not included in TTL.LIB, a new test pattern may be created.
To access this function, click on the Test menu from the main screen, and then click on New Pattern. This selection brings up the Append dialogue box, which consists of IC Type input line, Pin number input line, and the OK and Cancel buttons. You can input the new IC name and number of pins in the input lines. If the name and the number of pins is not in the current library, pressing OK will open the XXXX edit window (XXXX is the new device name that was entered). Refer to the following when editing a pattern. The following symbols are used throughout the sub-menus of Test:
V: Vcc pin

G: GND (Ground) pin
X: Unused or power pin; output values will not be tested
H: Output Logic High (Voh)
L: Output Logic Low (Vol)
C: Clock pin
1: Input Logic High (Vih)
0: Input Logic Low (Vil)

3.4.7.2 Edit Pattern

This selection is used for editing a test pattern in the library. When you are in the Test menu, click on Edit Pattern. This brings up the Select Chip to Edit dialogue box. It consists of Select Type input line, a device list viewer, and the OK and Cancel buttons. You can enter the device name into the input line, or select from the list viewer. On accepting the device name, the software opens the edit window with the name of the device at the top. You may now use the above information to edit your pattern.

3.4.7.3 Delete Pattern

This selection deletes a test pattern in the library. Click on the Delete Pattern sub-menu in the Test menu. This will bring up the Select Chip to Delete dialogue box. Highlight the test pattern you wish to delete and click on the OK button. The test pattern will be deleted and you will return to the main menu. Make sure you have made the correct selection. The only way to retrieve a deleted test pattern is to recreate the pattern (using 'New Pattern').

3.4.7.4 TTL&CMOS test

This selection tests TTL and CMOS devices. Click on the sub-menu TTL & CMOS test in the Test menu. This will open the Select Chip to Test dialogue box. Highlight the device that you wish to test, and then click on the OK button. The result of the test will be displayed in the window Test TTL, which will appear after clicking the OK button in the Select Chip to Test dialogue box. If the device passes the test, a 'passed' test message will be displayed. If the test fails, information about where it failed is displayed. If you want to repeat the test, click on the Repeat button. Otherwise click on the Cancel button.

3.4.8 Auto find device

This selection finds out the name of an unknown device. Click on the sub-menu Auto find device. This will initiate a search for the device type inserted in the programmer. The device name found will be listed in the list viewer, List of Detected Chips, if the device can be listed. If the device cannot be found, it will display an error message, No chip found. To make sure the chip is programmable, manually enter the manufacturer and the device name in the given boxes, under the Device menu.

4.4.9 Vector test

To perform vector testing of a device inserted into the programmer, click on the Vector Test sub-menu in the Test menu. This will initiate the test procedure and open up the Test vector window. Here, the result of the testing will be displayed. If you wish to repeat the test, click on the OK button. This will initiate the test procedure again. If you want to return to the main screen, click on the Cancel button.

3.4.10 Option: <Alt-O>

This menu deals with optional settings, such as changing the port address, and specifying paths for the software. In short, all the optional menu settings and configurations for running the program can be set here.

Check “Insertion Test” if you want the programmer to do an insertion test before any operation on the device. Check “Beeper” if you want the programmer to beep after any operation .
Initialize Programmer: Clicking on communication will search the programmer through all possible parallel port. If find the programmer, the communication will be successful. If not, a window named Error will appear .

Data Buffer: If checking, buffer will be clear before changing IC or loading data from file.

4. Error Messages

Invalid File Type:
An input file with wrong format is selected. Please select right format for the input file.

Init Programmer error(Programmer init error):
Communication searches no programmer connected. Software only works in Demo method. Be sure programmer powering on, search programmer with communication(See Setting sub menu) again.

Algorithm file not found:
Every device has an algorithm file. After selecting the device, running will load algorithm file(DLL file).If loading failed, show this message, and algorithm file name with full path.

\man\SpeedPro.doc