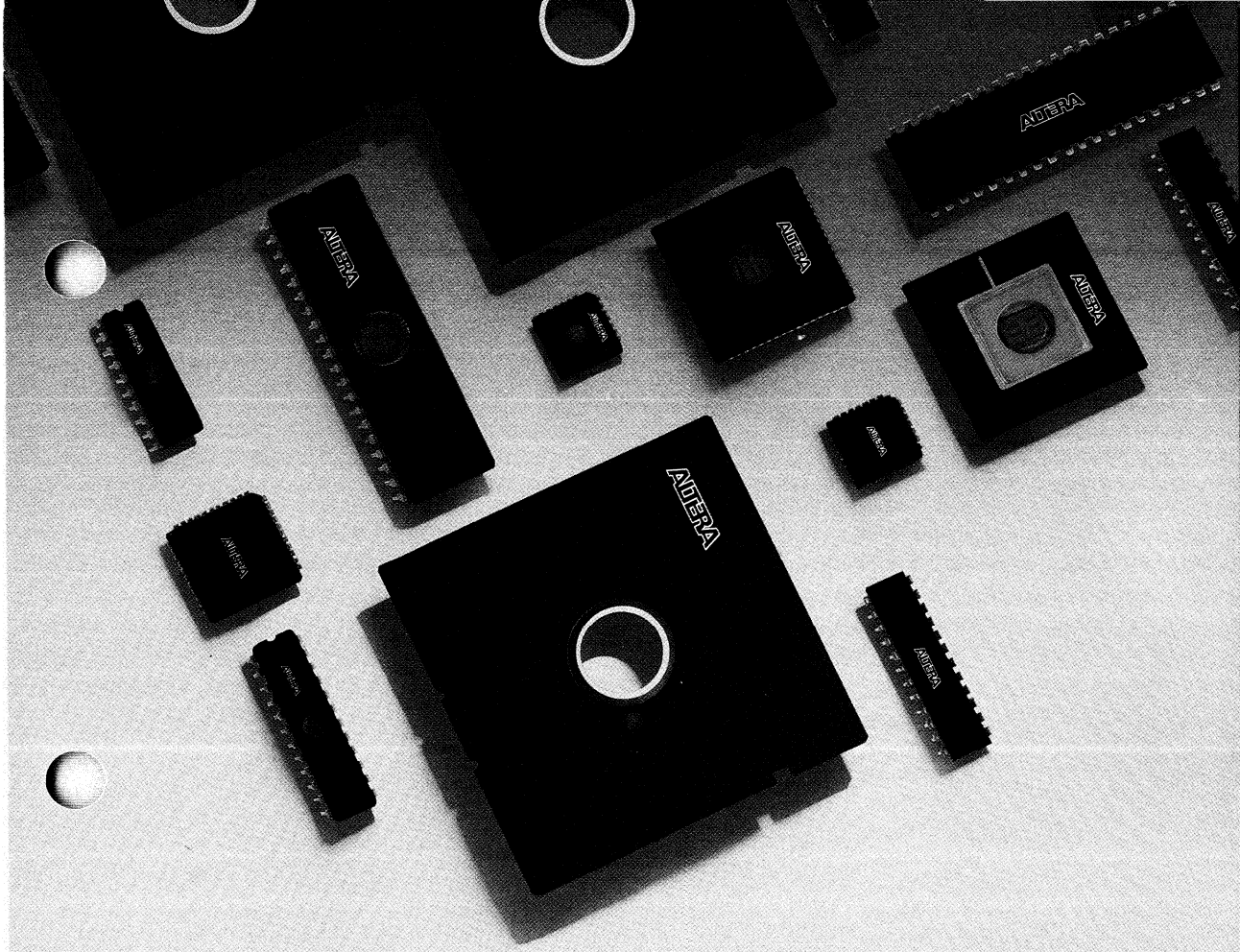


LogicMap II

LogicMap II
Version 5.0
September 1987 P25-02137-01

Changes are made periodically to the information contained in this manual. These changes will be incorporated into subsequent editions.

Altera Corporation
3525 Monroe Street
Santa Clara, CA 95051
(408) 984-2800
TELEX: 888496

Read This First...

Your LogicMap II documentation consists of three main parts:

Installation provides instructions for installing the hardware and software required to program an Altera part.

Using LogicMap provides information on how to invoke LogicMap II; a detailed description of LogicMap's hierarchical structure; and instructions on how to use the program.

Keystroke Conventions summarizes the keystroke conventions applicable to the various hierarchical levels.

In addition, the manual contains a section with *Error Messages*, an *Appendix,* a *Glossary*, and an *Index.*

At the back of the manual, you find a *Customer Comment Form* and a *Problem Report Card.*

Manual Updates

Altera documentation is updated with Change Pages, Section Reprints, and a **READ.ME** file.

Change Pages are issued for minor changes to the manual. New information is identified with vertical change bars in the margins next to the changed text. In addition, the date of issue is printed at the bottom of each page.

Section Reprints are issued if a section requires a substantial number of changes. The date of issue is indicated at the bottom of each page.

A **READ.ME File** is provided on the **INSTALL** diskette. This file contains information about recent changes to the software that are not yet reflected in the manual.

Printing Conventions

The following notational conventions are used throughout this manual:

Times Bold	—	all A+PLUS commands, prompts, and messages
	—	all user input, including keyboard keys
Times Light	—	most file output as displayed on screen
Helvetica Italics Bold	—	all references to Altera manual titles
Helevetica Italics Light	—	all references to sections within Altera manuals
	—	information that requires special attention

Contents

Section 3: Keystroke Conventions

Illustrations

Figure **Page**

Tables

SECTION 1

Installation

This section assists you with the installation of the LP3 and LP4 Logic Programmer cards. For installation of the software, refer to *Installation* in your A+PLUS or SAM+PLUS manual.

For information on the Master Programming Unit, PLE3-12, refer to Section 3 of the Altera ***Data Book***.

Some of the data files on Altera-supplied floppy diskettes contain information characteristic of the presently available Altera parts. Additional files will become available for new parts.

To receive all future update information on this product, please insert the Warranty Card at the back of this manual into the Registration Envelope and mail it together with the completed main Registration Card.

Hardware Requirements

Hardware required for programming and verifying Altera parts consists of a Logic Programmer card, LP3 or LP4, and the Master Programming Unit, PLE3-12.

If you are using a third-party programmer and have any questions regarding specific requirements, please contact:

Altera Corporation
Applications Dept.
3525 Monroe Street
Santa Clara, CA 95051
(408) 984-2805 ext. 102

A+PLUS and SAM+PLUS programs require an IBM Personal Computer XT, AT, or other compatible computer capable of running MS-DOS Version 2.0 or a later version. The computer must have at least one floppy disk drive capable of reading 360-Kbyte, double-sided, double-density disks, a 10-Mbyte or larger hard-disk drive, and 640 Kbytes of RAM memory. In addition, one full length expansion slot is needed for the Logic Programmer card. Requirements for the optional modules are specified in the description of the individual modules.

Hardware Installation

Before installing the hardware, be sure to record the serial numbers of the logic programming card and the programming unit. These numbers are required to register your system and may be needed for future reference. Please take a moment to record them here:

	Serial Number
Programming Card:	
Programming Unit:	

Figure 1-1 shows you where to find the serial number on the solder side of your programming card:

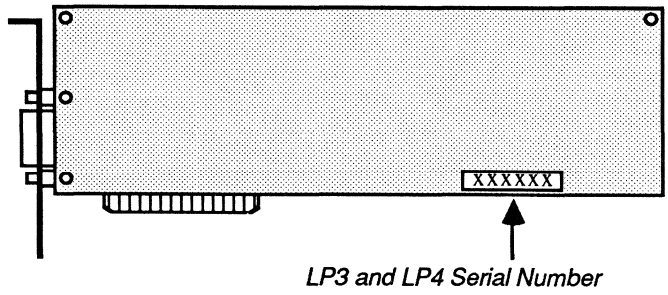


Figure 1-1. Location of LP3/LP4 Serial Number

Figure 1-2 shows you where to find the serial number on the programming unit:

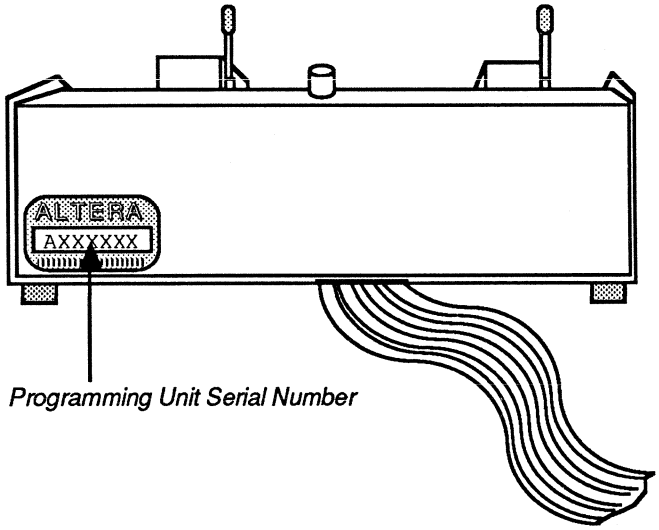


Figure 1-2. Location of Programming Unit Serial Number

Follow these steps to install the Logic Programmer card:

1. Be sure the computer's power is turned off.

2. Refer to the Operations Manual accompanying your particular computer for instructions on how to remove the cover.

3. On the Logic Programmer card, locate the dipswitch and set all four switches to ON as shown in Figure 1-3.

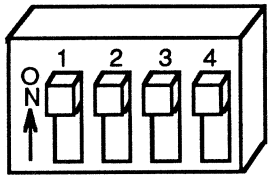


Figure 1-3. Switch Settings on the Logic Programmer Card

The default I/O address is 280 hex. If this address must be changed because of an addressing conflict, see *Appendix A* for switch settings for other I/O addresses.

Switch settings are different for the LP3 and LP4 Logic Programmer cards. To verify which card you have, refer to Figure 1-4. Then set the switch settings as shown in *Appendix A*.

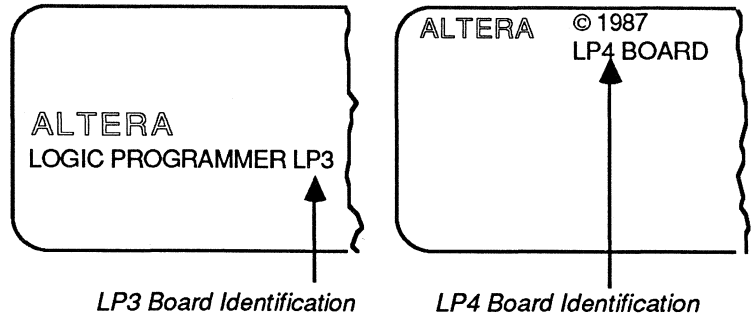


Figure 1-4. LP3 and LP4 Identification

4. The computer should contain a number of expansion slots. Select any convenient empty slot for the Logic Programmer card. If the expansion slot is covered, remove the screw that holds the expansion slot cover. Remove the cover. Refer to Figure 1-5.

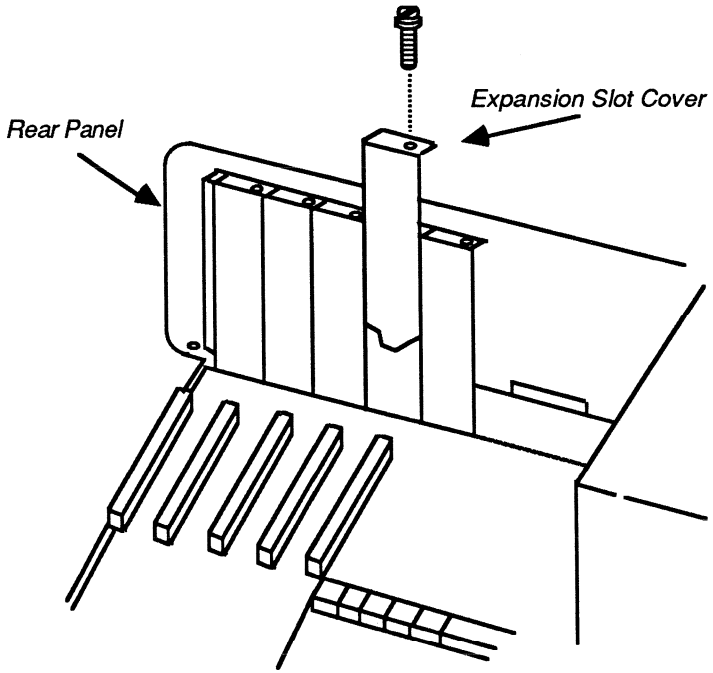


Figure 1-5. Removing the Expansion Slot Cover

5. Snap the plastic card guide supplied with the Logic Programmer card into the expansion slot on the front panel as shown in Figure 1-6.

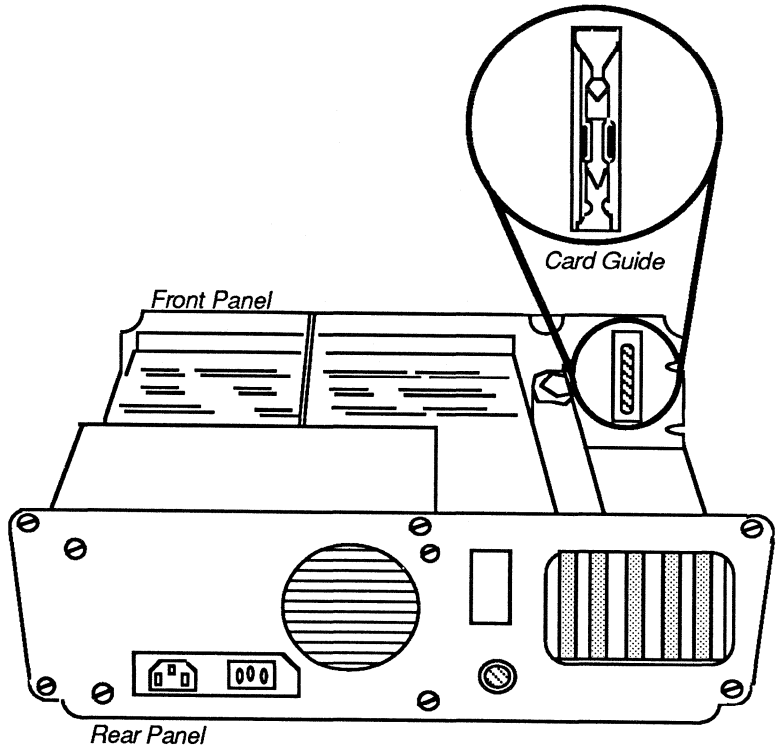


Figure 1-6. Installing the Card Guide

6. Insert the card into the expansion slot using the card guide; then lock the board in place by fastening the retaining bracket with the screw from the slot cover. See Figure 1-7.

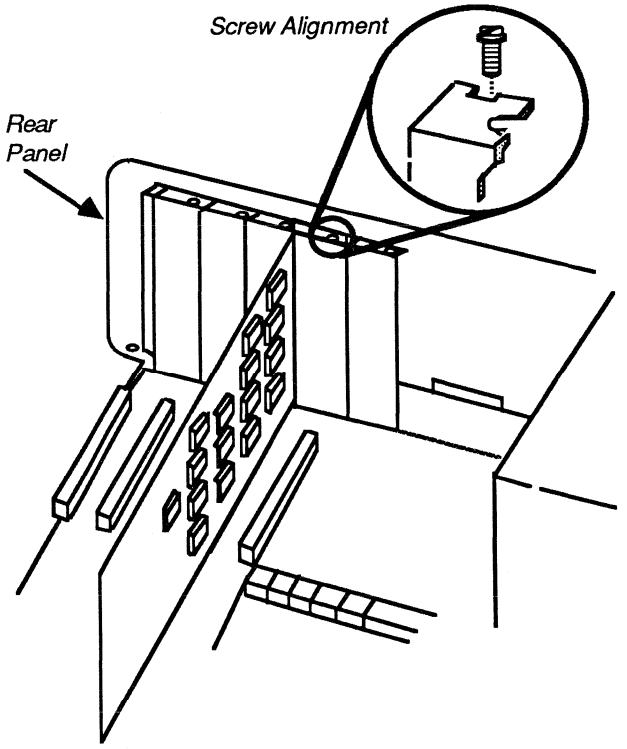


Figure 1-7. Locking the Board in Place

7. Use of the optional FutureNet schematic capture interface (DASH) requires installation of the graphics controller board or mouse interface card for FutureNet's Schematic Designer. Refer to the hardware installation instructions in FutureNet's Schematic Designer User Manual.

8. Replace the cover.

9. On the rear panel of the system, connect the Altera-supplied, 24-pin, flat ribbon cable to the Logic Programmer card.

10. Make sure all locking screws on all connectors are tightened.

Important Note

If you have an early version of the programming unit (PLE-3), your unit will not have the adaptor slot shown in Figure 1-8. In this case, you may *not* use an adaptor for programming a part. The sockets on the PLE-3 unit enable you to program an EP310, EP320, EP1200, or EP1210. Do *not* try to saw off the lip on the adaptor that is meant to slide into the adaptor slot.

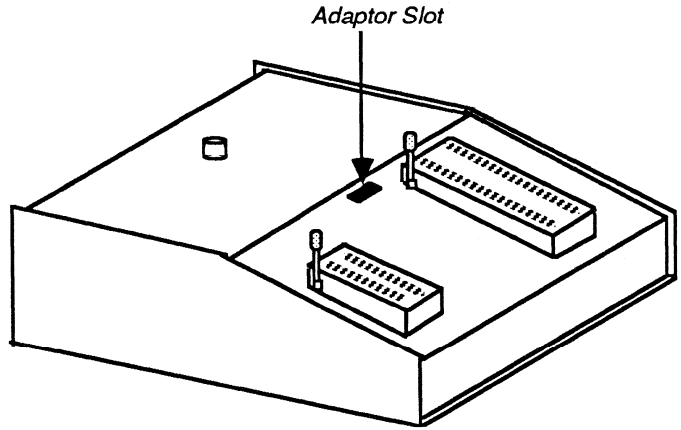


Figure 1-8. Location of Adaptor Slot on the PLE3-12 Programming Unit

The hardware installation is now complete.

Software Installation

If you have purchased the PLCAD-SUPREME, PLCAD4, or PLDS2 development system, you should follow the installation instructions described in *Installation* of the ***A+PLUS User Guide***.

If you have purchased the PLASAP development system, install your software according to the following instructions.

If You Have Purchased PLASAP...

...you should have the following distribution diskettes:

- **INSTALL**
  Contains the installation procedures for all A+PLUS modules.
- **LOGICMAP**
  Contains the LogicMap II program and support files.
- **ECF**
  Contains data files used by LogicMap II.

Backups of Altera Distribution Diskettes

Before installing Altera software, you must make backups of all Altera distribution diskettes in case one of the distribution diskettes fails to work. Note that the copy-protected diskettes are specially formatted. You can copy the files from these diskettes onto a backup diskette, but you cannot start the copy-protected programs with the backup diskette. So, if you lose or damage an original copy-protected diskette, you must use the backup of that program and copy it onto a disk that has the special formatting.

The DOS **DISKCOPY** command cannot be used to duplicate the distribution diskettes because they are copy-protected and **DISKCOPY** is unable to read them.

To make backups of all Altera distribution diskettes, you must go through the following steps:

1. Boot DOS.

2. Format a blank diskette for each distribution diskette with the DOS **FORMAT** command. (Refer to the DOS Manual.)

3a. If your system has one floppy disk drive and one hard disk drive, put an Altera distribution diskette into drive **A** and type:

COPY A:*.* B: <Enter>

You are prompted to replace the Altera distribution diskette with a formatted diskette.

3b. If your system has two floppy disk drives and one hard disk drive, put the Altera distribution diskette into drive **A** and the formatted diskette into drive **B** and type:

COPY A:*.* B: <Enter>

4. Repeat step 3 for each Altera distribution diskette.

5. Store your backup diskettes in a safe place.

LogicMap II Installation

Once you have completed the following installation procedure, you will be able to run LogicMap II directly from your hard disk.

Before installing A+PLUS software on your hard disk, you must ensure that it has at least 1 Mbyte of free disk space on the hard disk and 640 Kbytes of RAM memory, otherwise installation will not be successful. Available space is verified with the DOS **CHKDSK** command. (For information regarding DOS, refer to *IBM Disk Operating System, Version 3.30 User's Guide*; *IBM Disk Operating System, Version 3.30 Reference;* and *IBM DOS Technical Reference.*)

IBM AT high density 1.2 Mbyte diskette drives are not compatible with the standard 360 Kbyte diskette drives. If you anticipate moving A+PLUS from an AT to an XT computer, you must use a 360 Kbyte diskette drive during the installation. This installation procedure can be run on drive **B** exclusively.

This procedure assumes that your hard disk is drive **C**. (If you have another hard-disk drive, substitute the appropriate letter.)

1. Boot the computer from the hard disk.

   Remove any write-protect tabs that may be on the distribution diskettes.

2. Put the Altera-provided INSTALL diskette into drive **A** or **B** and enter:

   A:INSTALL <Enter> (if you use drive **A**)
   B:INSTALL <Enter> (if you use drive **B**)

   The program will perform some basic checks after which the Installation Main Menu, shown in Figure 1-9, is displayed.

```
A L T E R A Programmable Logic User System
Software Configuration Program
Copyright (C) 1987 Altera Corporation
version 5.0

        Main Menu

[1]  Install Software
[2]  De-Install Software
[3]  Change Programming Hardware & A+PLUS menu configuration.

Press a number key to select an option:
Press <Esc> to exit to DOS.
```

Figure 1-9. Installation Main Menu

Item [1] Guides you through A+PLUS installation.

Item [2] De-installs the software from the current system so that you can move the software to another computer.

Item [3] Allows you to change the A+PLUS hardware installation file called EPLD.SYS. This file is used only if you want to change the address location of the programming card or disable the color display option. Refer to *Appendix A* for valid programming card addresses.

3. Press **<1>** to select option [1]. A set of prompts will guide you step by step through your system configuration.

4. Next, you are prompted step by step through the actual installation procedure.

This installation process has a master menu that provides installation for all optional products. If you try to install an option which you have not purchased, you will be returned to the Main Menu.

You are guided through the following process:

a. You create a directory on your hard disk that contains all files from the installation diskette. You are also asked to confirm the installation results.

b. You are requested to enter the address location of your programming card. Type:

280 <Enter>

This is the default address.

c. You are asked to indicate whether you wish color or monochrome display.

d. You are asked to enter the name of your editor. (LogiCaps is the default editor.) Press **<Enter>**.

5. Now, the Installation Menu is displayed. See Figure 1-10.

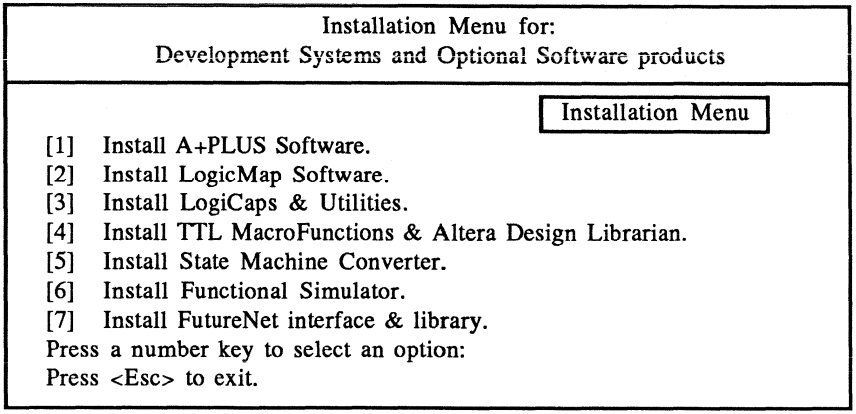


Figure 1-10. Installation Menu

a. Press <2> to install LogicMap II.

   You are prompted to insert the LOGICMAP and ECF distribution diskettes.

b. When you are finished, press <Esc> to return to the Main Menu.

6. You may select other menu items from the Main Menu or press <Esc> to return to DOS.

7. **Remove any remaining diskette from the floppy disk drive and press <Ctrl><Alt><Del> to reboot the system before using any of the A+PLUS programs.**

Your CONFIG.SYS and AUTOEXEC.BAT files may have been modified to make LogicMap II run properly. Therefore, after completing the installation procedure, you may want to examine these files to determine whether they are compatible with other software on your system. The original files will have been saved as CONFIG.BAK and AUTOEXEC.BAK.

De-Installation

If you wish to install your LogicMap II software on a different computer, you must first reverse the installation from your original system.

If you are using a programming card, you must move this card *after* you have de-installed the software and *before* you re-install it on the new computer.

With the original INSTALL distribution diskette in drive **A** or **B**, type:

A: INSTALL <Enter> (if you use drive **A**)
B: INSTALL <Enter> (if you use drive **B**)

The Installation Main Menu is displayed (see Figure 1-9).

Select menu item [2] on the Main menu. The De-Installation Menu is displayed as shown in Figure 1-11.

```
                    A+PLUS
        USE ONLY TO DE-INSTALL VERSIONS INDICATED

                                    De-Installation Menu

[1]  De-install A+PLUS (ver 5.0) Software.
[2]  De-install LogicMap Software.
[3]  De-install LogiCaps (ver 1.6) & Utilities.
[4]  De-install TTL MacroFunctions & Altera Design Librarian.
[5]  De-install State Machine Converter.
[6]  De-install Functional Simulator (ver 2.5).
[7]  De-install FutureNet interface & library.
Press a number key to select an option:
Press <Esc> to exit.
```

Figure 1-11. De-Installation Menu

Press **<2> <Enter>**.

As a result of this de-installation, you can no longer use the LogicMap II software on the hard disk of the first system.

After the initial installation, you cannot use the original distribution diskettes for additional installations unless you run the de-installation program first.

After de-installation, some APLUS files that are not application files will be taken off the directory on your hard disk.

Organization

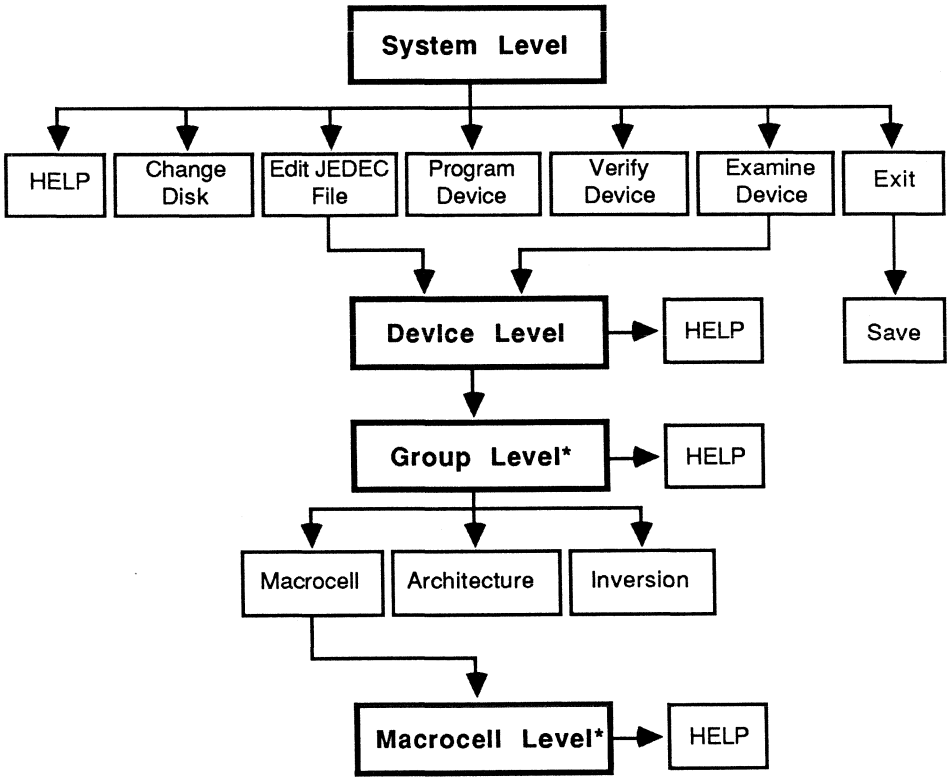
LogicMap II operates on four hierarchical levels, each with its own windows providing quick access to a desired level. The lower the level, the more detailed the area displayed and the information supplied.

The four major windows are as follows:

- System Level Window
- Device Level Window
- Group Level Window
- Macrocell Level Window

These windows provide access to other windows that offer additional information—e.g., HELP—or perform additional functions such as part selection.

Figure 2-1 shows the system hierarchy and the functions associated with each level.



These levels are not accessible for SAM and Buster parts.

Figure 2-1. LogicMap II System Hierarchy

System Level Window

The highest, most general operational level is the System Level. The System Level menu, which is displayed when you invoke LogicMap II, shows a list of functions on the left side of the screen, a window to the right of the function list, and on top of the screen a line identifying the particular part to be programmed as well as the currently open JEDEC file. Refer to Figure 2-2.

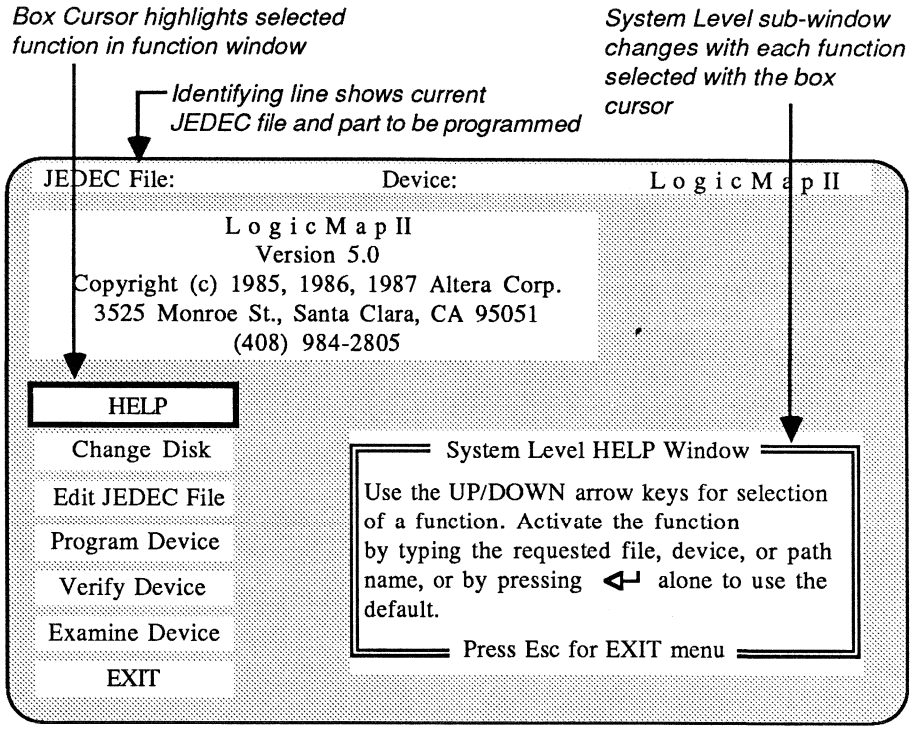


Figure 2-2. LogicMap II System Level

On start-up, a box cursor highlights the Help function. You press <↓> or <↑> to move this cursor to the other available functions. To execute a function, press <**Enter**>.

Whichever LogicMap level you're in, <Esc> usually gets you to the Exit menu on the System Level.

The following functions are accessible from the System Level menu:

HELP

Displays information on all the functions available on the System Level. See Figure 2-2.

Change Disk

Displays the current path name and directory list. If you have changed a diskette and press <Enter>, a new directory is read in from the current diskette. You may type in a new drive/path in DOS format to display a new directory. See Figure 2-3.

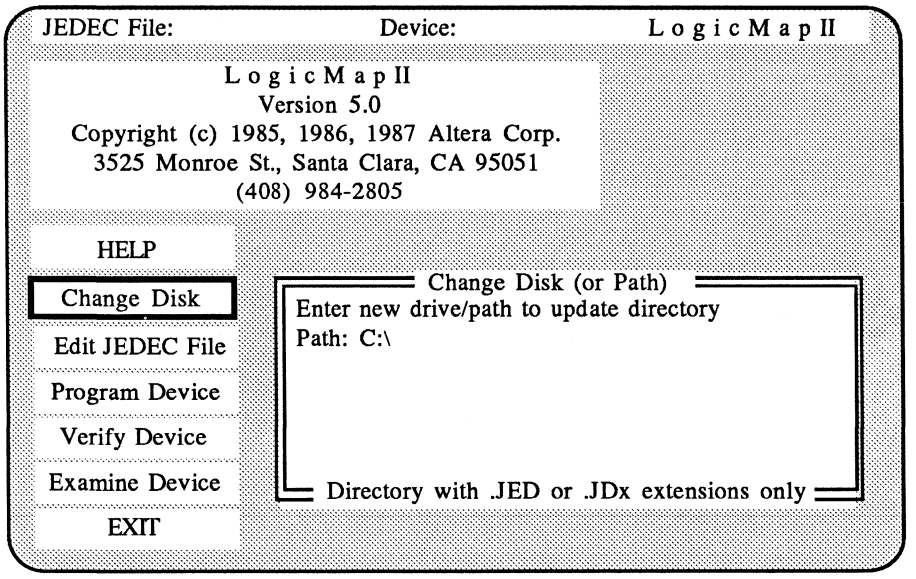


Figure 2-3. Change Disk Window

Edit JEDEC File

Allows you to edit JEDEC files. The Edit JEDEC File window is displayed and you are prompted for the filename. Enter a JEDEC filename (**.JED** extension need not be typed) and press **<Enter>**. The most recently updated JEDEC file is provided as a default. Refer to Figure 2-4.

If you enter a JEDEC filename that is not in the current directory, you are asked if you wish to create a new file by that name. If you answer **Y** (Yes), the data in memory are cleared and you are prompted for a part name again.

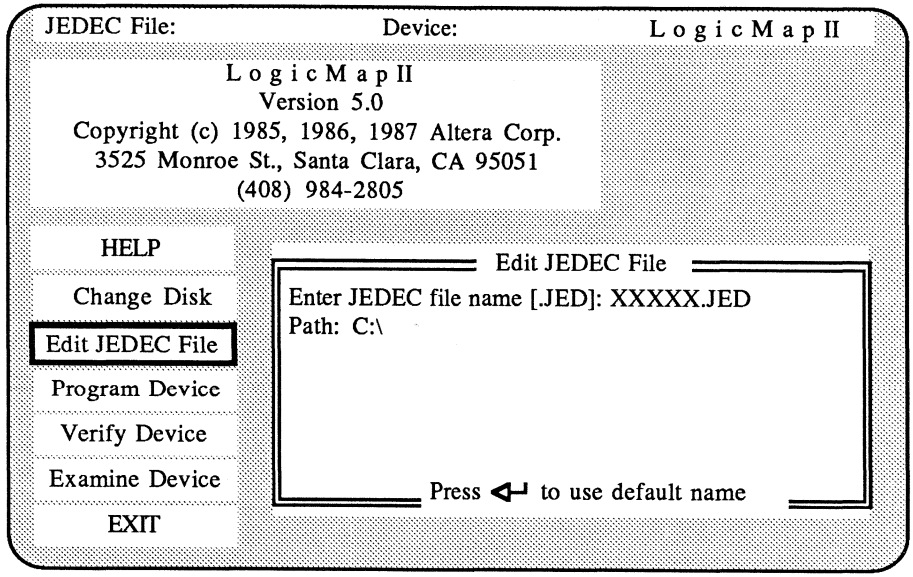


Figure 2-4. Edit JEDEC File Window

After you have entered the filename, the Select Device for Editing window is opened and you are prompted to type a part name. If the specified JEDEC file already provides a part name, that name is displayed as the default.

After you select a part either by pressing **<Enter>** or by typing in a name and pressing **<Enter>**, the JEDEC File Header Text window is opened showing information from the header section of the JEDEC file. See Figure 2-5.

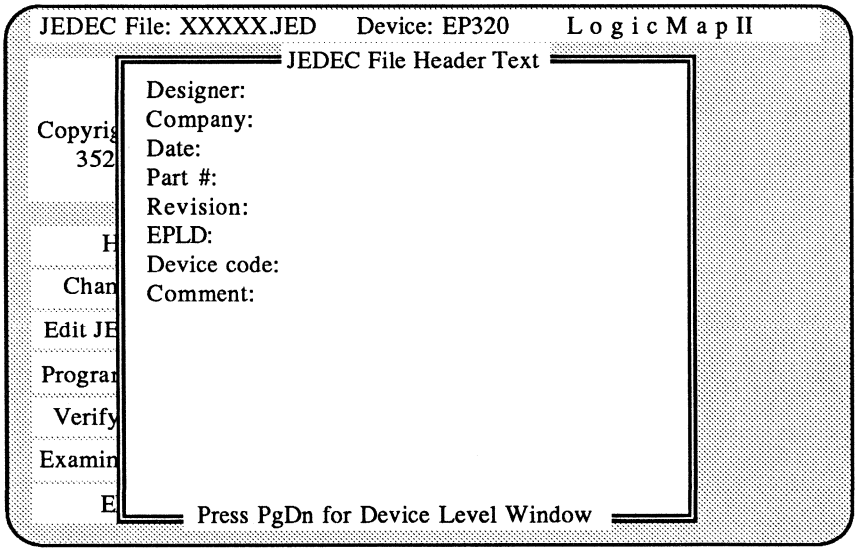


Figure 2-5. Header Window

Changes made at this level are not documented in the ADF. Any changes you make should also be incorporated into the ADF of the design.

A file is not written to disk until you execute the Save JEDEC File command explained below under **Exit**.

You may now make changes in the header text by moving around with **<↑>** and **<↓>** or by pressing **<Enter>**.

Table 3-3 lists the keys for entering and editing information in the Header window.

After completing the Header window, press <PgDn> to move to the Device Level, or press <PgUp> to return to the System Level. Refer to **Device Level Window** for detailed information on that level.

Program Device

Opens the Program Device window on the System Level and prompts you for the JEDEC filename. Type the name and press <Enter> or press <Enter> to use the default name.

If you had entered the part name previously with the Edit JEDEC File function, you will not be prompted for the part name again. If you have not yet specified a part, the Select Device for Editing window prompts you for a part name. You may also change the default part name.

After you press <Enter> the JEDEC File Header Text window is displayed as shown in Figure 2-6.

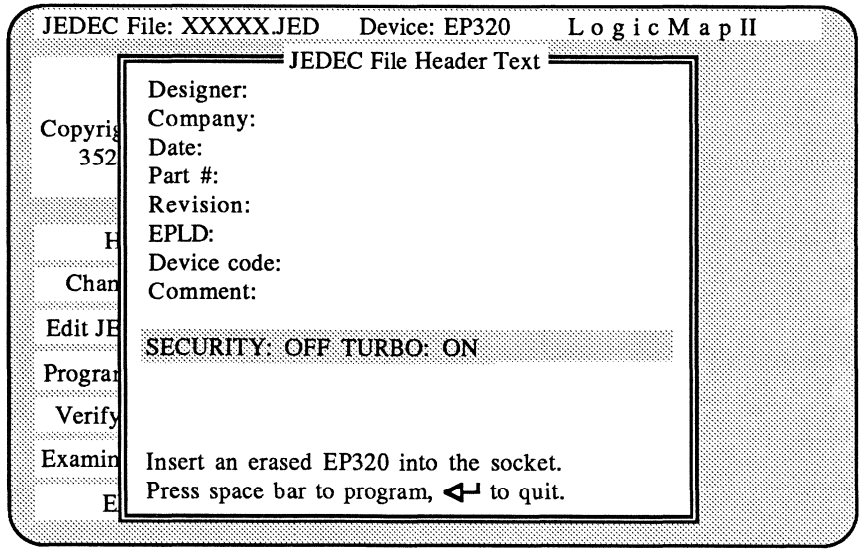


Figure 2-6. Program Device Window

This window performs three functions:

1. It displays the header information of the specified file.

2. It displays the status of the Security and Turbo-Bit options. (Refer to **Device Level Window** for information on these features).

    EP310, SAM and BUSTER have no Turbo-Bit option.

3. It prompts you to insert the part into the Master Programming Unit adaptor socket and press **<Space>** to program the part. If you decide not to program the part, you may also quit and return to the System Level. If you inserted the part incorrectly or the socket is not connected properly, a message on the screen prompts you to re-insert the part or connect the cables.

    If the part is an EP1210, the warning message is not displayed. Be sure to insert it correctly.

When programming is completed, you are asked if you wish to program another part. If you don't, you are returned to the System Level.

LogicMap checks whether the part you inserted is blank. If it isn't, you are asked if you wish to continue. Type **N** (No) to return to the System Level. If you type **Y** (Yes), LogicMap II will try to program the part. If it can't, you will get additional error messages.

 Most Altera parts have an ID. If LogicMap II does not recognize the part you have inserted into the socket, the message **Unrecognized EPxxx device!** is displayed. Be sure your part is a valid Altera-supported part.

Verify Device

Compares the information programmed into the part against the JEDEC file resident in memory. *Note, however, that this function may only be used if the Security bit is set to OFF.* When the box cursor highlights the Verify Device function on the System Level, the Verify Device window prompts you for the JEDEC filename. Type the name or press **<Enter>** to use the default name.

If you had entered the part name previously with the Edit JEDEC File function, you will not be prompted for the part name again. However, if you had not previously specified a part, the Select Device for Editing window opens and you are prompted for a part name. You may also change the default part name.

The JEDEC File Header Text window is opened. You are prompted to insert the desired part into the adaptor socket of the Master Programming Unit and press **<Space>** to verify the part. Refer to Figure 2-7. If you don't want to verify the part, press **<Esc>** to return to the System Level.

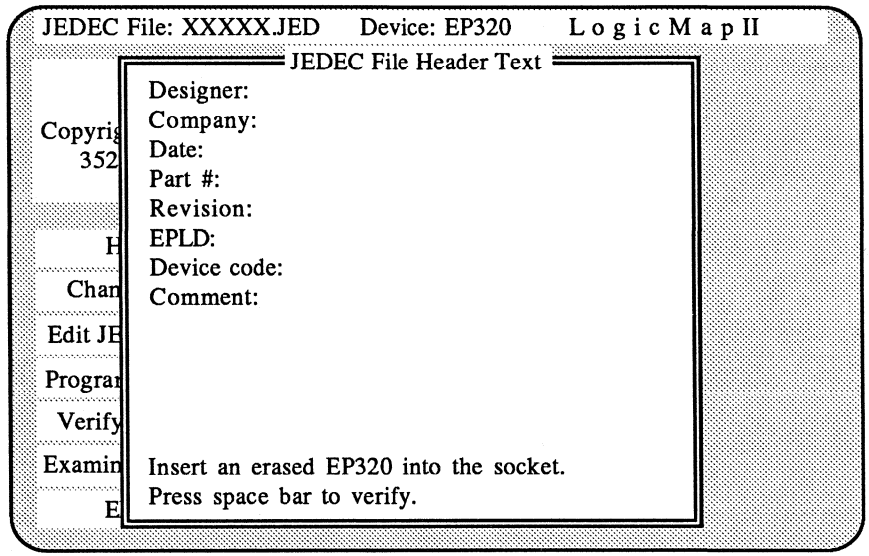


Figure 2-7. Verify Device Window

If your specified part is an EPLD...

...and an error message indicates discrepancies between the JEDEC file resident in memory (i.e., generated by the ADP) and the JEDEC data programmed into the part, you are asked,

Do you wish to view error data? [Y/N]

Press **Y** to view the data. You are returned to the Edit JEDEC File function on the System Level. Note that the default filename is **ERROR%%.JED**. You may inspect this file by going to the Group and Macrocell Levels and comparing the data to the data in the original JEDEC file.

If your specified part is a SAM or BUSTER part...

...and an error message indicates discrepancies between the JEDEC file resident in memory (i.e., generated by the SDP or ADP) and the JEDEC data programmed into the part, you are asked,

Do you wish to view error data? [Y/N]

Press **Y** to view the data. You are returned to the Edit JEDEC File function on the System Level. Note that the default filename is **ERROR%%.JED**. Use the Edit function to add a header to **ERROR%%.JED**. Since SAM and BUSTER parts cannot be inspected on the Group and Macrocell Levels, you must go to the Exit window on the System Level and press **S** to save the **ERROR%%.JED** file. You may then print a hard copy of this file and compare it to the original JEDEC file.

Regardless of the part specified, **ERROR%%.JED** is composed of **1**s and **0**s. The **1**s indicate discrepancies, the **0**s indicate agreement with the original JEDEC file. Thus, you only need to look for **1**s to find the errors.

In any case, you must remove the chip from the socket, erase it, and reprogram it.

The Verify Device function is also useful for checking with which JEDEC file a chip has been programmed, or for checking whether a chip is blank. (Suggestion: To verify a blank chip, create a JEDEC file with a blank part; then compare this JEDEC file, which will have all **0**s, against your current one. This will not work with SAM parts, however.)

Examine Device

Enables you to examine data from the part. *Note, however, that this function is only useful for non-protected parts. If you try to examine a part whose Security bit is set to on, you will get incorrect data.* When the box cursor highlights the Verify Device function on the System Level, the Examine (Read) Device window prompts you for the part name. Type the name and press **<Enter>**, or press **<Enter>** to use the default name.

After you have pressed **<Enter>**, the JEDEC File Header Text window is opened and you are prompted to load the part into the adaptor socket of the Master Programming Unit and press **<Space>**. Refer to Figure 2-8.

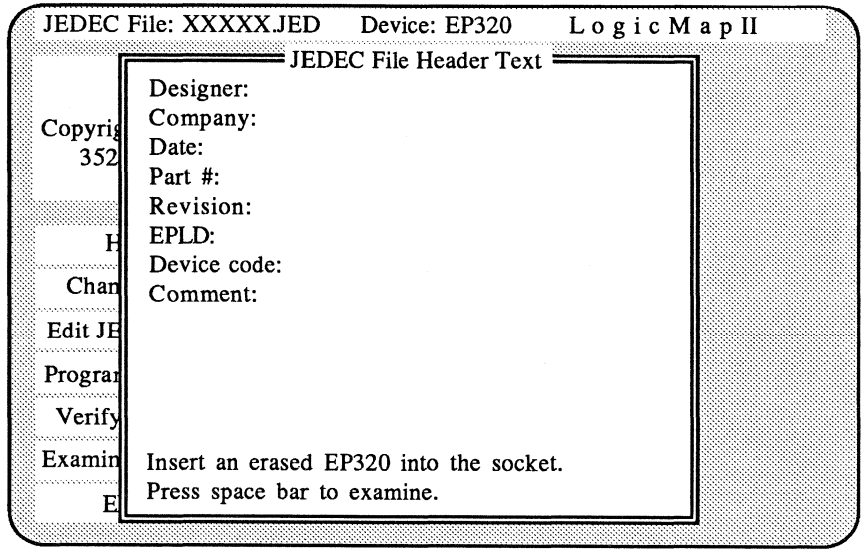


Figure 2-8. Examine Device Window

Data read from the part are used for the creation of JEDEC file data. These can be viewed via the Edit JEDEC File function, or they can be saved with the Save JEDEC File command available in the Exit window

described below. Names of JEDEC files depend on the part read. Examples: **EP600%%.JED**, **EP1800%%.JED**, **EPS448%%.JED**.

Exit

Saves the JEDEC file or exits LogicMap without saving. See Figure 2-9.

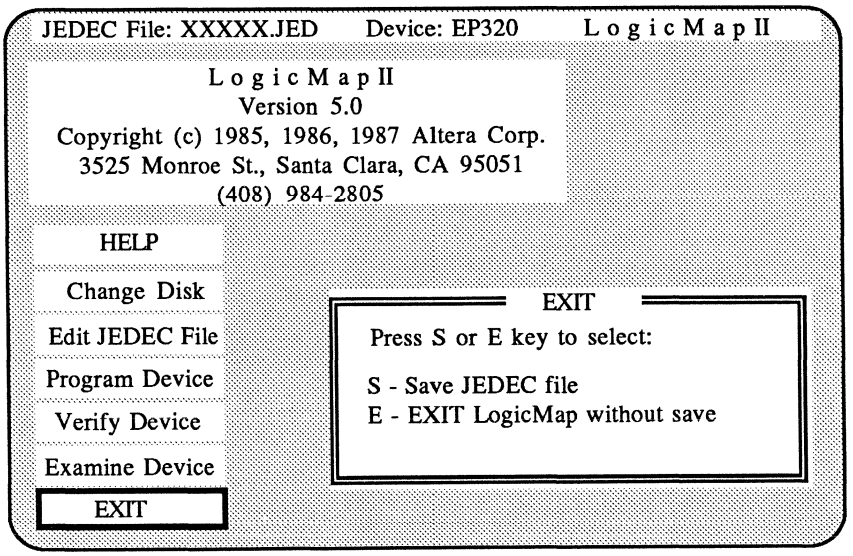


Figure 2-9. Exit Window

Type **S** to open the Save menu for saving a JEDEC file. Type the filename (the current filename is provided as the default) and press **<Enter>** to write the file. Note that you remain in the Exit menu.

Type **E** to exit LogicMap.

Device Level Window

The Device Level window displays a block diagram of the selected part. To reach the Device Level, select Edit JEDEC File in the System Level window and open the JEDEC File Header Text window. Press **<PgDn>** to open the Device Level window. Refer to Figure 2-10.

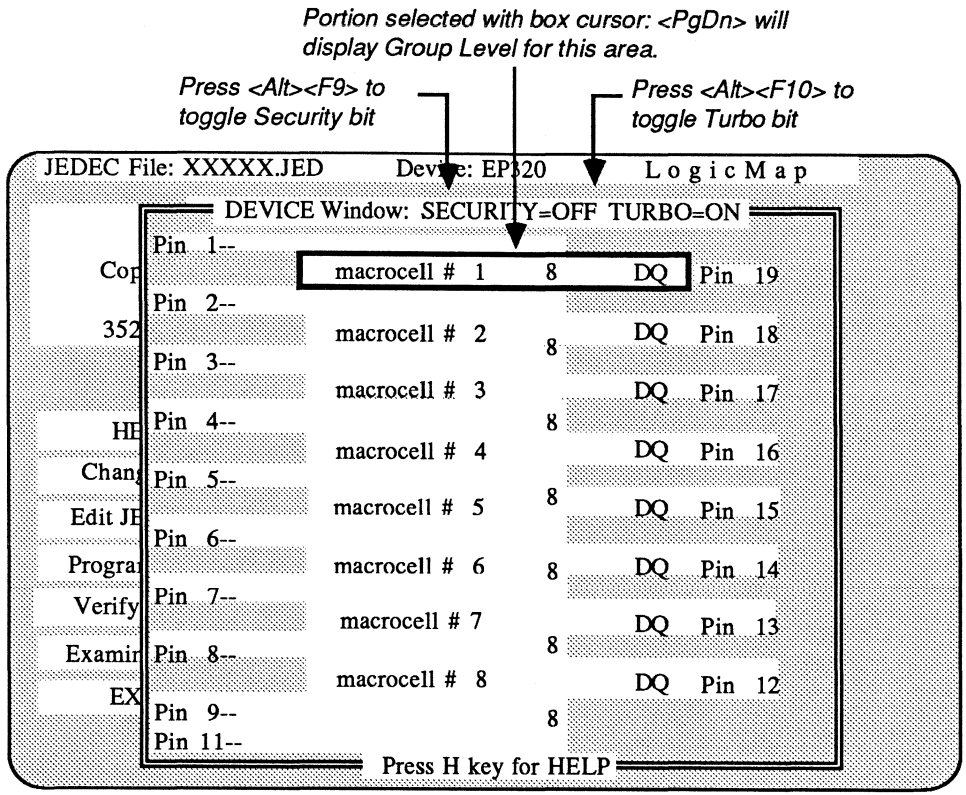


Figure 2-10. Device Level Window

At the Device Level you perform three functions:

- Set the Turbo-Bit option.
- Set the Security option.
- Select the portion of the part you wish to edit at the next lower (i.e., Group) level.

At this level, only the Security option is available for the EP310, SAM, and BUSTER parts. These parts don't allow setting the Turbo-Bit or editing groups and macrocells. When you open the Device Level window and your specified part is an EPB1400, EPS444, or EPS448, the following message is displayed:

> The EPB1400 bus-oriented* EPLD device cannot be edited on the bit level as can previous EPLD devices. Therefore, no graphical support has been provided and you may not page down below this level. You may, however, still protect your design by setting the security bit to ON. To do this, press any key to leave this HELP window, and then press <Alt><F9>.

* or: EPS444 micro-sequencer
or: EPS448 micro-sequencer

Setting the Turbo-Bit Option

The Turbo-Bit is a control bit available on some Altera parts. It enables you to alter speed and power characteristics for that part. To change the current setting, the Device Level window must be displayed on the screen. Press **<Alt><F10>** to toggle Turbo-Bit ON or OFF. ON is the default.

Setting the Security Option

The Security bit is a control bit available on Altera parts. If it is ON, the part cannot be interrogated or inadvertently reprogrammed. To change the current setting, the Device Level window must be displayed on the screen. Press **<Alt><F9>** to toggle Security ON or OFF. OFF is the default.

Other JEDEC data cannot be altered at this level.

Selecting a Portion of the Part to be Edited

While in the Device Level, you use the box cursor to select that particular group or portion of the part that you wish to edit or examine, and press **<PgDn>**. The screen then displays the Group Level window for the area you had targeted. The **<PgUp>** key returns you to the System Level.

Refer to Table 3-4 for Device Level keystroke conventions.

Group Level Window

This window displays a block representation of each macrocell, an indication of output inversion, and the architectural configuration associated with a particular group. The number in the macrocell block represents the number of product terms in the macrocell. In case of shared product terms (EP1210) the block cursor highlights the shared product terms together. The numbers on the edges of the window represent the part pin numbers.

Changes made at this level must also be incorporated into the source of the design. Altera does not recommend design editing from LogicMap.

To open the Group Level window, press **<PgDn>** while you are at the Device Level. Figure 2-11 shows a Group Level window.

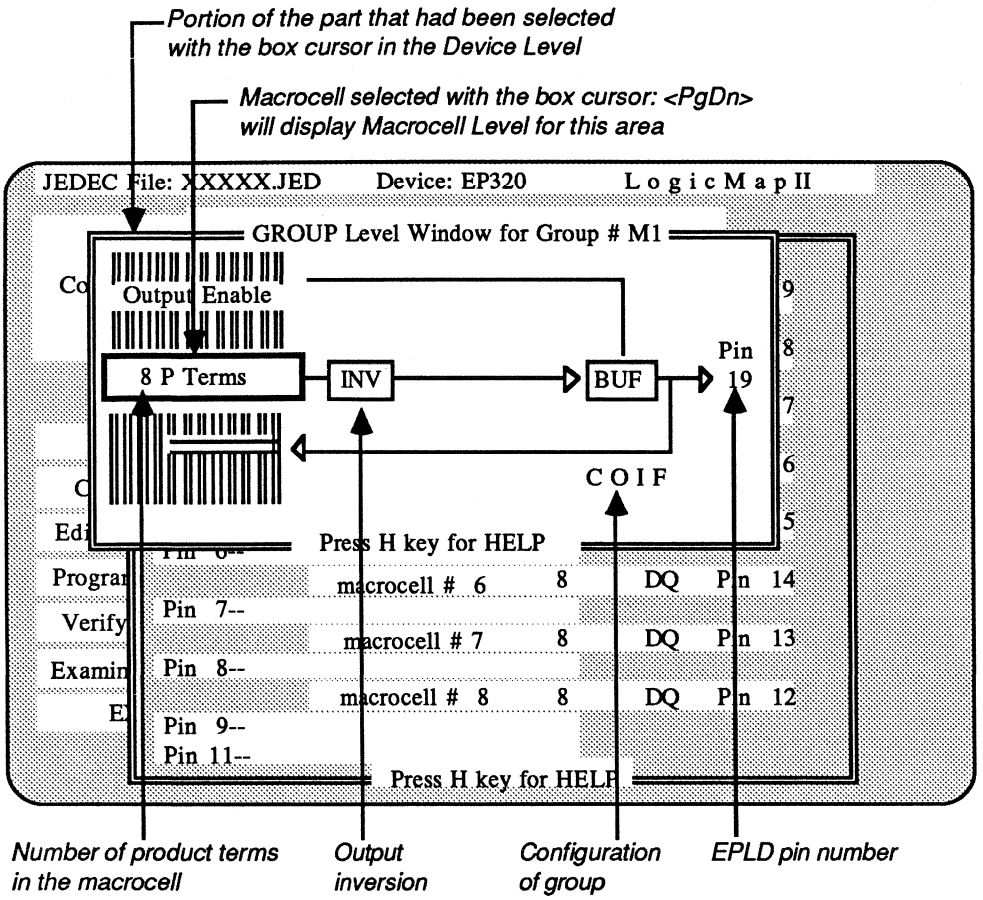


Figure 2-11. Group Level Window

To edit a particular macrocell, you target it with the box cursor and press **<PgDn>**. The system then moves to the Macrocell Level. Press **<PgUp>** to return to the Device Level.

An inverter may be inserted or deleted. To delete it, you select it with the box cursor by pressing **<→>**; then you press **<Enter>** or **<Del>**. To insert it, you select the inverter location by pressing **<→>** (the

location will be defined by the cursor but will otherwise be blank); then you press<Enter> or <Ins> to display the inverter.

You may reconfigure an architectural block by selecting it with the box cursor, then pressing <Enter>, <Ins>, or <Del> until you get the configuration you desire.

Refer to Table 3-5 for Group Level keystroke conventions.

Macrocell Level Window

The Macrocell Level is the lowest, most "local" level in the LogicMap II hierarchy. At this level, you may edit product term bits.

Changes made at this level are not documented in the ADF. Any changes you make should also be incorporated into the ADF of the design.

See Figure 2-12.

More than one macrocell can be displayed so that you may view shared product terms, input line labels, and product term labels.

The intensified crosshairs cursor is used for editing on the Macrocell Level. Each cell, shown as "+" or "." in the Macrocell Level window, represents an element of a product term. The "+" indicates an unprogrammed element that makes an electrical connection; the "." indicates a programmed element that has no connection. The numbers at the bottom of this window are pin numbers—both true and complemented—driving the input lines. Refer to Table 3-6 for Macrocell Level keystroke conventions to manipulate this cursor.

Press <PgUp> to return to the Group Level.

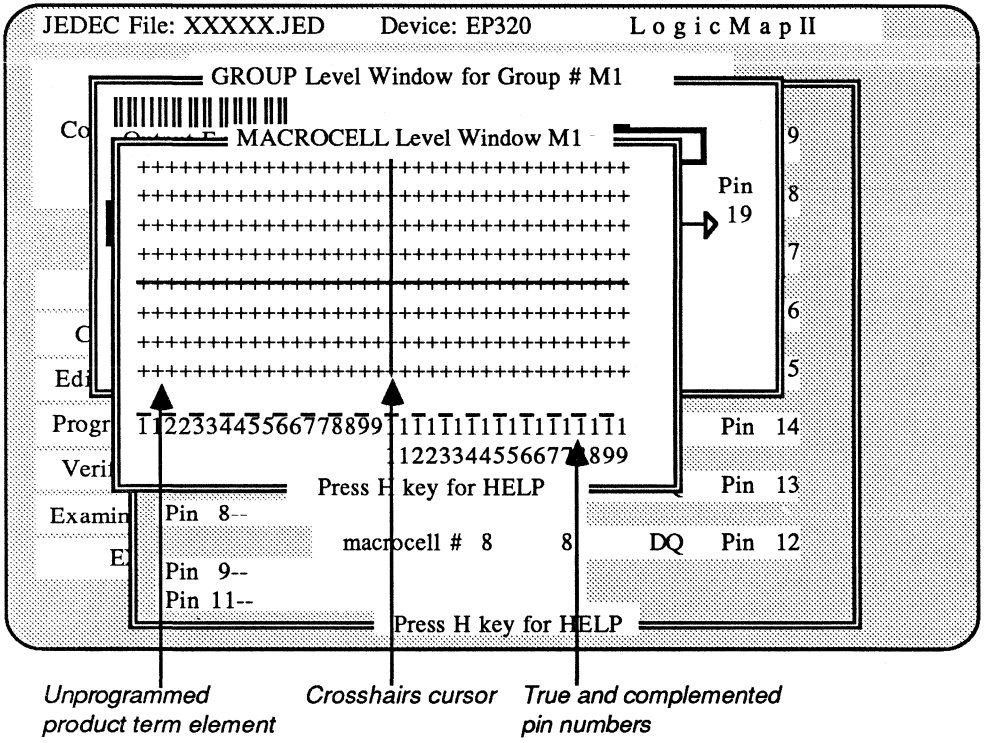


Figure 2-12. Macrocell Level Window

SECTION 3

Keystroke Conventions

LogicMap II commands are entered by simple keystrokes with the cursor and function keys. Some commands are used globally throughout the program, others are level-specific. The following tables list the keystroke conventions that are used globally and those that are used specifically at the System, Group, Device, and Macrocell Levels.

Global Keys

Table 3-1 lists and defines the functions of the keys used globally throughout the LogicMap II program.

Table 3-1. Global Keys

GLOBAL KEYS	
PgUp	Moves to the next higher level in the LogicMap II hierarchy.
PgDn	Moves to the next lower level in the LogicMap II hierarchy.
Esc	Moves to the System Level.

System Level Keys

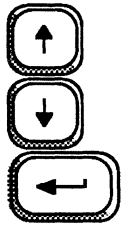
At the System level, you have two sets of key conventions:

- System level keys
- Header window keys

System Level

Table 3-2 lists and defines the functions of the keys used at the System level.

Table 3-2. System Level Keys

SYSTEM LEVEL KEYS	
	Moves the block cursor up one menu item. Moves the block cursor down one menu item. Selects the specified file or part name.

Header Window

When the JEDEC file header information is displayed, i.e., you are in the Header window, the following keystroke conventions apply (see Table 3-3):

Table 3-3. Header Window Keys

JEDEC FILE HEADER TEXT WINDOW KEYS	
	Moves the cursor to the beginning of the previous line.
	Moves the cursor to the beginning of the next line.
	Moves the cursor to the beginning of the next line.
	Moves the text from the cursor to the end of the line one position to the right.
	Removes the character to the left of the cursor and shifts the rest of the line one position to the left to close the gap.
	Moves the cursor to the beginning of the first line.
	Moves the cursor to the beginning of the last line.

Device Level Keys

Table 3-4 lists and defines the functions of the keys used at the Device level.

Table 3-4. Device Level Keys

DEVICE LEVEL KEYS	
↑	Moves the box cursor up one item.
↓	Moves the box cursor down one item.
→	Moves the box cursor one position to the right.
←	Moves the box cursor one position to the left.
H	Opens the Help window. The information displayed in the Device Level Help window depends on the part selected. (Press any key to continue.)
Alt + F9	Toggles the Security option.
Alt + F10	Toggles the Turbo-Bit option.

Group Level Keys

Table 3-5 lists and defines the functions of the keys used at the Group level.

Table 3-5. Group Level Keys

GROUP LEVEL KEYS	
↑	Moves the box cursor up one item in the current column.
↓	Moves the box cursor down one item in the current column.
←	Moves the box cursor left one item in the current row.
→	Moves the box cursor right one item in the current row.
↵	Toggles the architecture configuration (e.g., from inverted to non-inverted; from ROLF to ROIF, etc.) or opens the Macrocell window.
Ins	Inserts output inversion, or adds changes to the next architecture. Select the INV or architecture location, then press <Ins>.
Del	Removes output inversion, or deletes changes to the previous architecture. First select the INV or architecture location with the box cursor, then press <Del>.
H	Opens the Help window. (Press any key to continue.)
CTRL + F5	Restores JEDEC data associated with the current display group to their previous condition, i.e., before the Group Level window was opened.
F9	Displays the next group in the device.
F10	Displays the preceding group in the device.

Macrocell Level Keys

Table 3-6 lists and defines the function keys used at the Macrocell level.

Table 3-6. Macrocell Level Keys (Part 1 of 2)

MACROCELL LEVEL KEYS (Part 1 of 2)	
	Moves the crosshairs cursor up one line.
	Moves the crosshairs cursor down one line.
	Moves the crosshairs cursor one position to the left.
	Moves the crosshairs cursor one position to the right.
	Moves the crosshairs cursor to the beginning of the next product term.
	Moves the crosshairs cursor eight position to the right.
	Moves the crosshairs cursor to the upper left corner of the screen.
	Moves the crosshairs cursor to the lower right corner of the screen.
	Sets the cell at the current cursor position.
	Clears the cell at the current cursor position.
	Sets the cell at the current cursor position, then moves one position to the right.
	Sets the cell at the current cursor position, then moves one position to the left.

Table 3-6. Macrocell Level Keys (Part 2 of 2)

MACROCELL LEVEL KEYS (Part 2 of 2)	
Space	Clears the cell at the current cursor position, then moves one position to the right.
Back Space	Clears the cell at the current cursor position, then moves one position to the left.
H	Opens the Help window. (Press any key to continue.)
Shift F1	Sets the entire macrocell.
Shift F2	Sets the entire product term.
Shift F3	Clears the entire macrocell.
Shift F4	Clears the entire product term.
Shift F5	Copies the last saved macrocell over the current one. If the saved one has fewer product terms than the current one, subsequent product terms are cleared. Used to transfer data between macrocells.
Shift F6	Copies the last saved product term over the current one. This function may be used to transfer data between product terms.
F7	Saves the macrocell for future recall.
F8	Saves the product term for future recall.
F9	Displays the next macrocell in the group.
F10	Displays the preceding macrocell in the group.

LogicMap II Messages

LogicMap II messages are listed alphabetically below. Each message is accompanied by an explanation and suggestion for corrective action.

<str> indicates text string
<n> indicates number

ABORTING NON-STANDARD FILE

CAUSE: You have specified a JEDEC file that does not have a standard header character.
ACTION: Specify the correct file.

Bad EPLD.SYS version — you must install A+PLUS software

CAUSE: The file EPLD.SYS on your system is not the correct version.
ACTION: Install the A+PLUS software to create the correct version of EPLD.SYS.

Can't find EPLD.SYS — you must install A+PLUS software

CAUSE: The file EPLD.SYS cannot be located.
ACTION: Install the A+PLUS software to create the EPLD.SYS.

Can't open EPLD.SYS — you must install A+PLUS software

CAUSE: The file EPLD.SYS may have been corrupted.
ACTION: Go through the A+PLUS software installation procedure to create the EPLD.SYS.

ERROR OPENING JEDEC FILE

CAUSE: The specified file could not be opened.
ACTION: Make sure that the specified directory and filename are correct. Also check disk space with the **DIR** command.

ERROR reading Device File
Bad ECF version: <version number>

CAUSE: The specified device file is incompatible with this version of LogicMap II.
ACTION: Be sure to use the correct version of ECF and LogicMap II. They are available on your distribution diskettes.

ERROR reading Device File
Corrupt ECF file: <version number>

CAUSE: The specified device file is corrupted.
ACTION: Reload it from the distribution diskette.

ERROR reading Device File
File is not an ECF file

CAUSE: The specified device file is not in the valid format.
ACTION: Load the correct file from the distribution diskette.

ERROR reading Device File
Premature EOF encountered

CAUSE: The End of File was reached before reading was completed.
ACTION: Try to re-install the Device File from the distribution diskette.

ERROR READING JEDEC FILE

CAUSE: The file specified is not a standard JEDEC file. Maybe the special header character is missing or you edited the file after it was processed by the Design Processor.
ACTION: Rerun the ADF to create a new JEDEC file.

ERROR WRITING JEDEC FILE
FILE IS TOO LARGE

CAUSE: You may not have enough disk space or the diskette may be bad.
ACTION: Check your available disk space and the diskette.

ERROR WRITING JEDEC FILE
INSUFFICIENT DISK SPACE

CAUSE: You don't have enough available disk space.
ACTION: Try to free some disk space.

ERROR WRITING JEDEC FILE
I/O ERROR: <error number>

CAUSE: The JEDEC file was not written.
ACTION: Check to see whether the diskette is inserted properly.

Errors during verify or security protected!!
Do you wish to view error data?

CAUSE: The device data did not match the JEDEC data during the verify process.
ACTION: Type **Y** if you want to see the data. When you view them, be aware that the resulting data are the logical Exclusive-OR of the device data and the JEDEC data.

Internal Error: <n> or <str>

CAUSE: There is an internal error.
ACTION: For assistance, please call Altera Applications and provide the error number/text (408-984-2805 ext.102).

INVALID DRIVE NAME

CAUSE: You entered an invalid drive letter.
ACTION: Pound your chest and shout, "Mea Culpa!"

INVALID PATH NAME

CAUSE: You entered an invalid path name.
ACTION: Not all paths lead to Rome—pick the correct one.

JEDEC address out of range

CAUSE: You probably specified the wrong device for the JEDEC file.
ACTION: Check the specified device.

JEDEC File not found

CAUSE: The JEDEC file you specified while in Program Device, Verify Device, or Examine Device mode was not found.
ACTION: Be sure you entered the name correctly. You may also go back to Edit JEDEC File mode to check the filename entry.

JEDEC File not found
Create a new file? [Y/N]

CAUSE: While in Edit JEDEC File mode you specified a JEDEC file that was not found.
ACTION: Type **Y** if you wish to create a new file.

JEDEC File size does not match requested device

CAUSE: You probably specified the wrong device for the JEDEC file.
ACTION: Check the specified device.

Location programming failure or security protected

CAUSE: The number of verify failures exceeded the algorithm limits. The cause may be a programming error, or you may have attempted to program a device that is already security protected.
ACTION: Be sure the JEDEC file specified corresponds to the device in the socket and the device is not security protected.

MISMATCHED EPROM BIT COUNT

CAUSE: The number of EPROM bits specified by the JEDEC file does not match the number of EPROM bits in the specified device.
ACTION: Be sure to specify the correct device for the JEDEC file.

Multiple locations failure or security protected

CAUSE: The number of verify failures exceeded the algorithm limits. The cause may be a programming error, or you may have attempted to program a device that is already security protected.

ACTION: Be sure the JEDEC file specified corresponds to the device in the socket and the device is not security protected.

Not blank, hardware failure, or security protected

CAUSE: The device in the socket is not blank, or it is security protected, or there is a hardware problem.

ACTION: If you have already erased the device as recommended, the error is most likely caused by a hardware problem. Be sure to check all your cable connections. Call Altera Applications for assistance (408-984-2805 ext.102).

NOT ENOUGH MEMORY

CAUSE: You ran out of memory.

ACTION: You need to remove other resident programs to free some memory.

Programmer not available...
Hit any key to continue...

CAUSE: You tried to program, verify, or examine a device but do not have the programmer card installed.

ACTION: Install the card.

Programmer self test failed
Device must NOT be in socket for this test to pass
Enter:
C to continue without programming card
T to run diagnostics again
Q to return to operating system

CAUSE: LogicMap II did not recognize either LP3 or LP4. Your programming card is either bad or not installed.

ACTION: Install the card or call Altera Applications (408-984-2805 ext.102).

Programming failure or security protected
Failing location: row: <row n> col: <column n>
Programming data: <data> Part data <data>

CAUSE: LogicMap II was not able to successfully program the above location. The chip may have been electrically damaged before you tried to program it.

ACTION: Erase the chip and start again. Or try another blank chip. If you have already erased the device, the error is most likely caused by a hardware problem. Call Altera Applications for assistance (408-984-2805 ext.102).

SUMCHECK ERROR <hex number> <hex number>

CAUSE: The checksum at the end of the JEDEC file does not match the calculated checksum. The file may have been corrupted.

ACTION: Recreate the JEDEC file.

Test indicates that the device is inserted backwards or that the socket is not connected. Please remove the device and reinsert it and check the cable connections.
Strike any key when ready, ESC to exit.

ACTION: Insert the device properly.

Unable to find Device File

CAUSE: The file describing the specified device was not found.

ACTION: Type in the device name correctly.

Unrecognized <device name> device!

CAUSE: LogicMap did not recognize the ID on the device as a valid ID.

ACTION: Make sure you have a valid Altera device in the socket.

You have specified <device name> as the device you wish to use. Please verify the device is an <device name>. If it is not the correct device, it may be damaged.
Is the device in the socket an <device name> [Y/N]?

CAUSE: This is a warning message. It may be displayed when you are in Program Device, Verify Device, or Examine Device mode.

ACTION: Make sure you have the correct device in the socket. If you don't and you proceed, you may damage the device.

APPENDIX A

LP3/LP4 Dip Switch Address Maps

Dip switch settings for the LP3 Logic Programmer card are as follows:

Location of Base (Range : 280 to 2BF)	**Bit Settings** (1 = ON ; 0 = OFF)			
(Required address space = 4 locations)	**1**	**2**	**3**	**4**
280 (default)	1	1	1	1
284	0	1	1	1
288	1	0	1	1
28C	0	0	1	1
290	1	1	0	1
294	0	1	0	1
298	1	0	0	1
29C	0	0	0	1
2A0	1	1	1	0
2A4	0	1	1	0
2A8	1	0	1	0
2AC	0	0	1	0
2B0	1	1	0	0
2B4	0	1	0	0
2B8	1	0	0	0
2BC	0	0	0	0

Dip switch settings for the LP4 Logic Programmer card are as follows:

Location of Base (Range : 200 to 2FF)	**Bit Settings** (1 = ON ; 0 = OFF)			
(Required address space = 16 locations)	**1**	**2**	**3**	**4**
270	0	0	0	0
260	1	0	0	0
250	0	1	0	0
240	1	1	0	0
230	0	0	1	0
220	1	0	1	0
210	0	1	1	0
200	1	1	1	0
2F0	0	0	0	1
2E0	1	0	0	1
2D0	0	1	0	1
2C0	1	1	0	1
2B0	0	0	1	1
2A0	1	0	1	1
290	0	1	1	1
280 (default)	1	1	1	1

Glossary

.JED (JEDEC file) An industry-wide standard for the transfer of information between a data preparation system and a logic device programmer. The Altera design Processor (ADP) converts your input Altera Design File (ADF) into a JEDEC file that describes your design.

A+PLUS (Altera Programmable Logic User System) A set of computer programs and hardware support devices that facilitate design and implementation of custom logic circuits with Altera programmable logic products.

ADP (Altera Design Processor) That portion of A+PLUS that processes the Altera Design File. It is the user-interface to the A+PLUS software. It controls the individual modules and converts the ADF into a JEDEC file which is used to program an Altera EPLD.

AUTOEXEC.BAT A DOS system file that is modified during installation of A+PLUS and SAM+PLUS software. (The original AUTOEXEC.BAT file is saved as AUTOEXEC.BAK.) This file is executed by DOS whenever the computer is booted.

Automatic Part Selection A design processing option that directs the Altera Design Processor and SAM Design Processor to select an appropriate part, based on the number of inputs, outputs, and resources used in the design. It is implemented by entering "AUTO" in the Part section of the ADF, SMF, or ASM file.

BUSTER The Altera family of programmable bus peripherals.

CONFIG.SYS A DOS system file that is modified during installation of A+PLUS and SAM+PLUS software. (The original CONFIG.SYS file is saved as CONFIG.BAK.) DOS uses this file to set up the DOS environment.

DASH FutureNet's schematic capture program.

Device Level That level of LogicMap II at which you set Turbo-Bit and Security options for most Altera parts. At this level, you also select the group or portion of a part which you wish to view at the next lower (Group) level.

DIP Dual In-line Package.

ECF (Environment Configuration File) One of several files needed to run LogicMap II. All ECFs are supplied on a distribution diskette that comes with your LogicMap II package.

EOF (end-of-file) A control character indicating that the last record in a file has been read.

EOL (end-of-line) Equivalent to **<CR> <LF>**.

EP310/320 These Altera parts are available in 20-pin DIP packages that provide 10 dedicated input and 8 programmable I/O pins. Each I/O pin is associated with a macrocell with 8 product terms and an I/O architecture control block. Each macrocell may be configured to provide registered or combinatorial output. Feedback from each macrocell can be combinatorial, regardless of how the output is configured. A dedicated clock input—pin1—feeds the macrocell registers and can also feed the product term array. Output Enables for each macrocell are supported by single product terms. The EP310 also provides Clear and Preset signals to the macrocells, each of which is supported by a single product term.

EP600/610 These Altera parts are available in 24-pin DIP or 28-pin J-lead packages that provide 4 dedicated input, 2 dedicated clock input, and 16 programmable input/output pins. Each I/O pin is associated with a macrocell with 10 product terms and an I/O architecture control block. Eight product terms are used for logic requirements, 1 for Output Enable/Asynchronous Clock implementation, and 1 for flipflop Clear control. The I/O architecture control block allows selection of D, T, JK,

or SR flipflops or combinatorial output. Each macrocell allows programmable selection of a combinatorial output path with registered or I/O feedback. The 2 dedicated clock pins provide positive-edge-triggered synchronous clock input to the registers.

EP900/910 These Altera parts are available in 40-pin DIP or 44-pin J-lead packages that provide 12 input and 24 programmable input/output pins. Each I/O pin is associated with a macrocell with 10 product terms and an I/O architecture control block. Eight product terms are used for logic requirements, 1 for Output Enable/Asynchronous Clock implementation, and 1 for flipflop Clear control. The I/O architecture control block allows selection of D, T, JK, or SR flipflops or combinatorial output. Each macrocell allows programmable selection of a combinatorial output path with registered or I/O feedback. The 2 dedicated clock pins provide positive-edge-triggered synchronous clock input to the registers.

EP1210 This Altera part is available in a 40-pin DIP or 44-pin J-lead package providing 12 input pins, 1 dedicated clock input pin, and 24 programmable input/output pins. Inputs may be latched or unlatched. Each I/O pin is associated with a macrocell that has from 4 to 12 product terms and an I/O architecture control block. To conserve I/O architecture resources, macrocells are arranged in groups of 4. Four of the 8 macrocell groups have local feedback capability, which allows them to direct feedback to a specific group of macrocells. In addition, 4 buried macrocells that are not associated with any pin may be usd to implement internal logic. These buried macrocells also support combinatorial feedback.

EP1800 This Altera part is available in 68-pin J-lead or 68-pin grid-array package that provides 16 dedicated input and 48 programmable input/output pins. It consists of 4 quadrants, each of which contains 12 macrocells. Each I/O pin is associated with a macrocell with 10 product terms and an I/O architecture control block. Eight product terms are used for logic requirements, 1 for Output Enable/Asynchronous Clock implementation, and 1 for flipflop Clear control. The I/O architecture control block allows programmable selection of D, T, JK, or SR flipflops or combinatorial output for each of the 48 macrocells. In all macrocells, the feedback path can be programmed to be registered, combinatorial, or I/O. Some of the global macrocells can be programmed as local macrocells that support dual I/O feedback, i.e., simultaneous internal and I/O feedback. One input pin in each quadrant may optionally be used a a synchronous clock input. Each macrocell also has an asynchronous clock option that allows both synchronous and asynchronous operation on the same EPLD.

EPB1400 (BUSTER) The BUSTER part is available in a 40-pin DIP or 44-pin J-lead package. It supports 8 dedicated input, 8 bus port, and 20 programmable input/output pins. BUSTER operation is controlled by 2 key functional blocks, the Microprocessor Interface Block (MIB) and

the Programmable Logic Core Block (PLCB).

The PLCB controls 10 general-purpose macrocells in each half of the device. Eight of these macrocells are associated with the byte-wide input and output latches of the MIB. Each of the 20 general-purpose macrocells uses 8 product terms for logic requirements, 2 for Output Enable/Asynchronous Clock implementation, and 1 for flipflop Clear control. Each macrocell may be configured with programmable D, T, JK, or SR flipflops or combinatorial output; with registered, combinatorial, or I/O feedback; and with programmable clocks. Dual I/O feedback is also available, allowing each macrocell to be used internally for buried logic functions, while the associated macrocell pin is used as an input pin. Each macrocell has an asynchronous operation on the same chip.

The MIB architecture provides access to an external microprocessor bus via 5 byte-wide elements: 1 bi-directional bus port with 8 dedicated pins, 2 input flipflops that can be configured as latched or edge-triggered, and 2 output latches. These 5 elements are controlled by a Read Strobe pin, a Write Strobe pin, and 7 Control macrocells.

EPLD Erasable Programmable Logic Device, i.e., any Altera part.

EPLD.SYS The A+PLUS and SAM+PLUS hardware installation file. You may access this file through the installation menu if you wish to change the address location of the programming card or disable the color display option of LogicMap II.

EPS444, EPS448 High-speed Standalone Microsequencers (SAMs) for applications requiring up to 448 lines of microcode, up to 12 outputs, 15-level-deep subroutining, and versatile counting and logic-flag capabilities. The EPS444 is available in a 24-pin DIP package and provides 8 input and 12 user-definable output pins. The EPS448 is available in a 28-pin DIP package or 28-pin J-lead package and provides 8 input and 16 user-definable output pins. The fully synchronous design has 1 external clock. All flipflops are positive-edge triggered. Both parts support 8 powerful operating codes that define the source of the microcode address, the transfers among internal registers and the external pins, and the control signals for the chip resources.

Group Level That level of LogicMap II that displays a block representation of each macrocell, an indication of output inversion, and the architectural configuration associated with a particular group. At this level, you also select the macrocells you wish to view at the next lower (Macrocell) level.

JEDEC file (.JED) An industry-wide standard for the transfer of information between a data preparation system and a logic device programmer. The ADP or SDP converts your input ADF, SMF, or ASM file into a JEDEC file that describes your design.

LHS Left-hand side.

LogiCaps Altera's logic schematic capture program.

LogicMap II The interface between the JEDEC file and the programming unit. The LogicMap II program allows you to program any Altera part. Data are stored in standard JEDEC format.

product term (p-term) Two or more factors in a Boolean expression combined with the AND-operator constitute a product term, "product" meaning "logic product." (See also George Boole's "The Laws of Thought" [1854].)

Logic Programmer Card The expansion card required to program an EPLD with the LogicMap II program. It occupies a single slot in the computer.

LP3 Programmer Card One of the two available Altera expansion cards required to program an EPLD with the LogicMap II program. It occupies a single slot in the computer.

LP4 Programmer Card One of the two available Altera expansion cards required to program an EPLD with the LogicMap II program. It occupies a single slot in the computer.

Macrocell A basic building block of Altera's programmable logic devices. It consists of two sections: combinatorial logic and output logic. The combinatorial logic allows a wide variety of logic functions. The output logic has two data paths: one leads to the other macrocells or feeds back to the macrocell itself; the other is configured as a pin connection acting as input, output, or bi-directional I/O port on the chip.

Macrocell Level The lowest level in the LogicMap II hierarchy at which you may edit product term bits.

Master Programming Unit The logic device programming box supplied by Altera. It contains zero insertion-force sockets into which the Altera parts are inserted. An indicator lamp on the unit shows when the unit is active. Add-on adaptors accommodate programming of the entire family of Altera parts.

PLASAP The Altera Development System that includes a programming card, the Master Programming Unit, LogicMap II software, and documentation, i.e., the hardware package for Altera development systems.

PLCAD4 A fully integrated Altera Development System that includes the LogiCaps schematic capture program, A+PLUS software, and the TTL MacroFunction library, as well as a programming card, the Master Programming Unit, LogicMap II software, and all documentation.

PLCAD-SUPREME Currently the most comprehensive Altera Development System. It includes all A+PLUS programs and support files, as well as the hardware required for running these programs, including sample devices and all necessary adaptors.

PLDS2 The basic Altera Development System that includes the hardware and software supplied with PLASAP, as well as A+PLUS software.

PLE-3 Altera's earlier version of the programming unit. It enables you to program EP310, EP320, EP1200, and EP1210 parts.

PLE3-12 Altera's Master Programming Unit with which you program all Altera parts. It contains 20-pin and 40-pin zero-insertion force sockets and a 30-inch ribbon cable terminated with a 25-pin D-type connector. An LED on the top of the unit lights up when the unit is active.

READ.ME File A file on the A+PLUS INSTALL and the SAM+PLUS INSTALLS distribution diskettes. It contains information on any changes made to the installation procedure since the release of the SAM+PLUS documentation, and should be read before A+PLUS and SAM+PLUS are installed.

SAM Altera's Standalone Microsequencer parts. (See *EPS444, EPS448.*)

SAM Design Processor (SDP) The component of SAM+PLUS that processes an Assembly Language File (ASM file). It is the user interface to the SAM+PLUS software. It controls the individual modules and converts the ASM into a JEDEC file used to program a SAM part.

SAM+PLUS Altera's Standalone Microsequencer Programmable Logic User System. SAM+PLUS is a set of computer programs and hardware support products that facilitates design and implementation of custom control circuits with Altera SAM parts.

SDP See *SAM Design Processor.*

Security bit A control bit that, when set to ON, prevents interrogation or inadvertent reprogramming of an Altera EPLD .

sum-of-products A Boolean expression is said to be in sum-of-products form if it consists of product terms combined with the OR-operator.

System Level That highest, most general operational level of LogicMap II. It is displayed when you invoke LogicMap II. At this level, you change paths and directories, edit a JEDEC file, and select the levels that let you program, verify, and examine an Altera part.

Turbo-Bit A control bit that, when set to ON, allows you to choose the speed and power characteristics of an Altera EPLD.

Index

D

E

F

G

H

I

R

S

T

U

V