XILINX®

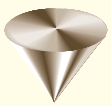
ONLINE

XEPLD
REFERENCE GUIDE

FOR WINDOWS

TABLE OF CONTENTS

INDEX



GO TO OTHER BOOKS

XACT™ STEP

0401306

Contents

Chapter 1

Design Flow

This chapter gives an introduction to the Design Manager design flow, including:

- How to prepare behavioral, schematic, and VHDL designs.
- How to translate designs into the required format.
- How to fit designs into a target device (or devices).
- Alternatives for simulation, programming, and reporting of designs that have been processed.
- A quick tutorial with a simple design example.

Overview of Design Processing

There are seven basic steps in processing your design:

1. Design Entry to create your design source file.
2. Project Creation to open a new project file.
3. Translation of your design file into Xilinx's internal .XFF format.
4. Implementation of your design into a specific target device.
5. Export of your design for timing simulation and programming.
6. Report Generation showing the status of your design.
7. Timing Analysis for design verification.

Design Entry

You can use a variety of schematic, behavioral, and VHDL tools to create your design.

The supported schematic entry tools are:

- Viewlogic.
- OrCAD.
- Mentor Graphics.

The supported behavioral entry tools are:

- Xilinx ABEL.
- PLUSASM.

The supported VHDL entry tools are:

- Viewlogic ViewSynthesis.
- Synopsys.

Prepare your design for input to the Design Manager as follows:

- For schematic designs — Save your design using the native file extension (.1 for Viewlogic or .SCH for OrCAD.
- For Xilinx ABEL designs — Generate an EPLD netlist for a stand-alone design (this saves the design as a .PLD file).
- For Viewlogic Synthesis or Synopsys designs — Save your design as an .XNF file.

Project Creation

This step is only done once for each project, to create the files which will contain all information about your design. Design translation, which is explained in the next section, is performed automatically when you create a new project file.

To create a new project do the following, after you have created a design source file:

1. Start the Design Manager.
2. Select the New Project command from the File menu. The New Project window appears.
3. Select XC7000 as the Target Family.
4. Select the Browse button to the right of the Input Design field. The Open browser window appears.

5. Change the List Files of Type value to Any File (*.*).
   - PLUSASM files have a .PLD extension.
   - OrCAD schematic files have a .SCH extension.
   - Viewlogic schematic files have a .1 extension.
   - All other files have an .XNF extension.
6. Select the drive if necessary, the directory, and the design file. (Viewlogic designs use the WIR directory.)
7. Select the OK button to close the browser.
8. Select the Translate button to close the New Project window.
9. The Translate Options window appears. If you did not specify the device type in the design file using the PART attribute, select the Read Part from Design box to remove the check. Click OK.
10. A translation window appears, which shows the translation of the design file into a standard Xilinx .XFF format, ready for processing. When these translations are complete, a message box appears telling you that the translation completed successfully. Click OK.

Design Translation

You must first translate your source design into the standard Xilinx data format (.XFF) before the software can process your design. For new projects, design translation is performed automatically when the project is first created. For existing projects, you must do design translation each time you make a change to your design source file. Figure 1-1 shows the basic flow.

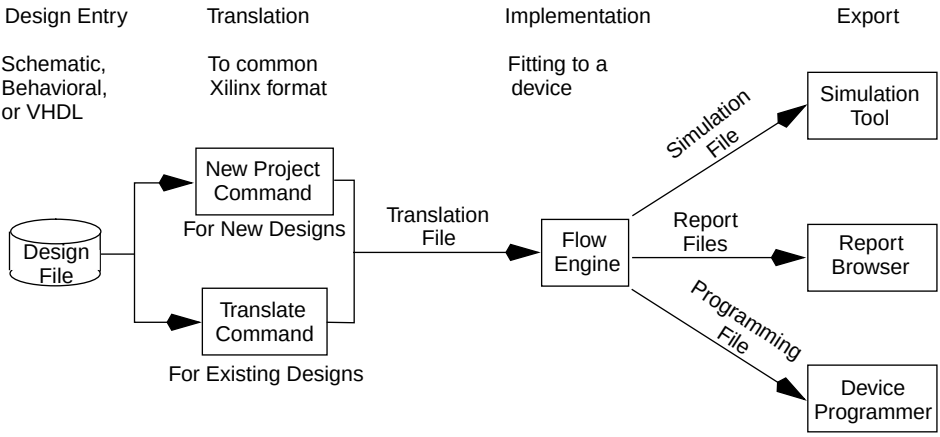


Figure 1-1 Design Manager Design Flow

To translate your design (after making changes to your source file) do the following:

1. Start the Design Manager.
2. Select the Translate command from the Design menu. The Translate Options window appears.
   - If you have want the software to read the target device part type from your design file, select Read Part From Design.
   - If you want to select a specific target device type, click Select Part and you will see the Part Selector window. From here you can choose any available device.

Design Implementation and Device Programming

Design implementation is the process of fitting your design into a target device. To implement your design:

1. Select the Implement command from the Design menu. The XC7300 Design Implementation Dialog box appears with the following options:
   - Produce Timing Report — If this box is checked, a timing report showing the calculated worst-case timing for your design is created.

- Produce Timing Simulation Data — If this box is checked, the Flow Engine produces a timing simulation file in the appropriate format and copies this file into your design directory.
- Produce Configuration Data — If this box is checked, the Flow Engine produces an Intel HEX file that you can use to program your device. For programming instructions, refer to the *HW130 Programmer User Guide*.

2. Select the Run button to start the Flow Engine. The Flow Engine window appears, displaying the stages in the design processing.

See Chapter 2 for more information about the Design Manager and the Flow Engine. For information about simulation, see the Interface User Guide for your system.

Report Generation

The Flow Engine produces several reports, which are described in more detail in the "Reports" chapter.

- Translation Report — Shows the results of the translation process.
- Resource Report — Lists the device resources used by the design and the resources remaining.
- Pinout Report — Shows how the external nets in your design were mapped to the device pins.
- Mapping Report — Tells you how the logic in the design was mapped to the device.
- Fitting Report — Shows the allocation of Function Block resources.
- Optimization Report — Shows you how logic was optimized for your design.
- Timing Report — Shows the calculated worst-case timing for the logic paths in your design.
- Timespec Report — Tells you if you have made any errors in assigning T-Spec attributes in your design and lists the T-Specs that the fitter used.
- Multi-chip I/O Report — Shows the pinouts of each device

resulting from multiple chip partitioning. This report is available only for multi-chip designs.

Timing Analysis

You can analyze the timing of your implemented design using the Timing Analyzer. To open the Timing Analyzer, double click on its icon; the current design in the Design Manager is automatically loaded.

You can choose from the following timing reports, which are listed on the Analyze menu:

- Performance to TimeSpecs — Compares the implementation of the design with the imposed timing constraints. (For more information about timing constraints, see the "Controlling Design Timing" chapter.)
- Performance Summary — Summarizes the overall design performance.
- Detailed Path Report — Lists worst-case path delay information for all paths in the design that have not been excluded by filters.
- Check for Asynchronous Logic — Lists signals that are possibly asynchronous and the product-term clock signals that control them.
- Show Clocks — Lists the clock signals in the design.
- Show Settings — Lists the current settings of the timing analyzer, including which path filters have been applied.

You can apply filters to include in the reports only the paths you are interested in. You can also change other options that affect timing, such as the speed grade of the device. For more information, see Chapter 5.

Quick Tutorial

This section shows you how a 4-bit Johnson counter design is processed through the Design Manager. The schematic version of this design is shown in Figure 1-2.

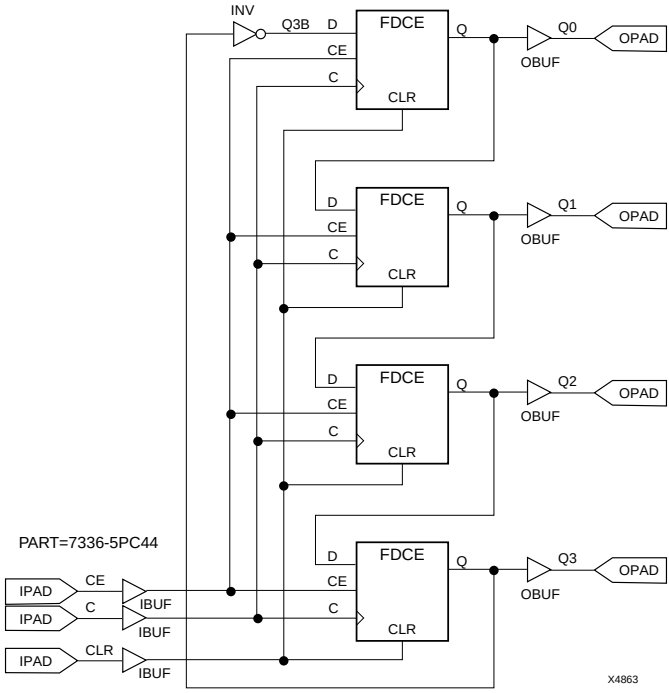


Figure 1-2 Example 4-Bit Johnson Counter Design (jcount)

Enter the Design

Create the design in your design entry tool and save it to a file named "jcount." Be sure to include the target device part type in your source file. The design entry software will add the appropriate file extension to the design file.

Create a New Project

1. Start the Design Manager. The initial Design Manager window, before you have opened a design, looks as shown in Figure 1-3.

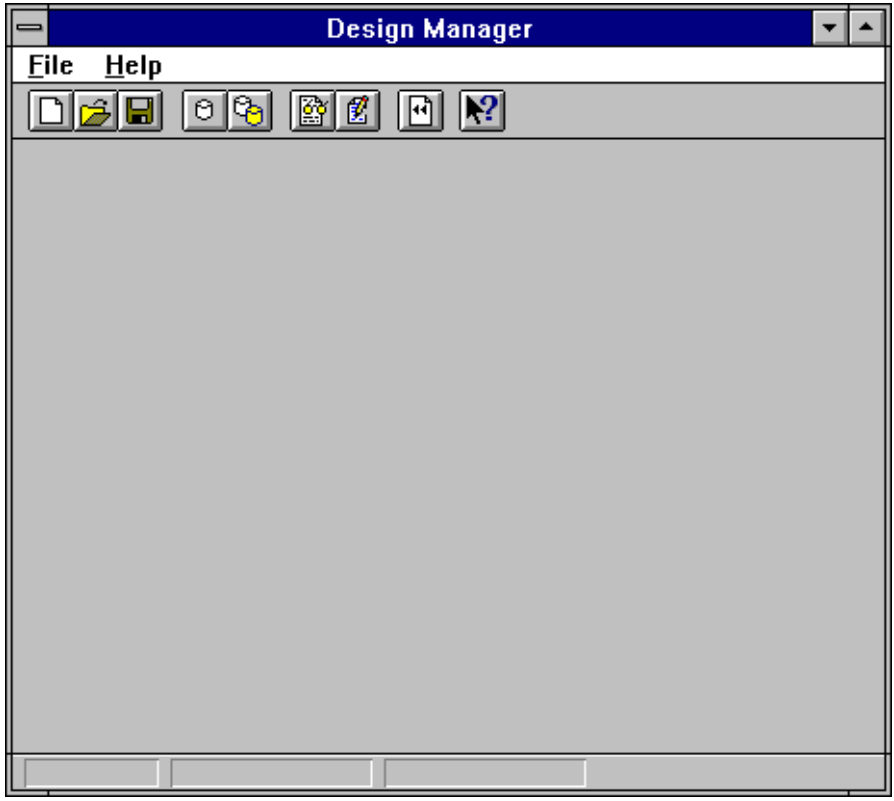


Figure 1-3 The Design Manager Window with No Design

2. Select the New Project command from the File menu. The New Project window appears as shown in Figure 1-4.

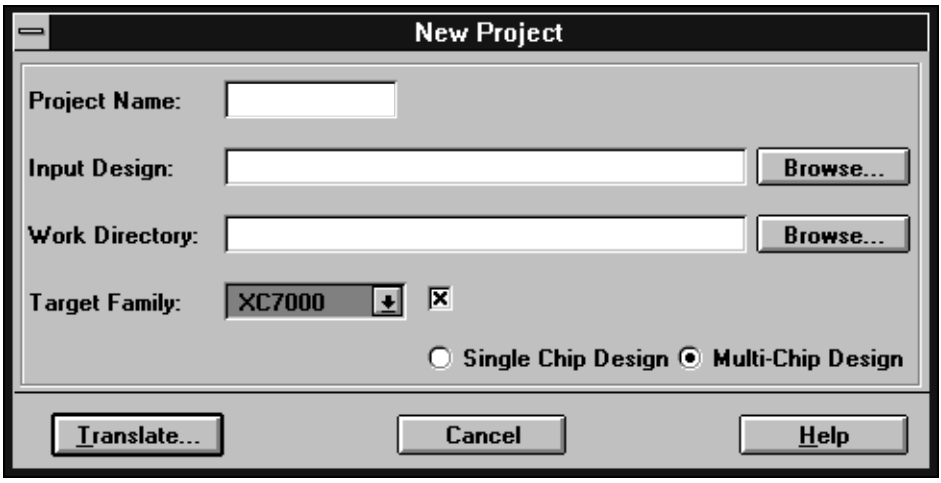


Figure 1-4 New Project Window

3. Select XC7000 as the Target Family.

4. Select the Browse button to the right of the Input Design field. The Open browser window appears as in Figure 1-5.

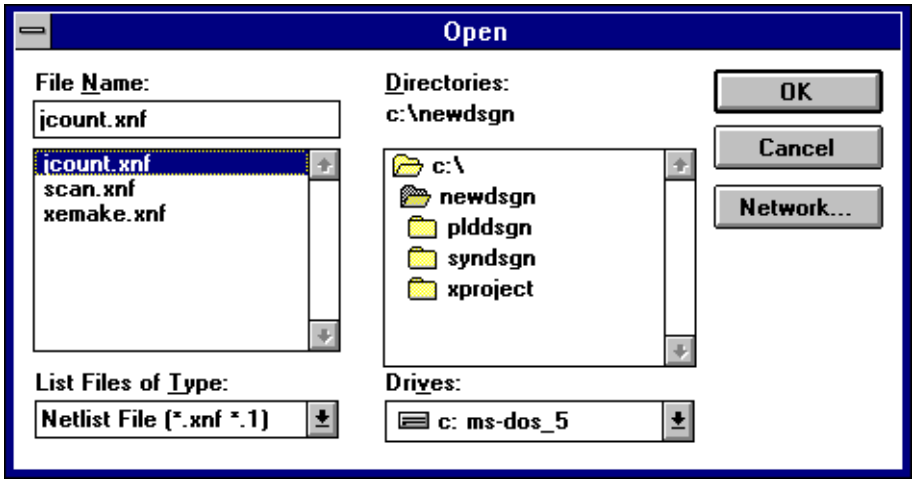


Figure 1-5 New Project Browser

5. For an OrCAD design or a behavioral design, change the List Files of Type value to Any File (*.*). OrCAD schematic files have a .sch extension and PLUSASM files have a .PLD extension. Other design files have extensions of .1 (for Viewlogic) or .XNF (generic).
6. Select the drive if necessary, the directory (the WIR directory for a Viewlogic design), and the design file.
7. Select the OK button to close the browser. Select the Translate button to close the New Project window.

8. The Translate Options window appears as in Figure 1-6:

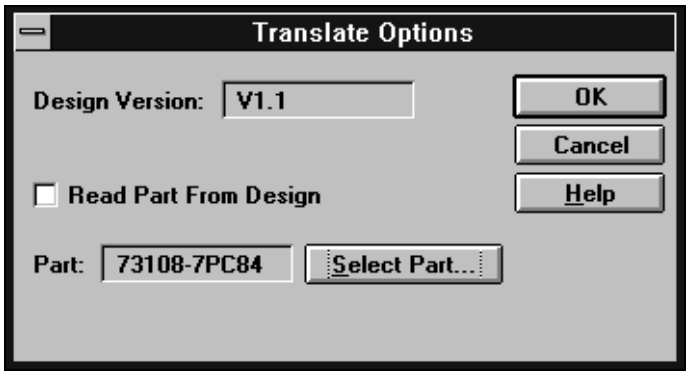


Figure 1-6 Translate Options Window

- If you specified a target device in your source file (and you want to use that device) select Read Part From Design.
- If you did not specify the device type in your design source file, remove the "X" from the Read Part From Design box. Then, specify the target device type by clicking on Select Part. You will see the Part Selector window as shown in Figure 1-7.

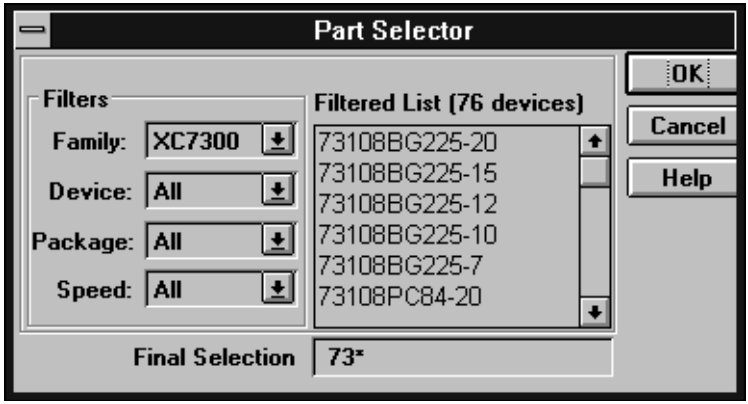


Figure 1-7 Part Selector

Select XC7336PC44-5 and click OK.

9. A translation window appears, which translates the design file into an .XNF and an .XFF file. When these translations are complete, a message box appears telling you that the translation completed successfully. Select the OK button.

   After your design has been loaded, the Design Manager window looks as shown in Figure 1-8.

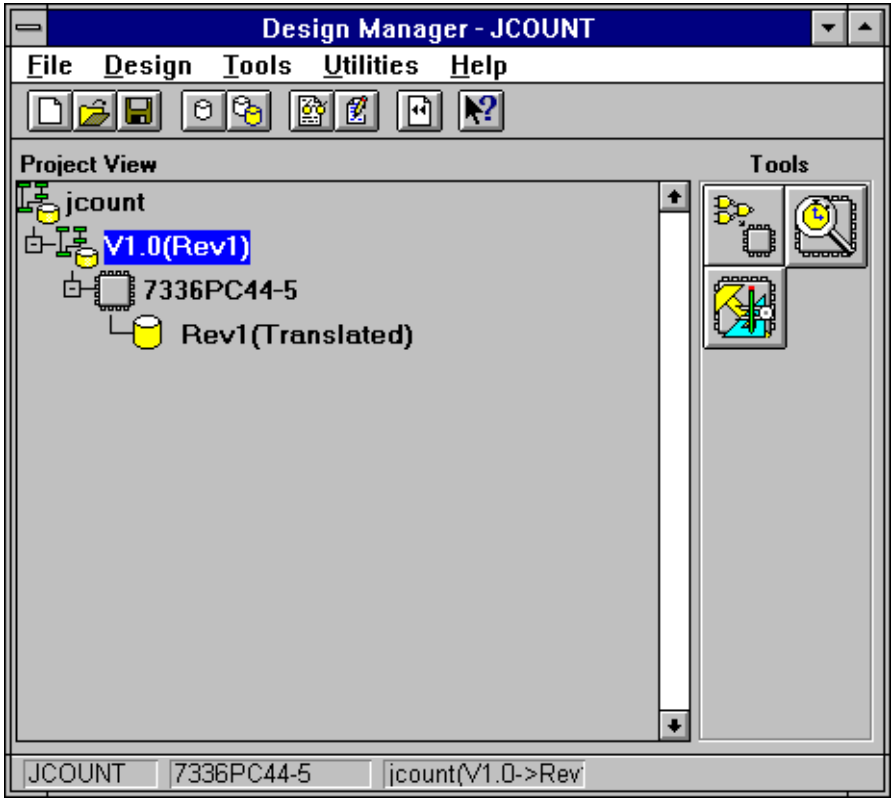


Figure 1-8 The Design Manager Window with a Design Loaded

Implement the Design

To implement the design, follow these steps:

1. Select the Implement command from the Design menu. The XC7300 Design Implementation Dialog box appears as shown in Figure 1-9.

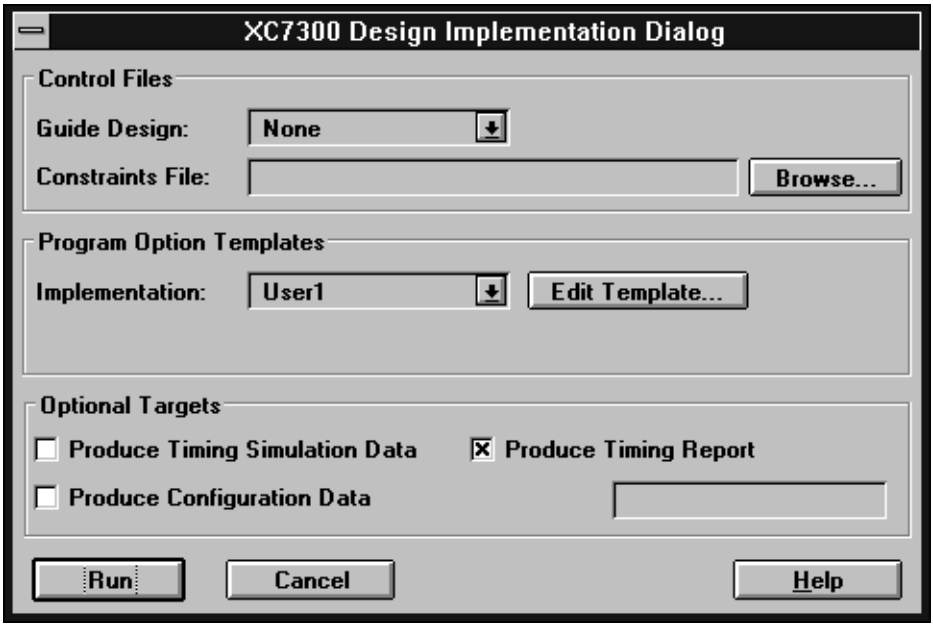


Figure 1-9 XC7300 Design Implementation Dialog Box

2. Select Produce Timing Report, Produce Timing Simulation Data, and Produce Configuration Data options.
3. Click Run to implement the design. The Flow Engine window appears as in Figure 1-10, displaying the stages in the design processing. As reports are generated, you see their icons appear in the Report Browser window.

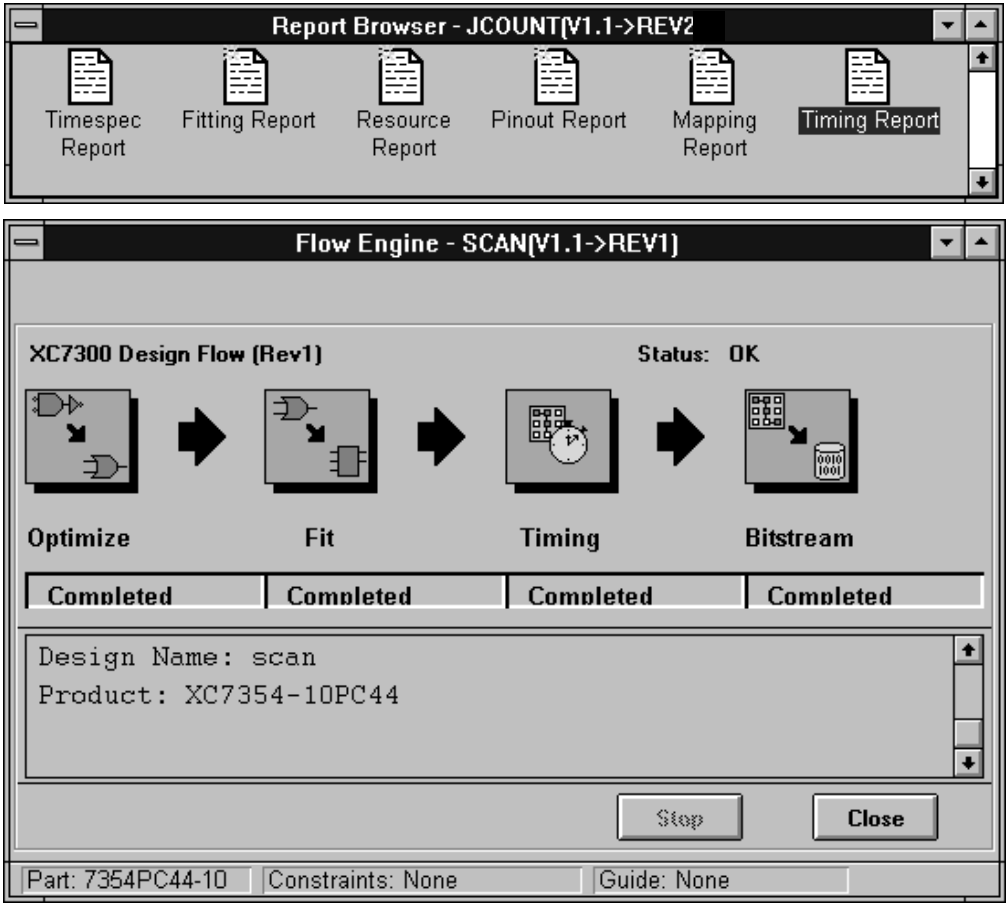


Figure 1-10 Flow Engine Window (Non-Interactive)

4. When processing is complete, you can view the fitter reports (including the timing report) as described in the next section, perform timing simulation, and program the device. Timing simulation is described in the Interface User Guide for your system. Device programming is described in the *HW130 Programmer User Guide*.

Use the Report Browser

The Report Browser opens automatically when you select the Implement command. (To manually open the report browser, select the Report Browser command from the Utilities menu.) The Report Browser window appears as shown in Figure 1-11.

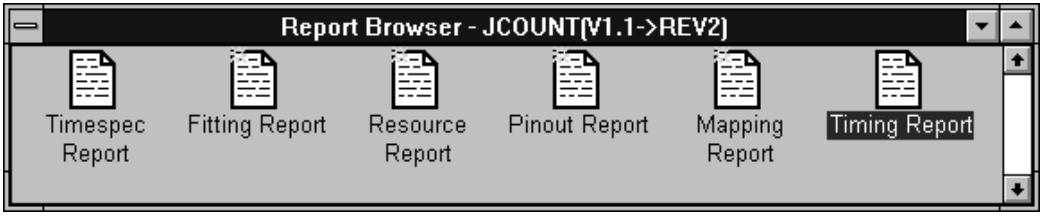


Figure 1-11 Report Browser

Double click on the Resource Report icon. The report appears in its own window, as shown in Figure 1-12.

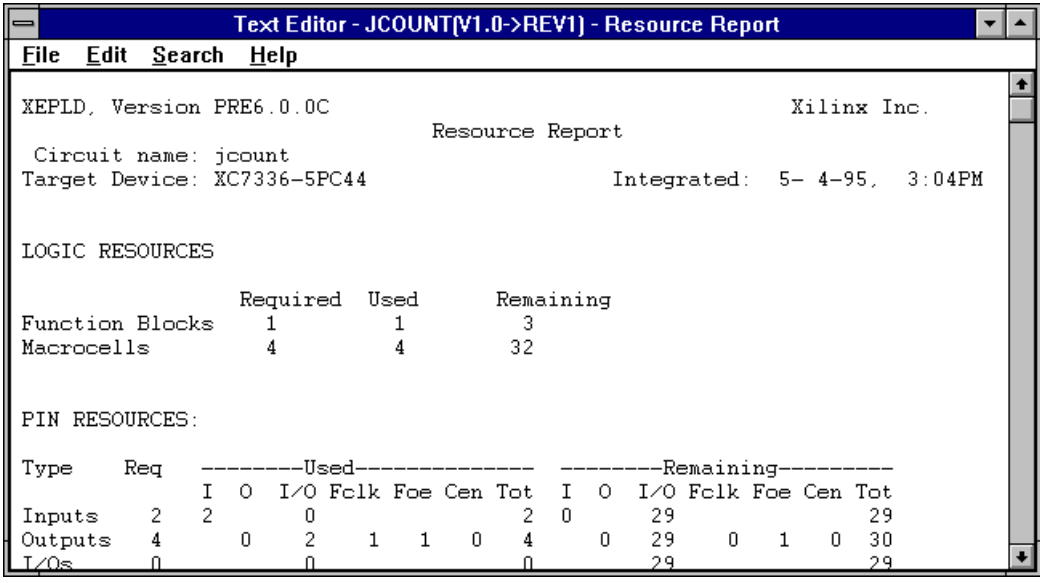


Figure 1-12 Resource Report Window

You can save the report to an ASCII file using the File-Save As command, print the report using File-Print, or close the report window using File-Exit. Select these commands from the report window, not the main Design Manager window.

Use the Timing Analyzer

Double click on the Timing Analyzer icon in the Tools subwindow, as shown in Figure 1-13.

Figure 1-13 Timing Analyzer Icon

The Timing Analyzer main window opens as shown in Figure 1-14, and the jcount design is loaded automatically.

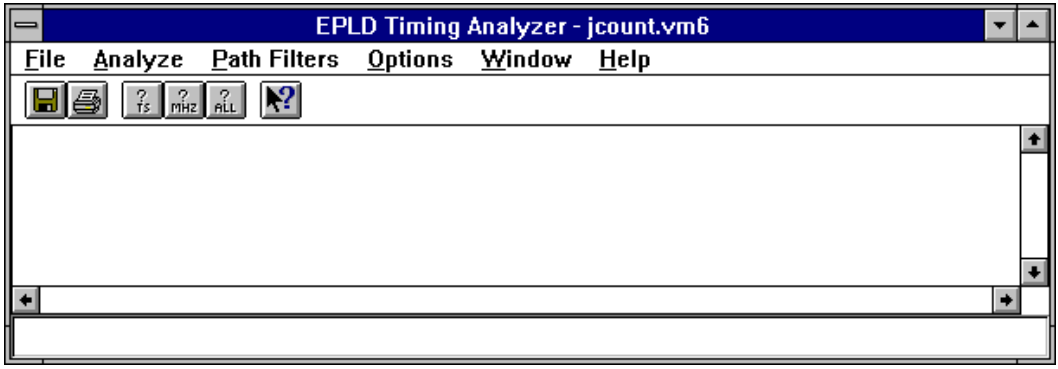


Figure 1-14 Timing Analyzer Main Window

Select the Performance Summary command from the Analyze menu. A report window appears containing the following report:

```
                        Performance Summary Report
                        --------------------------

Design:     jcount
Device:     XC7336-5PC44
Program:    EPLD_TA VER 1.0
SpeedsFile: model.tim Version 1 Release 1
Date:       Mon Apr 17 13:54:42 1995

Timing Analysis Options:

From: All
  To: All

Speed Grade:                                -5

Selected Path Types:

External
--------
Pad to Pad -                                Pad to pad combinatorial propagation
                                            delay. Paths through PRE/CLR register
                                            inputs are not reported.

Clock Pad to Output Pad -                   Clock pad(global or product term
                                            clock) to output pad propagation
                                            delay.

Setup to Clock at Pad -                     Setup time of data at PAD to clock at
                                            PAD. This parameter is only reported
                                            for global clocks and product term
                                            clocks driven directly from input
                                            pads

Internal
--------
Clock to Setup -                            Register to register cycle time.

Pad to Setup -                              Data path delay from PAD to register
                                            data input. Includes register tSU.
Clock to Pad -                              Delay from register clock input to
                                            output pad.

Path Ending at Clock Pin of FFs -           Clock path delay from clock
                                            pad(global or product term) to
                                            register clock input.

Other Options:
Maximum Number of Paths:                    1000
Maximum Number of Paths per TimeSpec:       1000
Report Delays Less Than:                    1000.0ns

Performance Summary:

Clock net 'C' path delays:

Worst case clock pad to output pad delay  :          4.5ns (1 macrocell levels)
(Includes an external input margin of 0.0ns.)
(Includes an external output margin of 0.0ns.)
Clock Pad 'C' to Output Pad 'Q0'                                    (Fast Clock)

Clock to Setup path delay                 :          5.0ns (1 macrocell levels)
```

```
Clock to Q, net '$1N5.Q.FFB.REG.Q' to DFF Setup(D) at '$1N5.Q.FFB.D' (Fast Clock)
Target FF drives output net '$1N5.Q'

Clock to Pad path delay                    :          3.0ns (1 macrocell levels)
(Includes an external output margin of 0.0ns.)
Clock to Q, net '$1N5.Q.FFB.REG.Q' to Pad 'Q0'                      (Fast Clock)

Pad to Setup path delay                    :          8.5ns (0 macrocell levels)
(Includes an external input margin of 0.0ns.)
Pad 'CE' to DFF Setup(D) at '$1N5.Q.FFB.D'                          (Fast Clock)
Target FF drives output net '$1N5.Q'

Pad Ending at Clock Pin path delay         :          1.5ns (0 macrocell levels)
(Includes an external input margin of 0.0ns.)
Clock Pad 'C' to DFF Clock Pin at '$1N16'                           (Fast Clock)

Setup to Clock at the Pad                  :          7.0ns (0 macrocell levels)
Data signal 'CE' to DFF D input Pin at '$1N5.Q.FFB.D'
Clock pad 'C' to DFF Clock Pin at '$1N16'                           (Fast Clock)

                        Minimum Clock Period: 5.0ns
                    Maximum Internal Clock Speed: 200.0Mhz
                          (Limited by Cycle Time)

                Estimated Maximum External Clock Speed: 142.8Mhz
                     (Limited by Setup to Clock at the Pad)

--------------------------------------------------------------------------------
                       Clock Pad to Output Pad Delays(nsec)

\ Clock     C
  \
    \
      \
        \
          \
            \
Output \------

Q0         4.5
Q1         4.5
Q2         4.5
Q3         4.5

--------------------------------------------------------------------------------
                         Setup to Clock at Pad Delays(nsec)

\ Clock              C
  \
    \
      \
        \
          \
            \
              \
                \
                  \
                    \
                      \
                        \
```

```
                \
 Data            \------

$1N5.Q.FFB.REG.Q   3.5
$1N6.Q.FFB.REG.Q   3.5
$1N7.Q.FFB.REG.Q   3.5
$1N8.Q.FFB.REG.Q   3.5
C                  5.0
CE                 7.0

------------------------------------------------------------------------------
                        Register to Register Delays(nsec)
                                    (Clock: C)

\ From                 $     $     $     $
 \                     1     1     1     1
  \                    N     N     N     N
   \                   5     6     7     8
    \                  .     .     .     .
     \                 Q     Q     Q     Q
      \                .     .     .     .
       \               F     F     F     F
        \              F     F     F     F
         \             B     B     B     B
          \            .     .     .     .
           \           R     R     R     R
            \          E     E     E     E
             \         G     G     G     G
              \        .     .     .     .
               \       Q     Q     Q     Q
  To            \------------------------

$1N5.Q.FFB.D       5.0                 5.0
$1N6.Q.FFB.D       5.0   5.0
$1N7.Q.FFB.D             5.0   5.0
$1N8.Q.FFB.D                   5.0   5.0
```

To close the report window, select the File-Exit command in the report window. To close the Timing Analyzer, select the File-Exit command in the main Timing Analyzer window.

Using The Design Manager

This chapter shows you how to manage your design files and how to control your design process flow. Each menu command is described and procedures for common tasks are provided. A typical Design Manager Window is illustrated below.

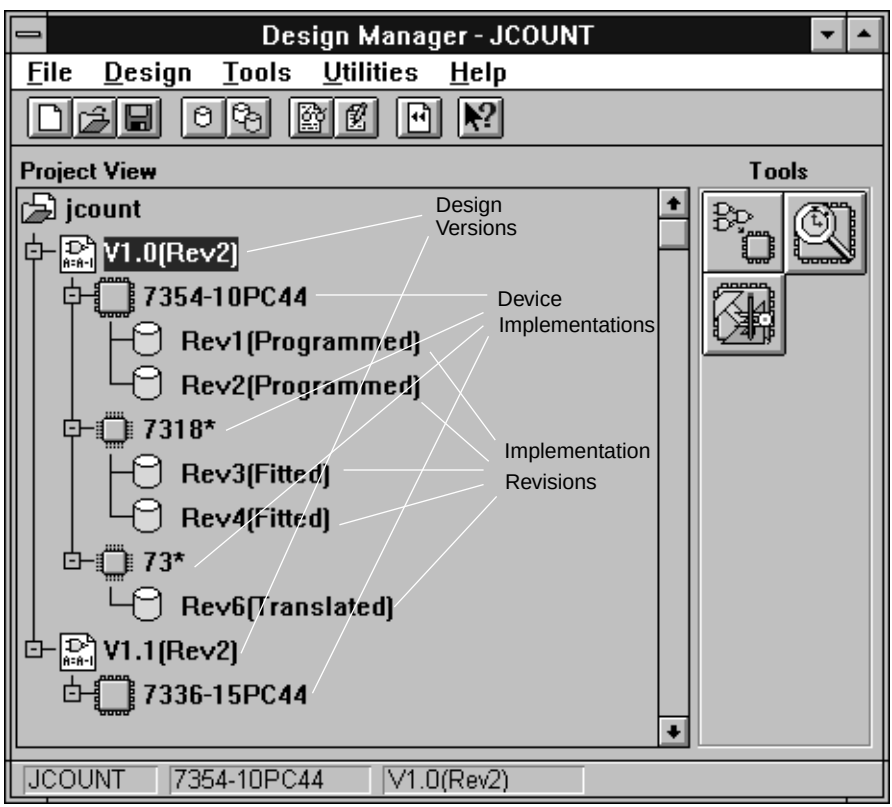


Figure 2-1 The Design Manager Window

Design Manager Fundamentals

There are four main functions of design management that are addressed by the Design Manager:

- Grouping all related Design Files (project). The Design Manager window displays all files related to a single project. The hierarchical file structure shows you the relationships of the files to each other.
- Maintaining Design Versions (logic). Each time you change the logic of your design a new version is created. This allows you to try modified versions of your design and easily keep track of them. Old versions can be deleted as desired.
- Selecting Device Implementations (target devices). Each version of your design can have multiple implementations (target devices) in which fitting can be attempted.
- Maintaining Implementation Revisions (fitting strategies). You can have multiple fitting strategies (revisions) for each device implementation. This allows you to optimize your design for speed and density while retaining your data.

The Design Manager Window, showing your design versions, implementations, and revisions appears when you have chosen an existing project in which to work or have opened a new project. In addition, within the Design Manager User Interface, there are three tools:

- The Flow Engine, used to control the fitting of your design.
- The Partitioner, used to manage multiple device partitioning.
- The Timing Analyzer, used to verify your design timing.

Managing Projects

A project includes all design versions, device implementations, and implementation revisions that are created as you develop your design. The Design Manager Window shown in Figure 2-1 displays the data associated with a single project. You can work with multiple projects (designs) but only one at a time is displayed.

Managing Design Versions

You create your design using third party front-end tools such as schematic editors, behavioral equation editors, and VHDL. These tools output either an .XNF or .PLD file that describes the logic of your design. You create a new .XNF or PLD file each time you change your source design and import that new design to the Design Manager.

Each new .XNF file you import becomes a new design version and is assigned a version number by the Design Manager. You can choose any one of the available versions for processing and for each version you may choose multiple target device implementations and different fitting strategies. The Design Manager also allows you to delete versions that are no longer useful.

Managing Device Implementations

You can fit your design into any number of different target devices; each is called an implementation. You can specify devices explicitly or you can specify a range of devices from which the software can choose the best one. For example, you could specify a device family and package type only, allowing the software to choose the best device type and speed grade for your design. For a description of the automatic device selection process, see Chapter 3.

For each version of your design you can have multiple implementations using different target devices or different fitting strategies.

Managing Implementation Revisions

After you have created a design version by inputting an .XNF file, and you have chosen a target device (or devices), you can try different fitting strategies on that design. This allows you to vary how your design is implemented in order to achieve your design objectives. For example, you can maximize speed and density for specific functions in your design by controlling the fitting process. Each of these fitting strategies is called a design revision.

Each design revision contains the output files and reports that are created based on a specific set of fitting strategies. the Design Manager also allows you to delete revisions that are no longer useful.

Design Management Procedure

The typical procedure for managing a design is as follows:

1. Create your design using a third-party front-end tool such as those available from Viewlogic, OrCAD, Data I/O, and Mentor Graphics. Your design is output as an .XNF, .1, .SCH, or .PLD file, depending on the tool you are using.
2. Open an existing project file; or create a new project file in which to import your logic design.
3. Import your logic design. Your source file is converted into the Xilinx internal data base format (.XFF).
4. Choose a target device (or devices) in which to implement your design. Or, allow the software to automatically choose target devices based on your selection criteria.
5. Define option templates to control your design process flow. Or, use the default template settings.
6. Process (fit) your design and:
   - Create timing simulation files (optional).
   - Create a device programming file (optional)
7. Review your design reports to verify that your design fits within the target device and that your timing requirements are met.
8. If your design requirements are not met you can do the following and process your design again:
   - Change your logic design.
   - Choose a different target device, package, or speed grade.
   - Define a different set of fitting options (option templates).

Using the Design Manager

This section describes the Design Manager menu commands and provides the procedures for using them.

The Initial Design Manager Window

When you first run the Design Manager you will see the initial Design Manager window as shown in Figure 2-2.

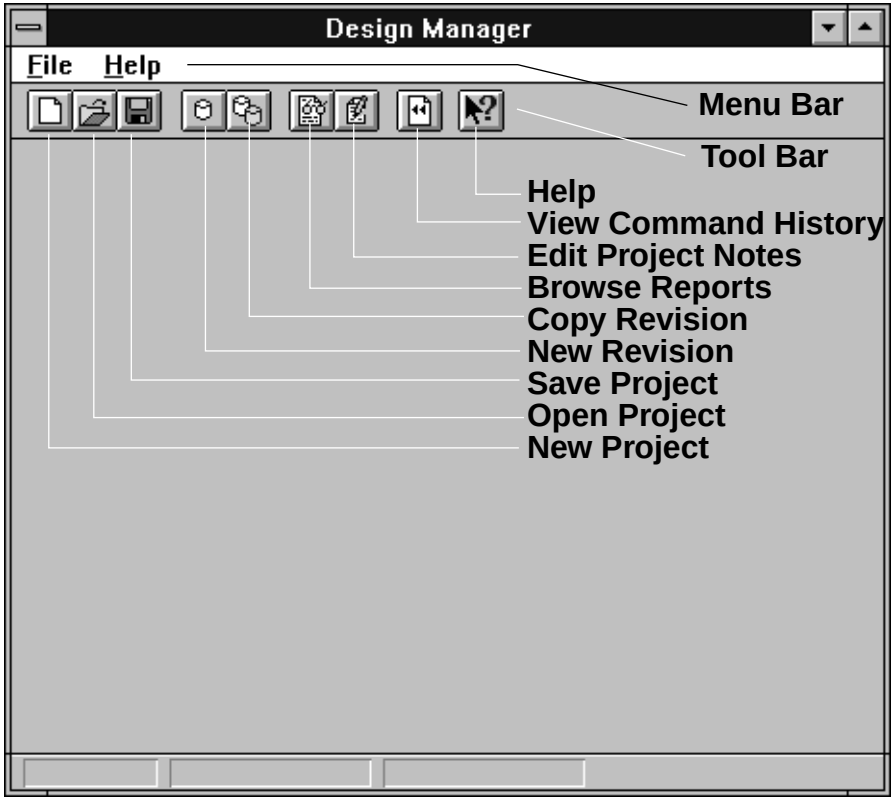


Figure 2-2 The Initial Design Manager Window

The Menu bar contains the primary menu commands arranged according to the types of functions you need to perform. When you click a Menu Bar selection, you see a submenu with more commands.

In the initial Design Manager Window (Figure 2-2) you see only the File and Help commands because these are all you need until you have opened a project. After a project is opened, you will see the full Menu Bar with all available menu commands as shown in Figure 2-1.

The Tool Bar contains icons that represent commonly used functions such as Help, for quick access. Each of these functions is also accessible through the Menu Bar sub commands.

Each menu command is described in the following sections.

File Menu

Use the File menu to manage projects and to exit the Design Manager. When you select File, you see the menu shown in Figure 2-3.

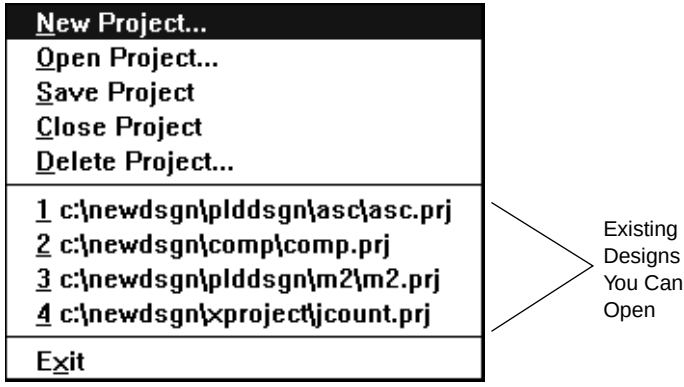


Figure 2-3 The File Menu

File — New Project

Select New Project to create an empty project directory to receive a new design; you will see the New Project dialog window as shown in Figure 2-4.

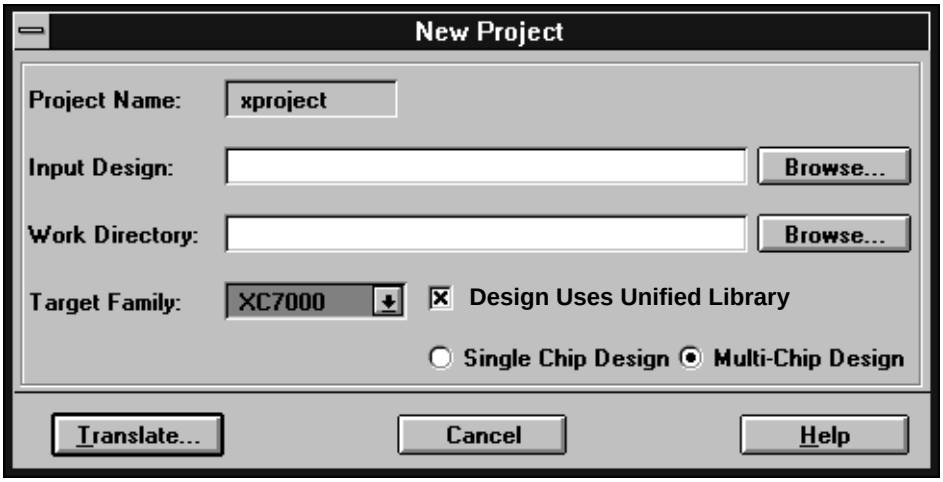


Figure 2-4 The New Project Dialog Window

To open a new project, follow these steps:

1. Enter a project name in the Input Design dialog box. Or, click on Browse and you will see the Open dialog window as shown in Figure 2-5. From here you can select a design file to input.

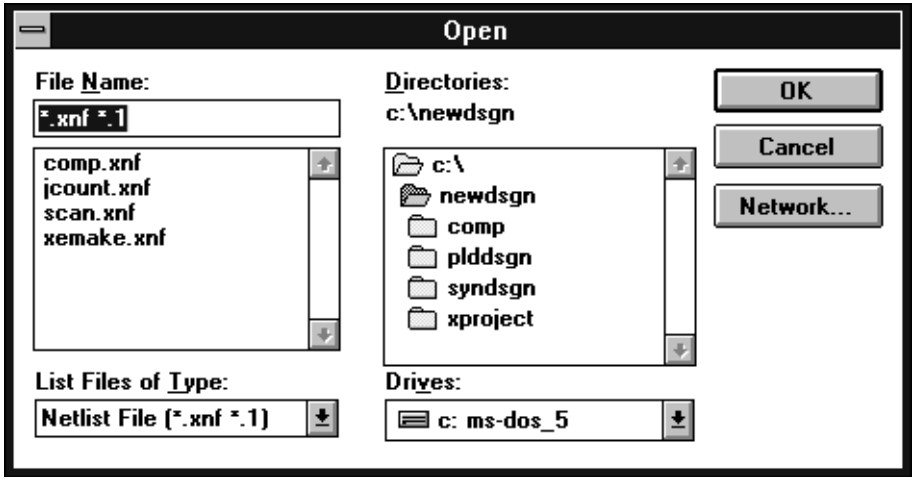


Figure 2-5 The Open Dialog Window

Note that the List of File Types in the Open Dialog Window is already set to find all .XNF files for you; .XNF is the most common input file type. You can change this to .PLD or any other file type for which you want to search.

2. Select the target device family from the Target Family pull-down dialog box. Select XC7000 to target Xilinx EPLDs. The other selections are for Xilinx FPGA devices.

3. Specify if your design is targeted to one device (Single Chip Design) or to multiple devices (Multi-Chip design).

Note: Use Single Chip Design only; multi-chip partitioning is not available in this beta release.

4. Specify your working directory. You can browse through the available directories by clicking on Browse, which brings up the dialog window as shown in Figure 2-6. From here, select the working directory and click OK.

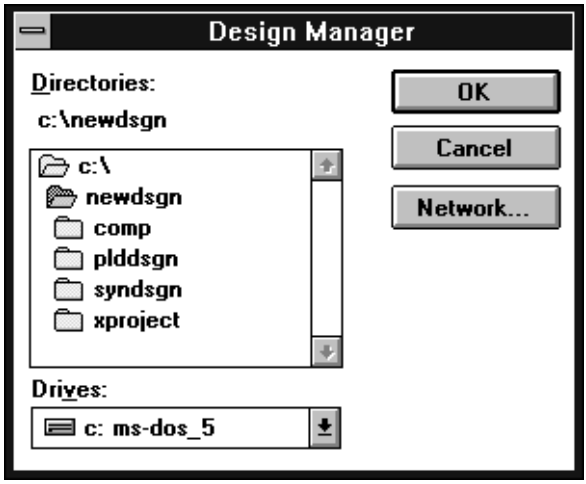


Figure 2-6 The Work Directory Browse Dialog Window

5. After you have specified the project name, device family, and working directory, you can input (translate) your logic design by clicking on the Translate button. You will see the Translate Options dialog window as shown in Figure 2-7.

6. From the Translate Options dialog window you can choose a design version number or accept the default version number. You can also specify whether the software is to read the device information contained in your source file or use the part you have selected. To select a Part click Select Part and you see the Part Selector window from which you can specify a target device as shown in Figure 2-8. Click OK when done.

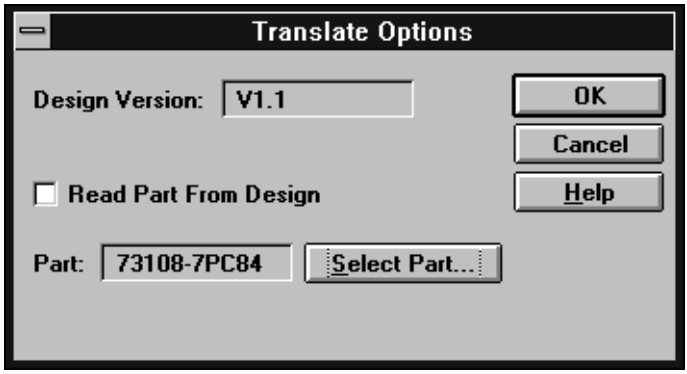


Figure 2-7 Import Design Translation Flow Window

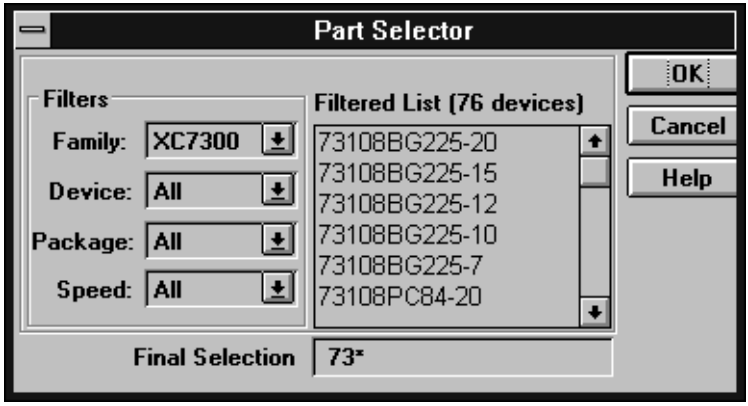


Figure 2-8 Part Selector

After you have selected the Translate Options, click OK and your design file is input and converted to the Xilinx .XFF internal database format, ready for processing.

When the translation is complete, you see the Translate Confirmation Dialog Window as shown in Figure 2-9.

Figure 2-9 Translate Confirmation Dialog Window

7. From here, if you select Review log, you will see the Translation log report as shown in Figure 2-10.

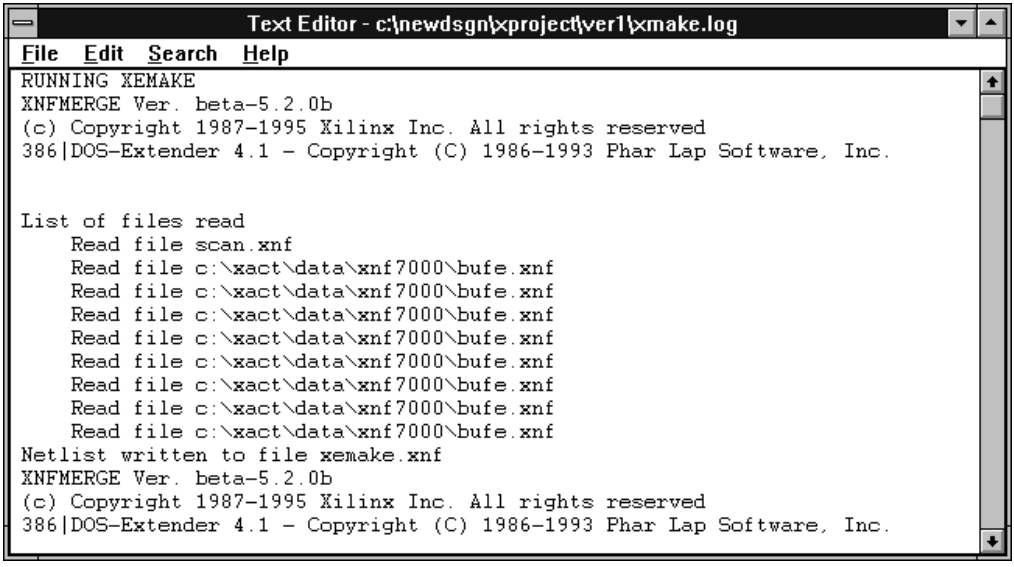


Figure 2-10 Translation Log Report

This log report shows you the commands that were executed in the translation of your design as well as any warnings or error messages.

File — Open Project

Select Open Project to open an existing project. You will see the following dialog box, which displays all available projects:

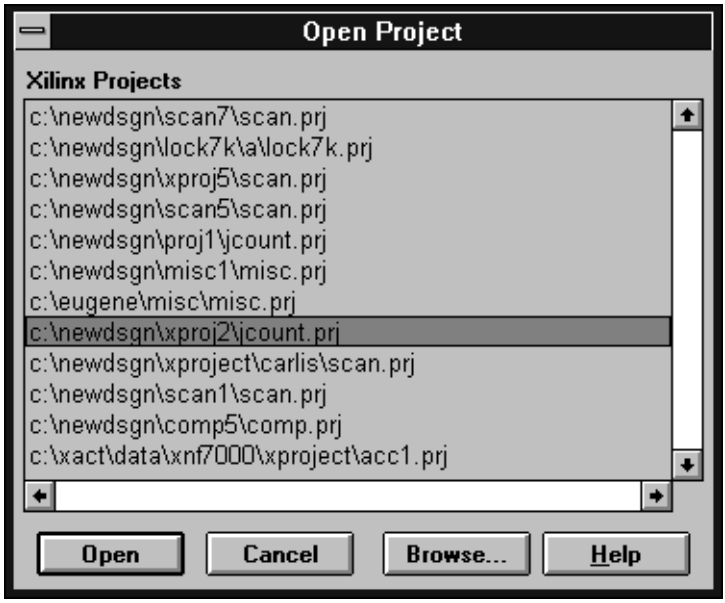


Figure 2-11 The Open Project Dialog Window

If you double-click a project, or highlight a project and click Open, the project file opens and you will be returned to the Design Manager window with the project hierarchy displayed.

File — Save Project

Select Save Project to save all files associated with your current project.

File — Close Project

Use Close Project to close and save all files associated with the current project.

File — Delete Project

Select Delete Project to remove a project from your list of active projects and you will see the dialog window shown in Figure 2-12.

Figure 2-12 The Xilinx Projects Dialog Window

Double-click on the project you want to delete, or highlight the project and click the Delete button.

If you do not wish to delete a project at this time, click Cancel and you are returned to the Design Manager window.

File — File name

The File menu automatically lists all available project names. You can open any listed project by simply clicking the name.

File — Exit

Select Exit to close your design and exit the Design Manager. Your project is saved automatically

Design Menu

Select Design from the menu bar and you see the menu as shown in Figure 2-13.

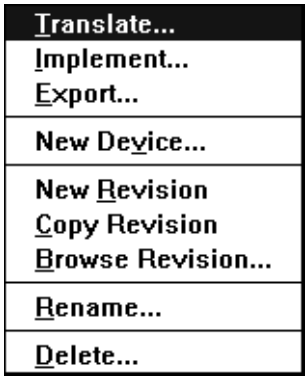


Figure 2-13 The Design Menu

Design — Translate

Select Translate to input your logic design and convert it to the Xilinx internal database format (.XFF) ready for processing. You will see the Translate Options dialog window as shown in Figure 2-14.

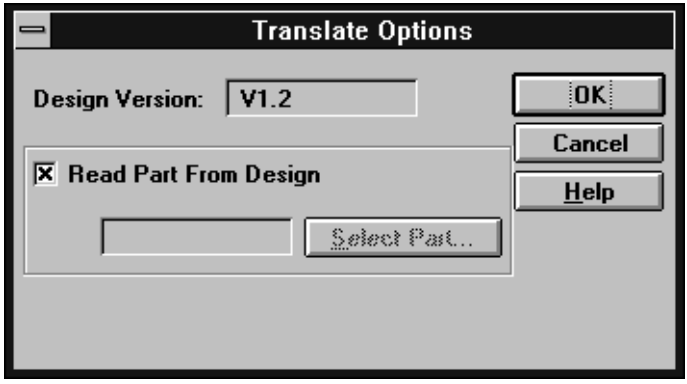


Figure 2-14 Import Design Translation Flow Window

From the Translate Options dialog window you can choose a design version number or accept the default version number. You can also specify if the software is to read the device information contained in your source file.

After you have selected the Translate Options, click OK and your design file is input and converted to a Xilinx .XFF internal database format.

The supported input formats are:

- .XNF — Xilinx Netlist Format, supported by most tools.
- .SCH — OrCad schematic output format.
- .1 — ViewLogic schematic output format.
- .PLD — PAL and PLD files.

When the translation is complete you see the Translate Confirmation Dialog Window as shown in Figure 2-15.

Figure 2-15 Translate Confirmation Dialog Window

From here, if you select Review log, you will see the Translation log report as shown previously in Figure 2-10. Select OK and you are returned to the previous menu.

Design — Implement

Select Implement and you see the XC7300 Design Implementation Dialog Window as shown in Figure 2-16. Use this dialog window to control the fitting process by specifying control files, program option templates, and optional target files for your design.

To implement your design follow these guidelines:

- Guide Design: If you want to use the pinouts from a previous implementation, click on the Guide Design scroll arrow to see a list of all available Guide Files (.GYD). A guide file is created each time you fit your design and you can reuse this pinout information to maintain design consistency. For more information on using Guide Files see Chapter 3.

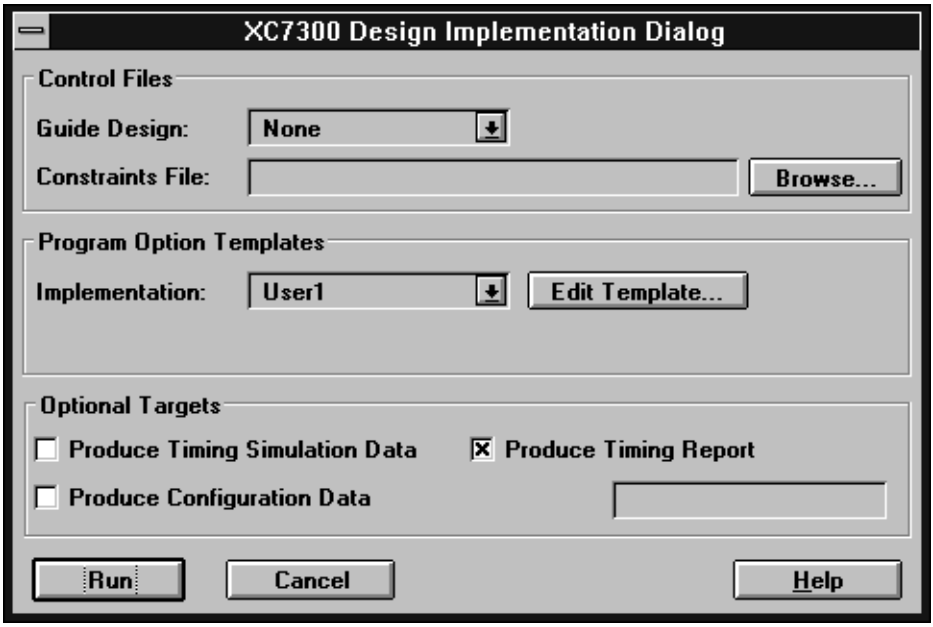


Figure 2-16 The XC7300 Design Implementation Dialog Window

- Constraints File: If you want to use timing-driven optimization, enter a constraints file name or click on the Browse button to see a list of all available timing constraints files. For information on how to create a constraints file, see Chapter 4.
- Implementation: If you want to use a Program Option Template, (a set of fitter options that control your design processing), click on the Implementation scroll arrow to see a list of all available Program Option Templates. Click Edit Template if you want to change the template parameters. See the "Template Manager" section (Figure 2-33) for more information on how to create and edit Option Templates.

- Produce Timing Report: If you want to produce a timing summary report click this check box; an X appears to indicate that the option is selected. For a more detailed timing analysis, you can use the static timing analyzer after the design is implemented. See Chapter 6 for more information on Timing Reports.
- Produce Timing Simulation Data: If you want to produce timing simulation data, click this check box. Check this option if you intend to use a third party timing simulator to verify your design.
- Produce Configuration Data: If you want to output configuration data (a device programming file), click this check box. You will also need to enter a unique text string in the Signature dialog box to identify this file.
- Run: When all options are entered, click the Run button to implement your design. You will see the simplified Flow engine Dialog window along with the Report browser, as shown in Figure 2-17. As the simplified Flow Engine processes your design, you will see the reports highlighted as they are created.

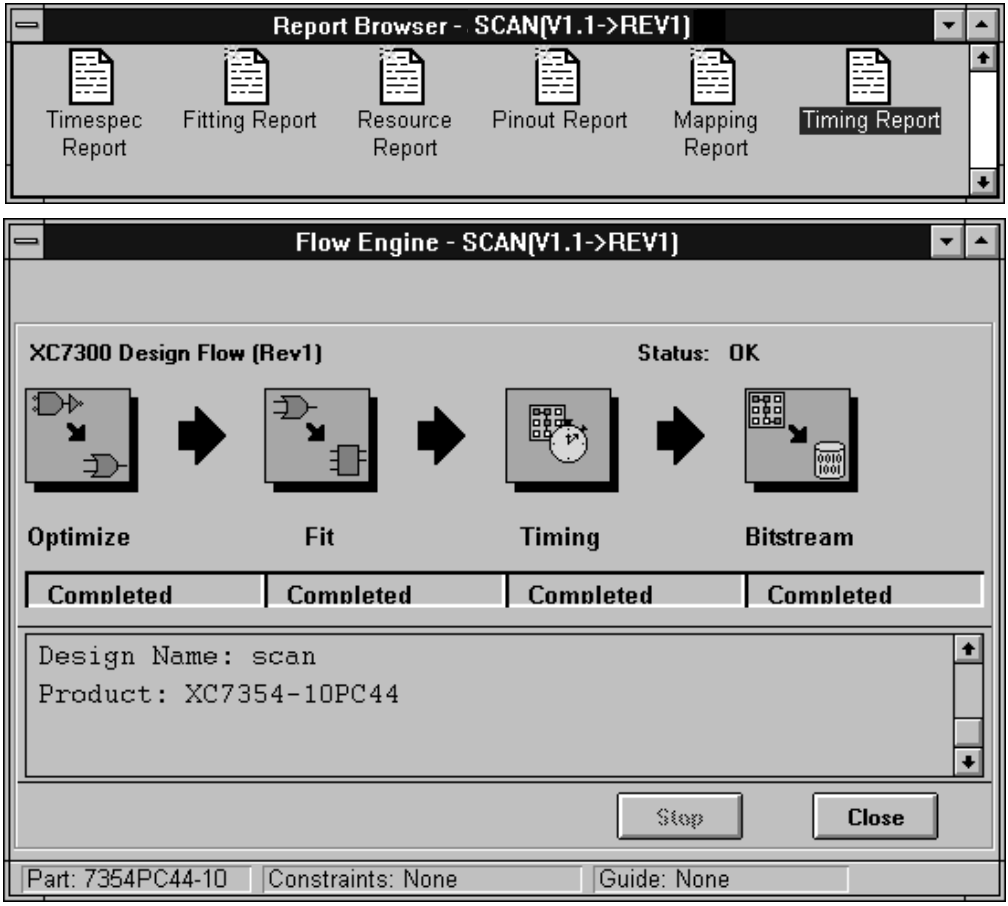


Figure 2-17 The Simplified Flow Engine with Report Browser

Note: When you implement your design from the Design menu, you see the simplified Flow Engine shown above. The simplified Flow Engine has a limited set of options. If you want the full set of options, to control the operation of the Flow Engine, select the Flow Engine from the tools menu or from its icon in the Design Manager Toolbox.

Design — Export

Select Export to control the types of output data produced by the software.

Figure 2-18 The Export Data Dialog Window

- Physical Design Data: Select this option to export a guide file (.GYD) containing your pinout information. This file is used to recreate you existing pinout after making design changes.
- Timing Simulation Data: Select this option to export a simulation file (.XNF) for third-party simulators such as those available from Viewlogic (ViewSim) and OrCAD.
- Configuration Data: Select this option to export a device programming file (.PRG).
- Files To Export: Here you see all files that match the data type you have selected for export.
- Export To: Enter the full path of the file to which you are exporting the data. Select Browse to see a list of all available files as shown in Figure 2-19.

Click OK when you are ready to export the data.

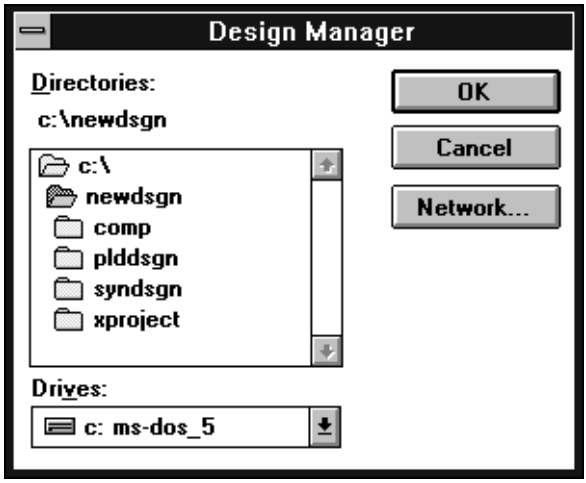


Figure 2-19 The Export Data Browse Dialog Window

Design — New Device

Select New Device to add a new target device under the selected version hierarchy. This allows you to try different devices (and fitting strategies) for a specific version of your design. You can either specify a single device or a range of possible devices from which the software can choose the best one. The device selection procedure is described as follows:

1. Select the version under which the new target device is to be added.
2. Select New Device from the Design menu. You will see the Part Selector dialog window as shown in Figure 2-20.

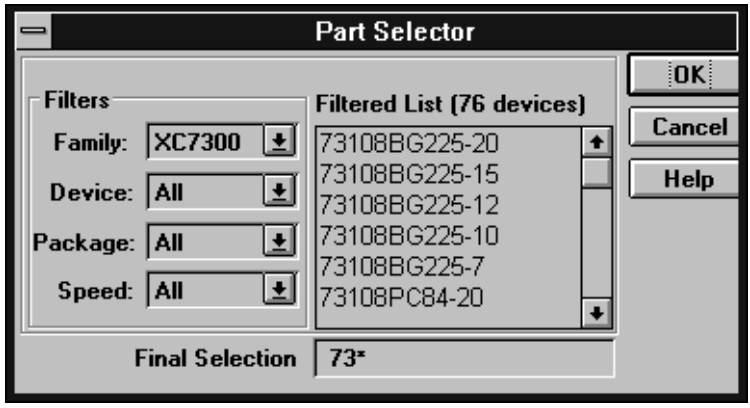


Figure 2-20 Part Selector Window

3. If you know the specific device you want to use, select the part type, package type, and speed grade by scrolling through the available options as shown in Figure 2-20. For EPLD designs select the XC7000 family.
4. If you want to allow the software to choose the best device for your design, specify only those parameters that you require. For example, you can specify the PC84 package and leave the device type and speed grade unspecified. The unspecified parameters are shown as an asterisk (*) in the Final Selection window and on the Design Manager window.

   Click OK and the specified device (or set of devices) is added to the design hierarchy as shown in Figure 2-1.

Design — New Revision

Select New Revision to create a new (translated) revision of your design in .XFF format. This does not save any of your previous implementation data.

Design — Copy Revision

Select a revision in the Design Manager window and Select Copy Revision to make a copy of that revision. You will see the new revision (a copy) displayed in the Design Manager window using the

next Rev number. For example, if Rev 1, Rev 2, and Rev3 already exist and you make a copy (of any one), Rev 4 will be created. Usually you copy a revision in order to modify the fitting strategy it contains. A revision copy contains the previous implementation data.

Design — Browse Revision

Select Browse Revision to edit, copy, move, rename, or delete the files contained within a revision; you see the Revision Browser dialog window as shown in Figure 2-21.

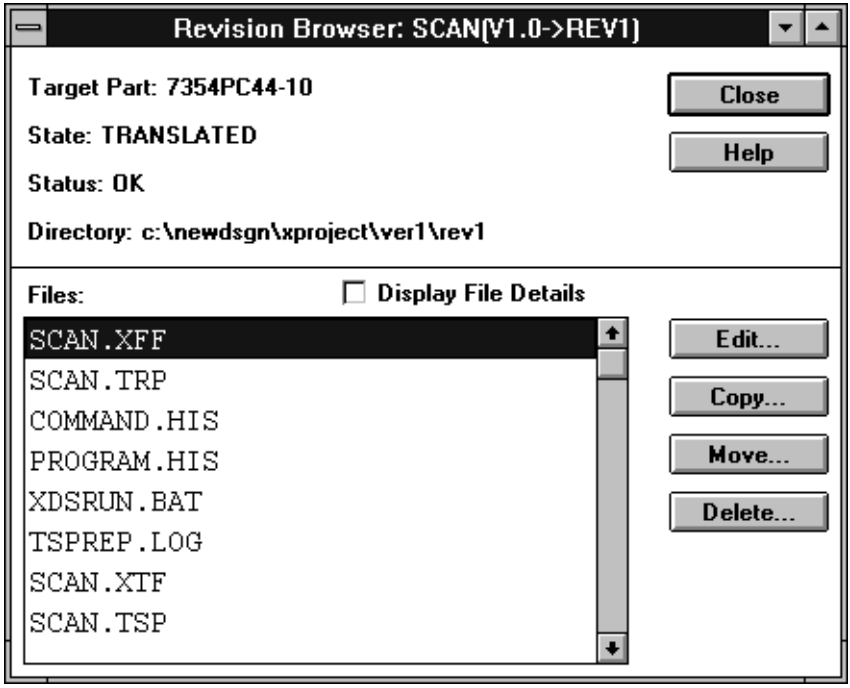


Figure 2-21 The Revision Browser Dialog Window

This dialog window shows you the following information:

- Target Part: The device to which the selected revision is targeted.
- State: The last completed process step for the selected revision, which is either: TRANSLATED, OPTIMIZED, FIT, TIMING, or PROGRAMMED.

- Status: The status of the design processing up to the current state:
  - OK — The software generated no errors or warnings.
  - Warning — The software encountered a situation that may require your attention.
  - Error — The software encountered a processing error.
- Directory: The directory in which the selected revision resides.
- Files: This scroll box shows you all the files contained within the selected revision. If you want to display the file details (creation date, creation time, and so on), select the Display File Details check box.
- Edit: If you want to modify a file, select the file and click Edit; you will see the dialog window shown in Figure 2-22:

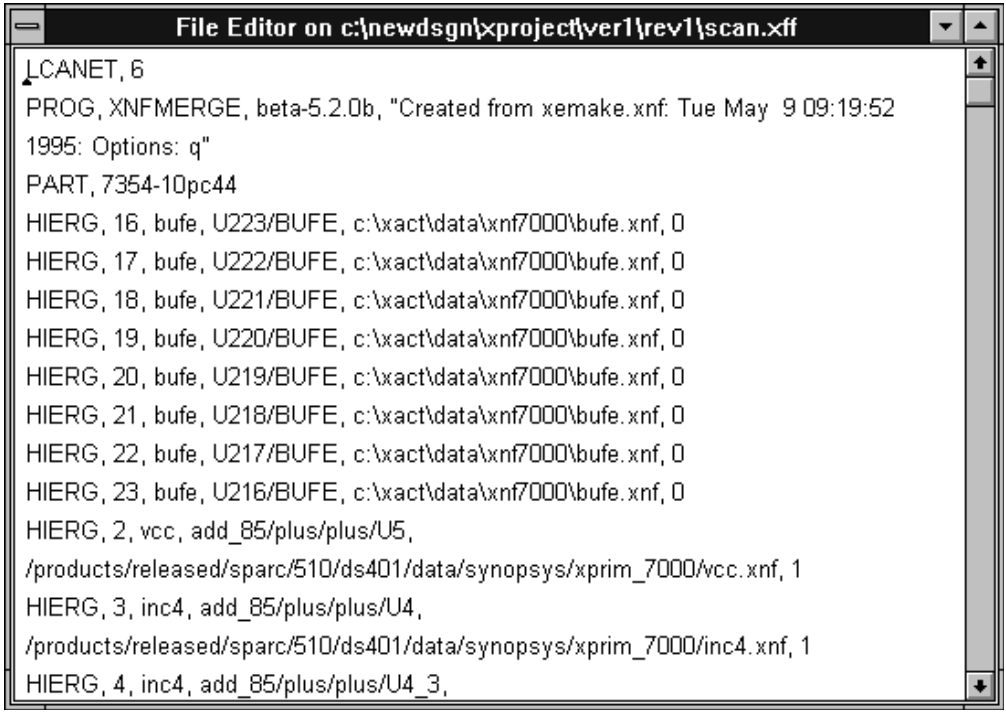


Figure 2-22 The Browse Revision Edit Dialog Window

Use this file editor to make changes to the selected file.

- Copy: If you want to copy a file, click Copy and you see the following Save As Dialog window as shown in Figure 2-23.

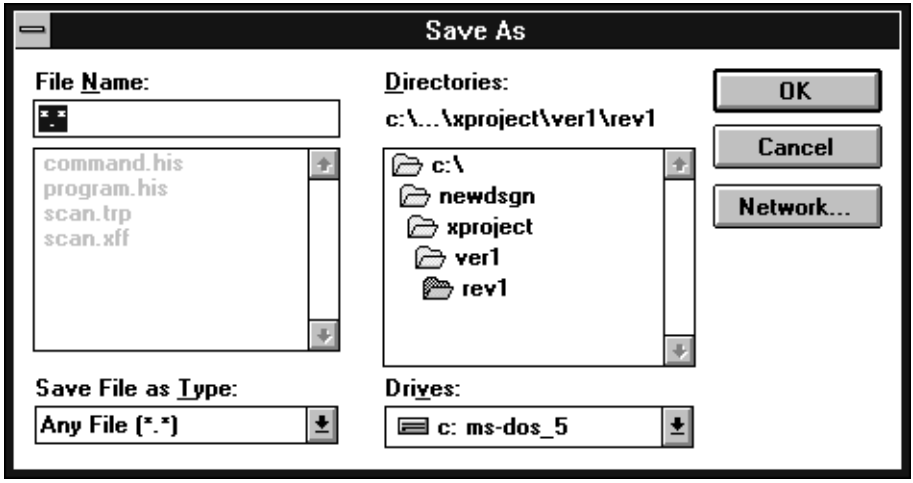


Figure 2-23 The Save As Dialog Window

- Move: If you want to move a file, highlight the file name in the Revision Browser window and click Move; you will see the same Save As dialog window as shown in Figure 2-23.
- Delete: If you want to delete a file, select the file name in the Revision Browser window and click Delete; you will see the delete confirmation window shown in Figure 2-24.

Figure 2-24 The Delete Confirmation Window

Click Yes and the file is deleted. Click No and you are returned to the previous menu.

Design — Rename

Highlight the revision in the Design Manager window and then select Name Revision to assign a name to the selected revision; you will see the Name Revision dialog window as shown in Figure 2-25.

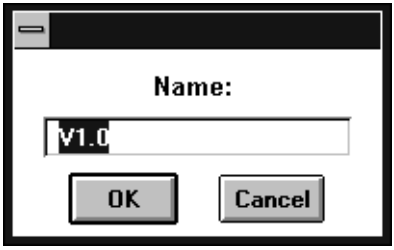


Figure 2-25 Name Revision Dialog Window

Enter the revision name in the dialog box and click OK to apply that name to the selected revision. The name you enter will replace the Rev number. For example, if you select “Rev 1 (Translated)” to rename and enter the name “MyRev8”, you will see the name change to “MyRev8 (Translated)”. Click Cancel to return to the Design Manager window.

Design — Delete Revision

Highlight a revision in the Design Manager window and then select Delete Revision and you will see the dialog window as shown in Figure 2-26.

Figure 2-26 Delete Revision Dialog Window

Click Yes and the highlighted revision is deleted. Click No and you are returned to the Implementation menu.

Tools Menu

Use the Tools menu to select the Flow Engine, the Timing Analyzer, or the Multi-Chip Partitioner, and you see the menu as shown in Figure 2-27.

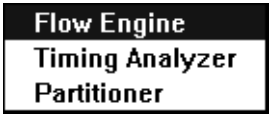


Figure 2-27 The Tools Menu

Tools — Flow Engine

The Flow Engine is used to control the processing of your design including optimization, fitting, timing simulation data generation, and device programming file generation. See “Using the Flow Engine” later in this chapter, for more information.

Tools — Timing Analyzer

The Timing Analyzer is used to verify the critical path timing in your design, as shown in Figure 2-28. See Chapter 5 for more information on how to use the Timing Analyzer.

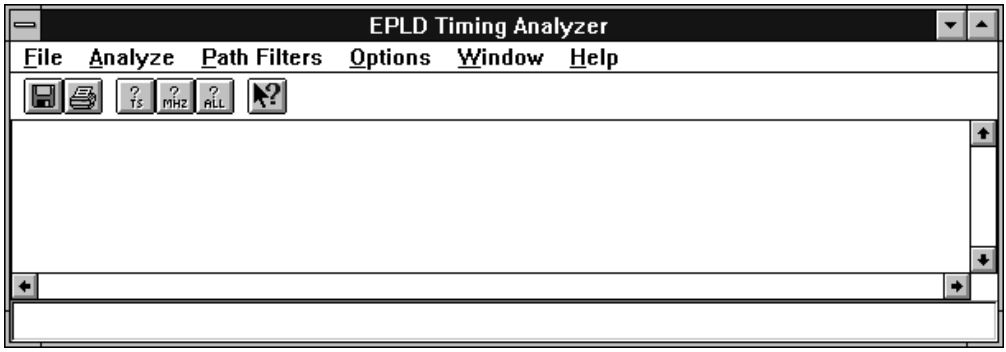


Figure 2-28 Timing Analyzer

Tools — Partitioner

The partitioner is used to fit your design into multiple target devices. This is a new feature. See the release notes for the latest information.

Utilities Menu

Use the Utilities Menu to access the various Design Manager utilities as shown in Figure 2-29.

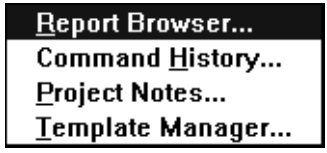


Figure 2-29 The Utilities Menu

Utilities — Report Browser

Select Report Browser to view and edit your design reports; you will see the Report Browser window as shown in Figure 2-30.

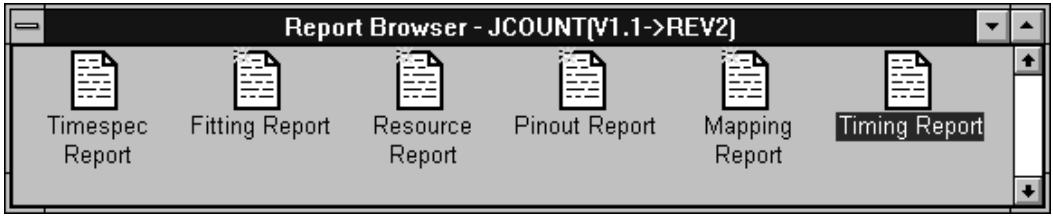


Figure 2-30 The Report Browser Window

Double click on a report icon to view the report. For more information on the Report Browser, see Chapter 6.

Utilities — Command History

Select Command History to view the commands that were previously executed when you implemented your design; you will see the Command History window as shown in Figure 2-31.

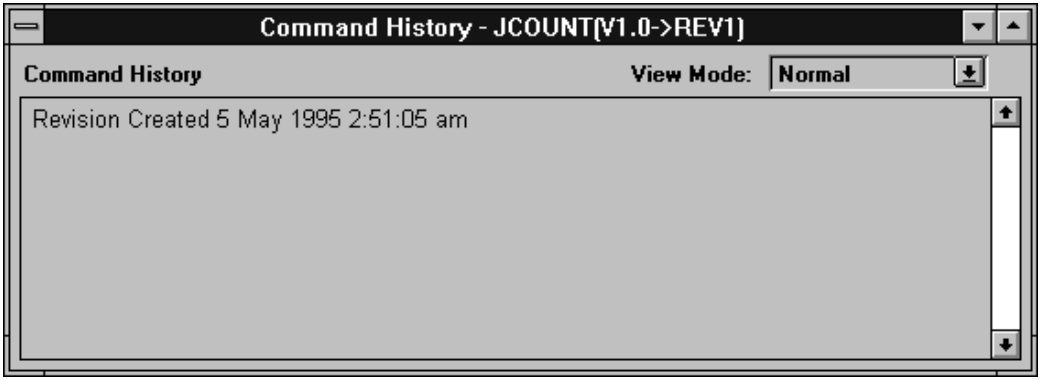


Figure 2-31 The Command History Window

You can view the commands in either Normal mode, which displays only the command names; or Primitive mode, which displays the command name and the associated control parameters.

Utilities — Project Notes

Select Project Notes to document your project; you will see the Project Notes dialog window as shown in Figure 2-32.

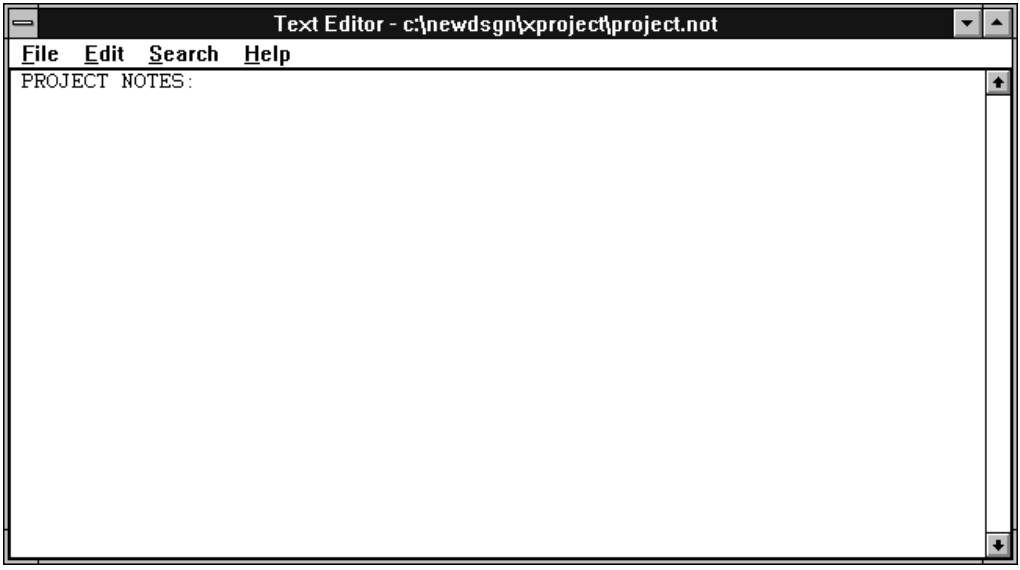


Figure 2-32 The Project Notes Window

You can enter any text into this basic text editor. Project notes are usually used to document the revisions, implementations, and versions of your design.

Utilities — Template Manager

Select Template Manager to create, edit, and delete implementation option templates, which control the processing of your design; you will see the Template Manager dialog window as shown in Figure 2-33.

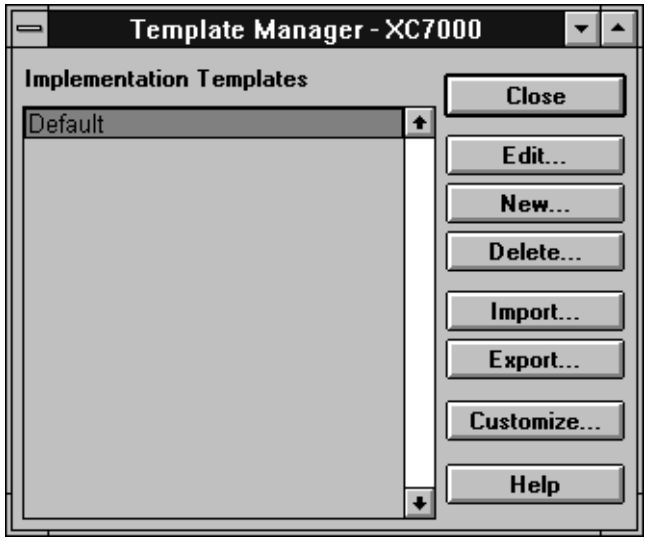


Figure 2-33 The Template Manager Window

Utilities — Template Manager — Close

Click Close to close the Template Manager and return to the Flow Engine dialog window.

Utilities — Template Manager — Edit

After you have highlighted one of the option templates, click Edit to open the following templates for controlling Fitting options, Optimization options, and Resource reservation options.

When you click Edit you will see the Fitting Template, as shown in Figure 2-34.

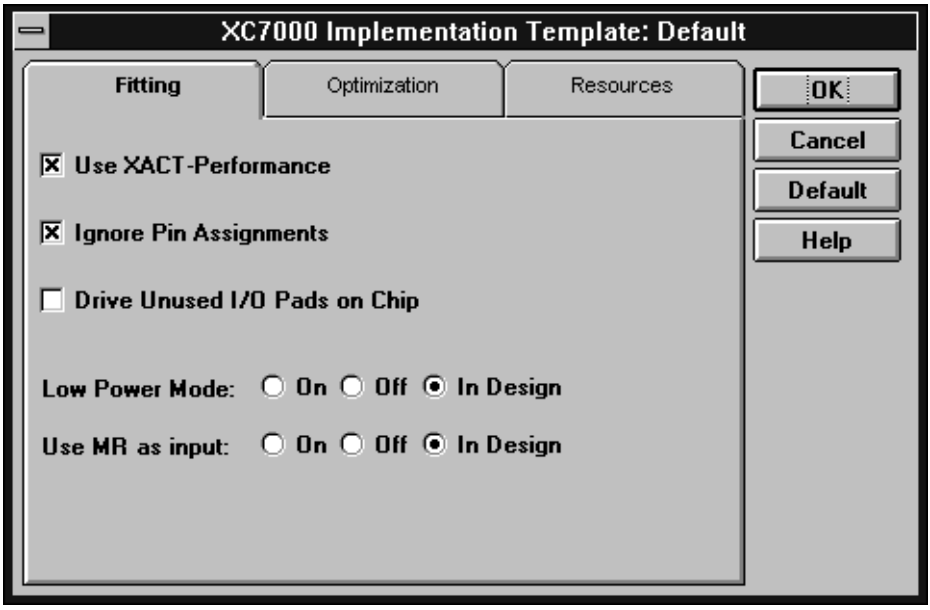


Figure 2-34 The Fitting Options Template

The Fitting Template allows you to control the fitting options, which are set as follows:

- Use XACT-Performance: This indicates that you want the software to use your T-Spec information and perform timing-driven optimization in the fitting of your design. For this option to be useful, you must have previously created a timing constraints file or you must have placed T-Spec information on your schematic. See Chapter 4 for information on using T-Specs.
- Ignore Pin Assignments: This indicates that you do not want to use any pinout information that may be in the design file. This allows the fitter to place pins anywhere.
- Drive Unused I/O Pads On Chip: This indicates that you want all unused I/O pads to be actively driven by on-chip circuitry. Because all unused I/O signals must be actively driven, this option alleviates the need for external drivers or pull-up resistors. See the device data sheet for more information about I/O pads. Low Power Mode: Controls device power consumption:

- On — Set Low Power mode for the entire design.
- Off — Set high performance (higher power mode) for the entire design.
- In Design — Allow the power mode to be controlled by information in your design file.

- Use MR As Input: Controls the use of the MR Input pin on the XC7318, XC7336, and XC7354:
  - On — Use the MR pin as an input pin.
  - Off — Use the MR pin as a Master Reset input pin.
  - In Design — Allow the pin use to be determined by information in your design file.

Click OK to accept the template, click Cancel to return to the previous menu, click Default to set the default options, or click Help to open the on-line help system.

Click Optimization and you will see the Optimization Template as shown in Figure 2-35.

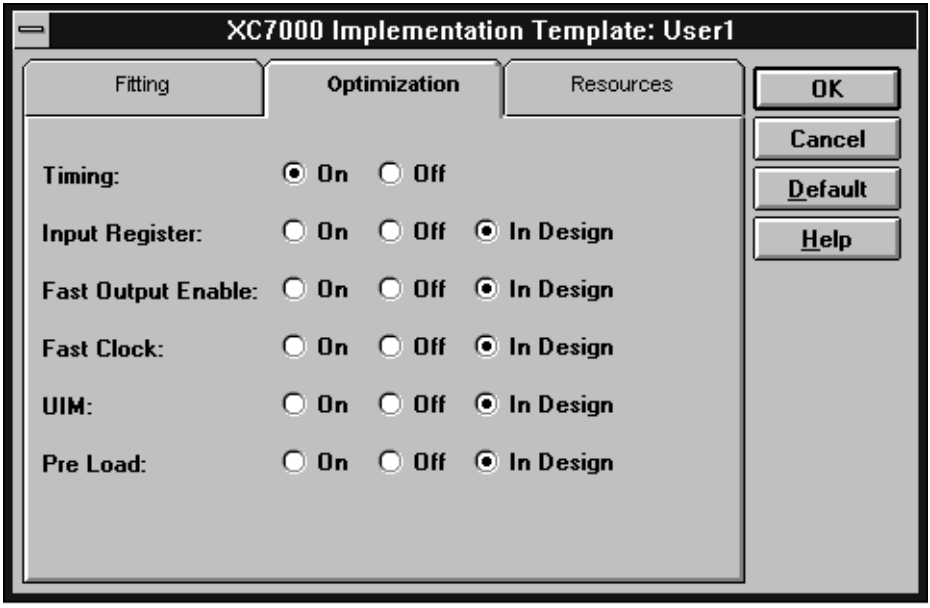


Figure 2-35 The Optimization Options Template

From the Optimization Template you can control the optimization options which are set as follows:

- On — The option is activated for the whole design and corresponding schematic attributes, behavioral commands, or VHDL commands are ignored.
- Off — The option is turned off for the whole design and corresponding schematic attributes, behavioral commands, or VHDL commands are ignored.
- In Design — The option is controlled exclusively by schematic attributes, behavioral commands, or VHDL commands. If the option is not specified in the design, it defaults to On.

The options are:

- Timing — Determines if the fitter will try to optimize your design to achieve the best possible timing for your critical paths. If this option is off, the fitter will try to achieve the best possible density, without regard for timing.

  This timing optimization is performed independently from the timing-driven optimization of XACT Performance, which was described in a previous section.
- Input Register — Determines if unused input registers can be used to implement logic.
- Fast Output Enable — Determines if output enable signals can use the dedicated high speed FOE nets.
- Fast Clock — Determines if clocks can use the dedicated FastClock nets.
- UIM — Determines if the UIM can be used to implement basic boolean logic functions.
- Pre Load — Determines if the fitter can optimize registers that have no assigned preload values. This optimization allows the software to choose preload values that simplify fitting.

Click OK to accept the template, click Cancel to return to the previous menu, click Default to set the default options, or click Help to open the on-line help system.

Note: See the XC7000 data sheets for more information on how these options affect your design.

Click Resources and you see the Resources Template as shown in Figure 2-36.

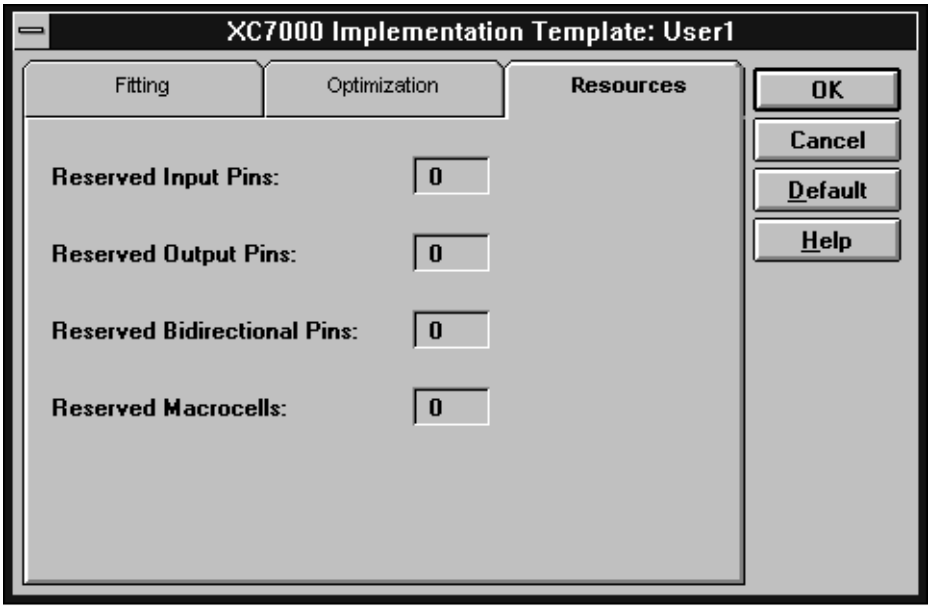


Figure 2-36 The Resource Reservation Template

From the Resources Template you can control the amount of device resources reserved for future use as follows:

- Reserved Input Pins — Specifies the number of input pins to leave unused.
- Reserved Output Pins — Specifies the number of Output pins to leave unused.
- Reserved Bidirectional Pins — Specifies the number of I/O pins to leave unused.
- Reserved Macrocells — Specifies the number of High Density Function Block macro cells to reserve.

Click OK to accept the Resources template, click Cancel to return to the previous menu, click Default to set the default options, or click Help to open the on-line help system.

Utilities — Template Manager — New

Click New to create a new template and you will see the dialog window as shown in Figure 2-37.

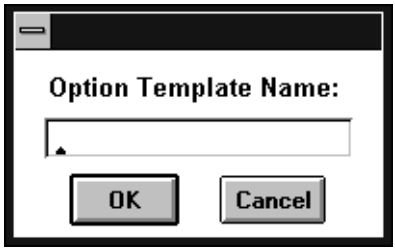


Figure 2-37 The New Template Name Dialog Window

Enter the new template name and click OK; you will see the new name added to the list of option templates. This new template will have default settings that you can change as required.

Utilities — Template Manager — Delete

Click Delete to remove a selected option template from the list. You will see the Delete Confirmation dialog window as shown in Figure 2-38.

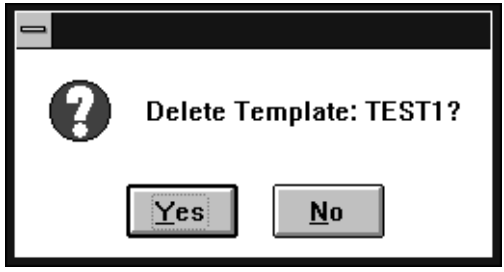


Figure 2-38 Delete Confirmation Dialog Window

Utilities — Template Manager — Import

Use Import to transfer a template file (created by Export as shown below) to a new design.

Utilities — Template Manager — Export

Use export to create a template file for outputting template information to another design.

Utilities — Template Manager — Help

Click Help to bring up the on-line help system.

Help Menu

Use the Help menu to get quick answers to questions about using the Design Manager. You will see the Help menu as shown in Figure 2-39.

Figure 2-39 The Help Menu

Help — Contents

Select contents to view the available help topics and you see the Help window as shown in Figure 2-40. Click on any topic to see the help it contains.

Figure 2-40 Help Contents

Help — Search

Select Search to search for help on any keyword you choose; you will see the search dialog window as shown in Figure 2-41.

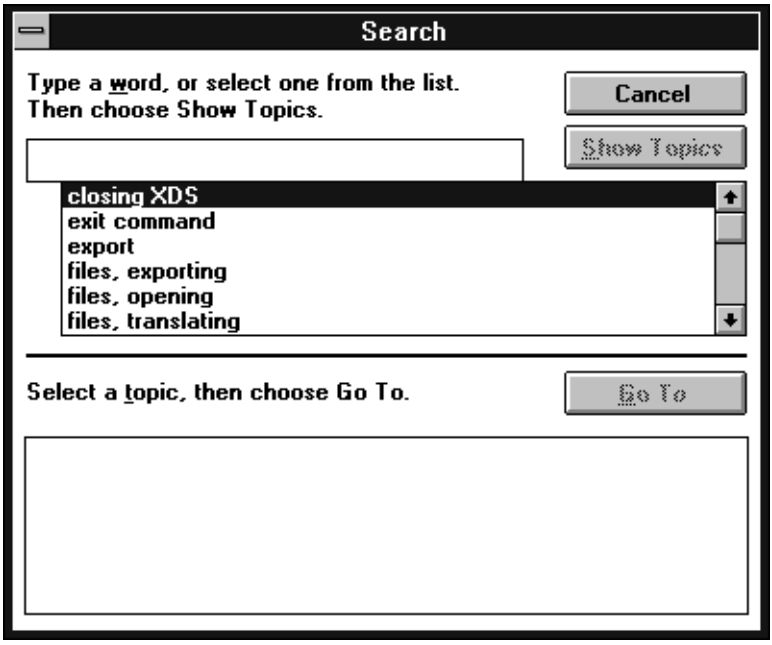


Figure 2-41 Help Search

Help — Tutorial

Select Tutorial to see a step-by-step procedure for using Design Manager.

Help — About Design Manager

Select About Design Manager and you see the Xilinx address, phone number, and software copyright notice.

Using the Flow Engine

The Flow Engine allows you to easily manage the process flow of your design. Select the Flow Engine from the Design Manager Tools menu or by double clicking the Flow Engine icon:

You will see the Flow Engine Window as shown in Figure 2-42.

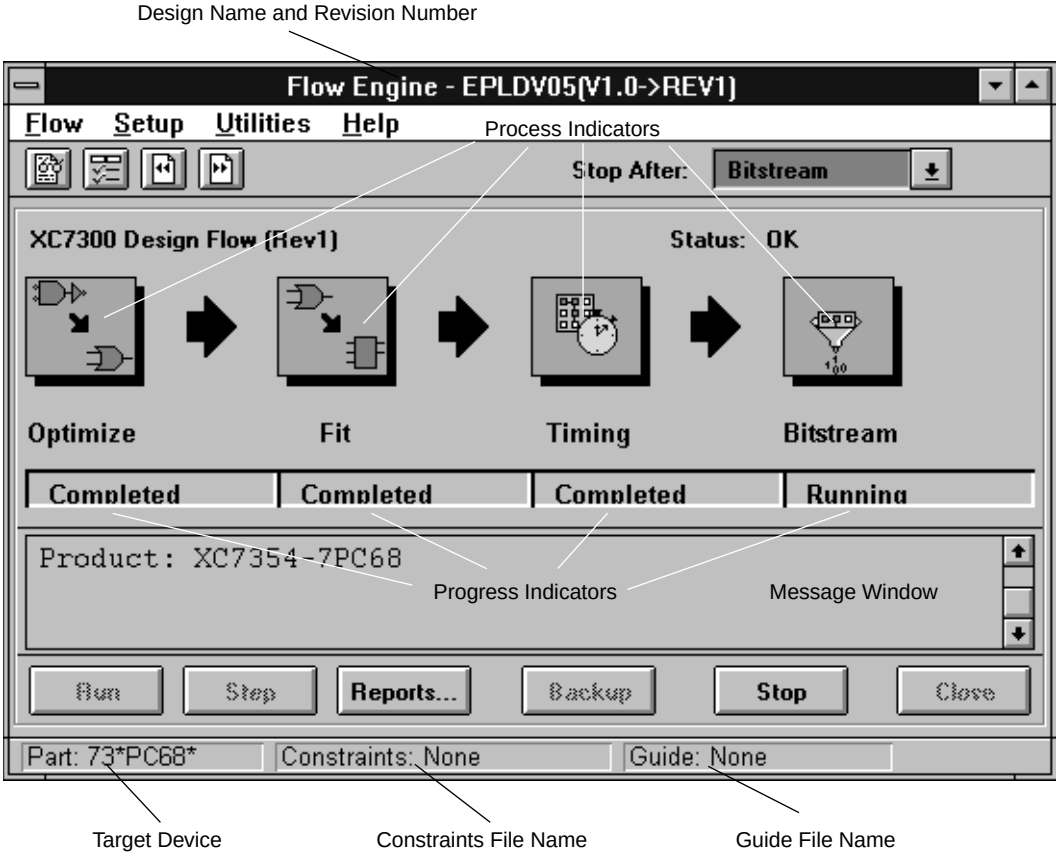


Figure 2-42 The Flow Engine Window

The Process Indicators show you the various software functions that are available for processing your design. You control how far your design is processed by using the Stop After pull down dialog box. For example if you want to optimize and fit your design, but not create a timing simulation file or device programming file, select Stop After Fit. You will see a stop sign indicator verifying where the process will stop.

The Progress Bars show you the status of each processing step and the arrows between each step turn black after the previous step is completed.

Setting Up The Process Flow

The Flow Engine allows you to interact with the process flow in several ways. For example, if you are new to Xilinx software or just want to run a quick "sanity check" on your design, you can simply click run and your design is processed using default fitting parameters. However, if you are an experienced user and want to fine tune your design to get the best possible speed or density, you can control each function in the process. These controls are located under the Setup menu.

Using a Constraints File (to Define T-Specs)

You can create timing specifications, which control the implementation of your design, as described in Chapter 4. These T-Specs may be described in a constraints file (*design_name*.cst). If you want to use a constraints file to control the implementation of your design, you can specify this file in the Flow Engine. The implementation software will then try to fit your design to meet the specified timing requirements. If the specified timing cannot be satisfied, the process flow is terminated.

Using a Guide File (to Re-Use Pinouts)

Each time you fit your design, a guide file is created (*design_name*.gyd) which contains your pinout information. You can re-use this file in subsequent iterations of your design if you want to keep the same pinouts. If you specify a valid guide file name in the Flow Engine window, the pinouts from that file will be used when the design is processed.

Flow Engine Menu Commands

This section describes each Flow engine menu command.

Flow Engine — Flow

The Flow menu contains basic flow control commands as shown in Figure 2-43. Each of these menu commands performs the same function as the corresponding button at the bottom of the Flow Engine dialog window.

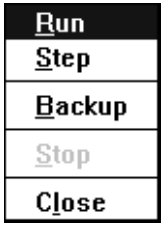


Figure 2-43 The Flow Menu

- **R**un: Select Run to begin the processing of your design. Processing will proceed until the Flow Engine encounters the Stop sign, until the last process is complete, or until a process error has occurred.
- Step: Select Step to process your design one process at a time. After each process is complete, you must click Step again to run the next process.
- Backup: Select Backup to cause the Flow Engine to return to the previous process step as indicated by the process indicator bars. Use this feature to save processing time by backing up to, and re-running, only those steps that you want to change.
- Stop: Select Stop to stop the processing of your design at any time.
- Close: Select Close to close the Flow Engine.

Flow Engine — Setup

Select Setup to specify the Flow Engine options, and you see the menu as shown in Figure 2-44.

Figure 2-44 The Setup Menu

Flow Engine — Setup — Options

Select Options and you see the Design Implementation Dialog Window as shown in Figure 2-45.

Figure 2-45 The Options Window

See the previous section “Design – Implementation” (Figure 2-16) for more information.

Flow Engine — Setup — Font

Select Font and you see the Font Dialog Window as shown in Figure 2-46.

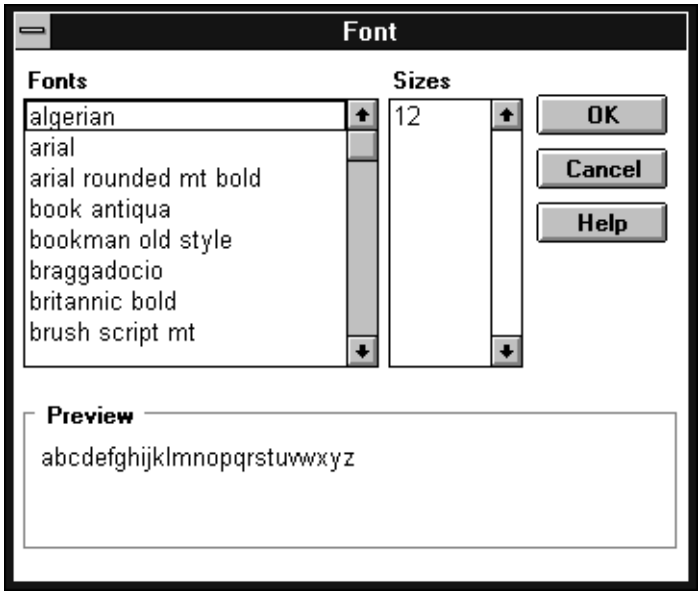


Figure 2-46 The Font Dialog Window

Use this dialog window to select the font type and font size used for displaying messages in the Flow Engine message window.

Flow Engine — Setup — Advanced

Select Advanced and you see the Flow Configuration Dialog Window as shown in Figure 2-47.

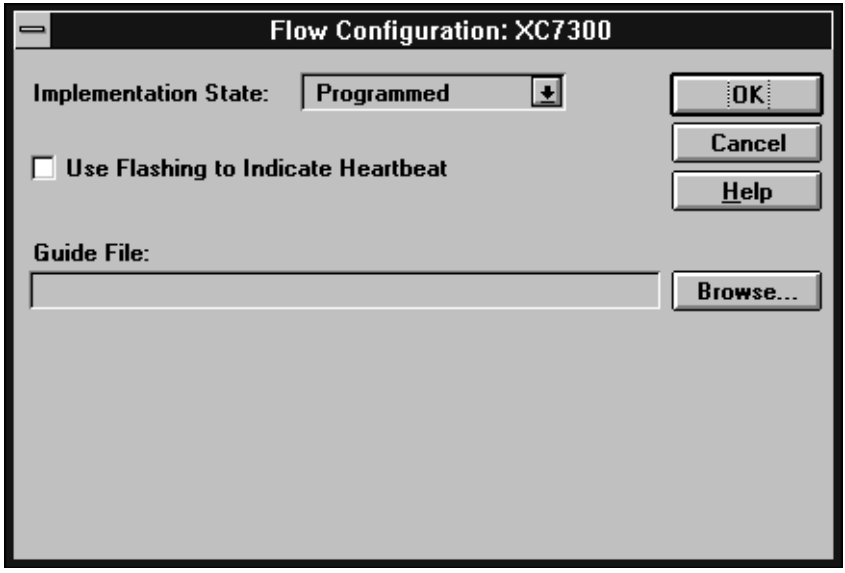


Figure 2-47 The Flow Configuration Dialog Window

From here you can specify the flow control options as follows:

- Implementation State: This allows you to control the state at which the Flow Engine is stopped. The data must already be available in the specified state in order for the Flow Engine to back-up to that state.
- Use Flashing to Indicate Heartbeat: This causes the Process Indicators to flash as the software processes your design.
- Guide File: If you want to use the pinouts from a previous run of your design, specify that guide file here. Click Browse and you see the Open dialog window as shown in Figure 2-48.

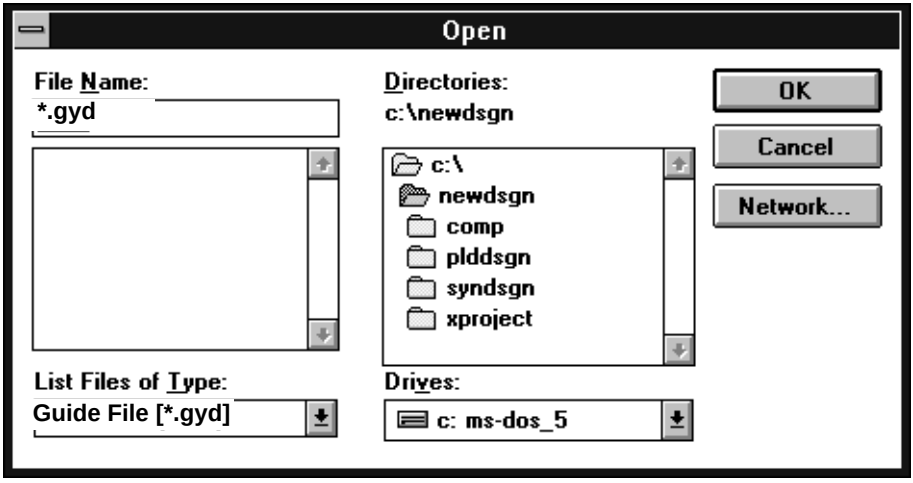


Figure 2-48 The Open Dialog WIndow

Flow Engine — Utilities

Open the Utilities menu and you see the options shown in Figure 2-49. These utilities are the same as those described in the previous section "Design Manager Utilities" (Figure 2-29). See this section for details.

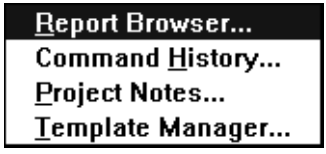


Figure 2-49 The Utilities Menu

Flow Engine — Help

Click Help to bring up the on-line help system.

Chapter 3

Controlling Design Implementation

This chapter shows you how to use the Design Manager to specify fitter guidelines. It also describes the design implementation control options that you can use within behavioral, schematic, and VHDL designs.

Design Implementation Overview

A typical design process is composed of three phases:

1. The fundamental logic design.
2. The implementation of your logic into a target device.
3. The design verification.

Design implementation is controlled during the logic design phase by using schematic attributes, behavioral commands, and VHDL attributes. After your logic design is completed, including any design implementation information, you process your design in the Design Manager environment which provides additional implementation control through Design Manager menu selections.

During design implementation you provide guidelines to the software for:

- Selecting a device or devices to contain your design.
- Placing logic to achieve the most compact design.
- Assigning signals to specific device pin locations.
- Controlling various design optimization options for density or speed improvement.
- Controlling the device power consumption.

- Controlling the slewrate of device output buffers.
- Controlling the initial states of registers.
- Reserving device resources for future design iterations.

The following table shows you the types of design control options and the environments from which they are applied:

Design Control Options	Design Environment			
	Design Manager	Schematic Attributes	Behavioral Commands	VHDL Attributes
Single Device Selection - Manual	X	X	X	X
Single Device Selection - Automatic	X			
Multiple Device Selection	X	Note 2		Note 2
Logic Placement & Pin Assignment	Note 3	X	X	X
Design Optimization Note 5	X	X	X	X
Device Power Management	Note 4	X	X	X
Master Reset Pin Control (MRINPUT)	Note 4	X	X	X
Output Buffer Slewrate		X	X	X
Register Initial State		X	X	X
Resource Reservation	X	Note 1	Note 1	Note 1

Note 1: You can reserve resources for future design iterations in your schematic, behavioral, or VHDL design, by instantiating unused components which act as place holders.

Note 2: There are properties to control this option.

Note 3: Allows you to ignore values in the design.

Note 4: Allows you to globally turn the option on or off.

Note 5: The LOGIC_OPT function is controlled only in the source design file. The TIMING_OPT function is controlled only in Design Manager.

EPLD Device Selection

During design implementation you must manually select target devices or provide a range of devices from which the software can automatically select the best ones to contain your design. The following table shows you all valid device options.

Table 3-1 Device Selection Options

Device	Speed Grades	Package Options
XC7318	-5 -7	PC44 PQ44
XC7336	-5 -7 -10 -12 -15	PC44 PQ44
XC7354	-7 -10 -12 -15	PC44 PC68
XC7372	-7 -10 -12 -15	PC68 PC84 PQ100
XC73108	-7 -10 -12 -15	PC84 PG84 PQ100 PG144 PQ160 BG225
XC73144	-7 -10 -12 -15	PQ160 BG225
XC7336Q	-10 -12 -15	PC44 PQ44 VQ44

Note: The XC7236A is also supported if explicitly targeted in your design file.

The package options are:

- BG — Ball-Grid-Array.
- PC — Leaded Chip Carrier.
- PG — Pin-Grid-Array.
- PQ — Quad Flat Pak.
- VQ — Very Thin Quad Flat Pak.

Working With Single Devices

This section shows you how to specify a single target device in which to implement your design. You can choose one specific device or you can specify a set of possible target devices from which the software will automatically select the most appropriate one.

If your design requires more than one device, please see the following section "Working With Multiple Devices."

Design Manager — Single Device Selection

After you have opened a design, select New Device from the Implementation menu; the device selection menu appears, as shown in Figure 3-1.

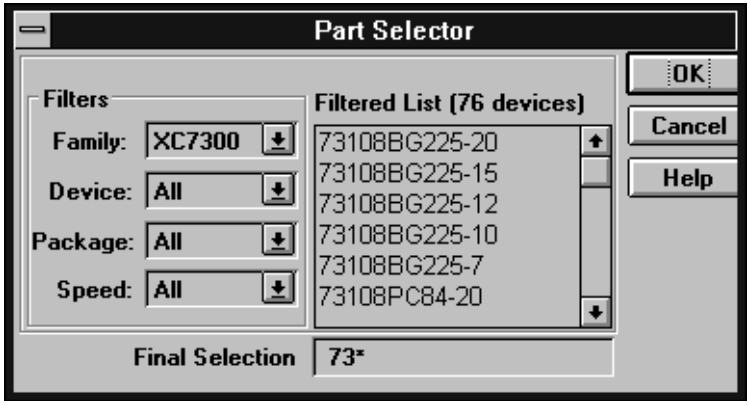


Figure 3-1 Device Selection Menu

From this menu, specify the target device parameters that you want for your design. For example if you choose "All" for all Filters fields the software is free to choose any device in the XC7300 family; in this case there are 76 possible devices to choose from as shown in Figure 3-1.

Note: You can also specify a part number in you design source file. This is described in the following sections.

Software Device Selection Criteria

If you select "All" for any device Filters field the software tries to fit your design within the range of possible devices you have selected by using the following selection criteria:

1. First, the software selects the smallest die.
2. Second, the software selects the smallest package.
3. Third, the software selects the slowest available device that meets your specified timing constraints.

The software will continue to try fitting your design into one of the possible range of selected devices until the first usable one is found or until there are no more devices to try.

Schematic — Device Selection

Use the following attributes in schematic designs to control device selection.

PART or PARTTYPE

For Viewlogic (or Mentor) designs, place the global PART attribute in your schematic to specify a target EPLD device; for OrCAD designs use the PARTTYPE attribute.

Note: Selecting a part type in the Design Manager menu overrides any PART or PARTTYPE attribute in your schematic.

The format of the PART attribute is:

PART=*dddd***-***sspppp*

Where dddd is the device number, *-ss* is the speed grade, and *pppp* is the package type and pin count. For example: XC7354-12PC68

The format of the PARTTYPE attribute is:

PARTTYPE=*dddd***-***sspppp*

Where dddd is the device number, *-ss* is the speed grade, and *pppp* is the package type and pin count. For example: XC7354-12PC68

Behavioral — Device Selection

Use the following attribute in behavioral designs to specify a device type.

CHIP

The CHIP statement can be used to specify a device type. The Syntax is:

CHIP *file_name XCddddpppp*

Where *file_name* is the name of your behavioral design file, XC*dddd* is the device number, and *pppp* is the package type. You cannot specify a speed grade; the software assumes the fastest speed grade for the chosen device.

For example, the following command will target your design to an XC7318-5PL44 device:

```
CHIP  my_design  XC7318PL44
```

Note: If you want to specify a different speed grade (or a different device), you can choose a target device using the Design Manager menus, which will override any part selection in the CHIP statement.

See the Xilinx PLUSASM Reference for a complete description.

VHDL — Device Selection

Use the following attribute in VHDL designs to specify a target device type.

set_attribute

This attribute (for Synopsys only) is used to specify the target EPLD device type for FPGA Compiler only. (If you are using Design Compiler, you must specify the part type by using the -p option of the syn2epld command.)

The format is:

set_attribute *design_name* **part -type string** *dddd-ssppnn*

where:

- *design_name* = the name of the top-level design entity.
- *dddd* = basic device type.
- *ss* = speed grade.
- *pp* = package type.
- *nn* = number of package pins.

For example:

```
set_attribute SCAN part -type string 7354-10PC44
```

Working With Multiple Devices

This section shows you how to manage designs that are too large to fit within a single target device.

Design Manager — Multiple Device Partitioning

The Design Manager contains a Partitioner tool for controlling how your design fits into multiple target devices. To open this tool select the Partitioner icon or select Partitioner from the Tools menu. You will see:

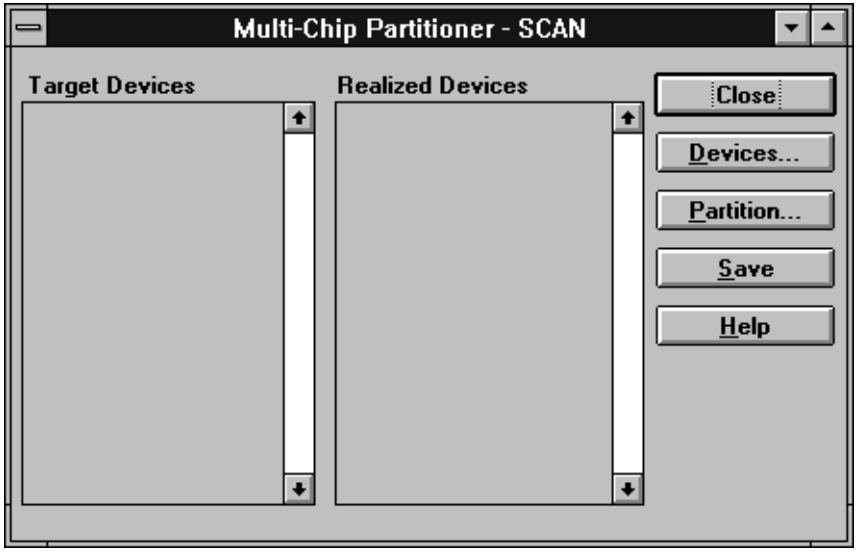


Figure 3-2 Multi-Chip Partitioner WIndow

From here you can select either the entire XC7300 family or a subset of the XC7300 family to be used as possible target devices; these show up in the Target Devices column after selection. The Partitioner then selects the best devices in which to fit your design and displays its choices in the Realized Devices column, after partitioning.

For more information on how to use the Multi-Chip Partitioner tool, see Chapter 2.

Schematic — Multiple Device Partitioning

GROUP

The GROUP attribute is placed onto any internal logic symbol (including macros) in a multi-device design to specify that the logic for that symbol is to be implemented in the same physical device as other symbols assigned to the same *group_name*. Groups of symbols with different *group_name*s can still be mapped into the same physical device unless otherwise prevented by using a CHIP attribute.

The format of this attribute is:

GROUP=*group_name*

CHIP

The CHIP attribute is placed onto any internal logic symbol (including macros) in a multi-device design to specify that the logic for that symbol is to be implemented in a particular named device, exclusive of all other *chip_name*s. That is, symbols with different *chip_name*s are mapped into different physical devices. If a CHIP attribute is placed on a symbol that also has a GROUP attribute, all other symbols with the same *group_name* are automatically assigned to *chip_name*. This allows you to conveniently change the partitioning of groups of logic symbols among physical devices by modifying one CHIP attribute for each group.

You can also specify a physical device (*device_type*) to be used for *chip_name*.

The format of this attribute is:

CHIP=*chip_name* **[@***device_type***]**

Manual Logic Placement

This section describes the schematic attributes, VHDL commands, and behavioral commands that you can use for manual logic placement. Pin assignment, described in the next section, also affects logic placement and therefore you should make sure these operations are coordinated and do not cause conflicting placement requests.

Note: For most designs you will achieve optimal speed and density if you allow the software to place logic automatically without restrictions. However, in some cases you may want to manually place logic to achieve the optimal density. To achieve optimal timing it is usually best to use T-Specs, as described in Chapter 4, rather than manually placing logic.

Design Manager — Logic Placement

The following Design Manager menus contain options that globally affect logic placement:

- Design - Implement
- Utilities - Template Manager

For information on theses menus and their options see Chapter 2.

Schematic — Logic Placement

Use the following attributes in schematics to control logic placement.

F

The F attribute, placed on a component output, specifies that it is to be implemented in a Fast Function Block.

The F attribute produces an error if placed on components that require features only present in High-Density Function Blocks, such as:

- ADD
- ADSU
- ACC
- BUFCE
- IFD
- IFDX1
- ILD
- COMPM
- LD
- FDCP, FDCPE
- OBUFT
- OFDT
- XOR7, XOR8, XOR9

H

The H attribute, placed on a component output, specifies that it is to be implemented in a High Density Function Block.

The F and H attributes are applied to nets in the Workview environment; they are represented by component symbols in the OrCad environment.

LOC=f-b-name_mc-location

The LOC=*f-b-name_mc-location* attribute, placed on a register or internal logic, assigns the signal to a specific function block and macrocell location.

For example, the following attribute places the signal to which it is attached into macrocell 5 of Function Block 3:

```
LOC=FB3_5
```

Behavioral — Logic Placement

Use the following command to control logic placement in behavioral designs.

PARTITION

Use physical PARTITION statements to group signals into a specific function block and optionally into specific consecutive macrocells.

For example, the following statement places the signals A1, A2, A3, and A4, in order, into function block 3 of the target device, beginning on a macrocell location chosen by the software.

```
PARTITION FB3 A1 A2 A3 A4
```

And the following statement places the signals A1, A2, A3, and A4, in order, into function block 4, beginning at macrocell number 3

```
PARTITION FB4_3 A1 A2 A3 A4
```

You can place PARTITION statements in the header of a behavioral design file to control the assignment of equations to macrocells, which also determines the pin assignment of those outputs. All linked equations (equations using either .ADD, .SHIFT, or .EXPORT) must appear in a PARTITION statement; the order in which their

names appear determines the order in which the macrocells are linked. The software will automatically connect the carry chain across function block boundaries if necessary.

VHDL — Logic Placement

Use the following signal attributes to affect logic placement in VHDL designs. These attributes are applied by instantiating the following components. Each of these components has one port (input) which you connect to the signal that receives the attribute.

F

The F attribute indicates either a Fast Function Block output signal or a Fast Input signal. Use this attribute to assign specific functions to EPLD Fast Function Blocks, which provide the highest performance.

To specify that a signal is driven from a Fast Function Block, use:

```
U1: F port map (signal_name);
```

H

The H attribute indicates a High Density Function Block output signal. Use this attribute to assign specific functions to High Density Function Blocks, which provide the most macrocell resources.

To specify that a signal is driven from a High Density Function Block, use:

```
U1: H port map (signal_name);
```

Manual Pin Assignment

This section describes the Design Manager options, schematic attributes, behavioral commands, and VHDL commands that you can use for manual pin assignment. Logic placement, as described in the previous section, must be coordinated with pin assignment to prevent conflicting placement requests.

For most designs you will achieve optimal speed and density if you allow the software to assign pins automatically, without restrictions.

Design Manager — Pin Assignment

The software always saves the pinout of your design, in a file named *design_name*.gyd, which can be used later as a guide file if you choose. You specify a guide design by using the Design-Implementation dialog window or by using the Flow Engine-Setup-Options dialog window.

Schematic — Pin Assignment

Use the following attributes in schematics to control pin assignment.

LOC=pin_name[@chip_name]

The LOC=*pin_name* attribute, placed on a PAD symbol, assigns the signal to a specific device pin and optionally, for multi chip designs, to a specific device (*chip_name*). The PAD symbols are IPAD, OPAD, and IOPAD. The pin name syntax varies according to the package:

- For PC packages: *pin_name* = P*nn*, where *nn* is a pin number. For example, to place a signal on pin 24 use: LOC=P24
- For PG packages: *pin_name* = *rc* where *rc* is row/column. For example, to place a signal on the pin at row G, column 2 use: LOC=G2

Pin assignments are unconditional; the software will not attempt to relocate a pin if it cannot achieve the specified assignment.

You can apply the LOC attribute to as many PADs in your design as you like. However, each pin assignment further constrains the software, making it more difficult to achieve the maximum density for your design.

Note: Pin assignment using the LOC attribute is not supported for bus components such as OBUF8.

For example, in a multi-device design, the following attribute places the signal to which it is attached onto pin 25 of the device named "MUX_2" (which is in a PC package):

```
LOC=P25@MUX_2
```

In a single device design, the following attribute places the signal to which it is attached onto the pin at row F, column 24 of the target device (which is in a PG package):

```
LOC=F24
```

you can have multiple pin specifications in a LOC attribute, by separating them with a semicolon. For Example:

```
LOC=P25@MUX_2;G2@CNTR_1
```

Behavioral — Pin Assignment

Use the following commands to control pin assignment in behavioral designs.

INPUTPIN

The INPUTPIN statement assigns signals to device input pins.

OUTPUTPIN

The OUTPUTPIN statement assigns signals to device output pins.

IOPIN

The IOPIN statement assigns signals to device I/O pins.

FOEPIN

The FOEPIN statement assigns 3-state output enable signals to the device Fast Output Enable pins.

FASTCLOCK

The FASTCLOCK statement assigns clock signals to the device FastCLK pins.

CEPIN

The CEPIN statement assigns clock enable signals to the device Clock Enable pins.

Assign specific device pins to signals by appending the PIN keyword to any pin declaration statement. For example:

```
INPUTPIN (RCLK=C) X Y Z PIN 9 10 11
OUTPUTPIN A B C PIN 9 10 11
IOPIN D E F PIN 9 10 11
```

```
FOEPIN F PIN 32
FASTCLOCK CLKIN PIN 27
CEPIN G
```

All physically possible pin assignments are accepted regardless of their effect on device partitioning or mapping efficiency. Pin assignments within pin declaration statements override pin assignments from the pin-save file.

Note: Pin assignment affects logic placement, and therefore you must coordinate manual logic placement with manual pin assignment to prevent conflicts.

VHDL — Pin Assignment

Use the following attributes to control pin assignment in VHDL designs.

set_attribute

This (Synopsys only) signal attribute is used to specify the pins on which to place output signals. The format is:

```
set_attribute port_name pad_location -type
     string pin_number
```

where:

- *port_name* = The name of the top-level design port.

The format of *pin_number* is:

- **P***nn* for PC and PQ packages, where *nn* is the pin number.
- *rc* for PG and BG packages, where *rc* are the row letter and column number.

For example, for PC and PQ packages:

```
set_attribute RDY pad_location -type string P23
```

For example, for PG and BG packages:

```
set_attribute RDY pad_location -type string K13
```

Design Optimization Control

There are a number of optimization control options the software can use before fitting your design. Each of these options is described in this section.

Design Manager — Design Optimization Control

To access the Design Manager global design optimization control options, do the following:

1. Open the DESIGN MANAGER window.
2. Select Design from the menu bar.
3. Select Implement from the Implementation menu, which brings up the XC7300 Design Implementation Dialog Window, as follows:

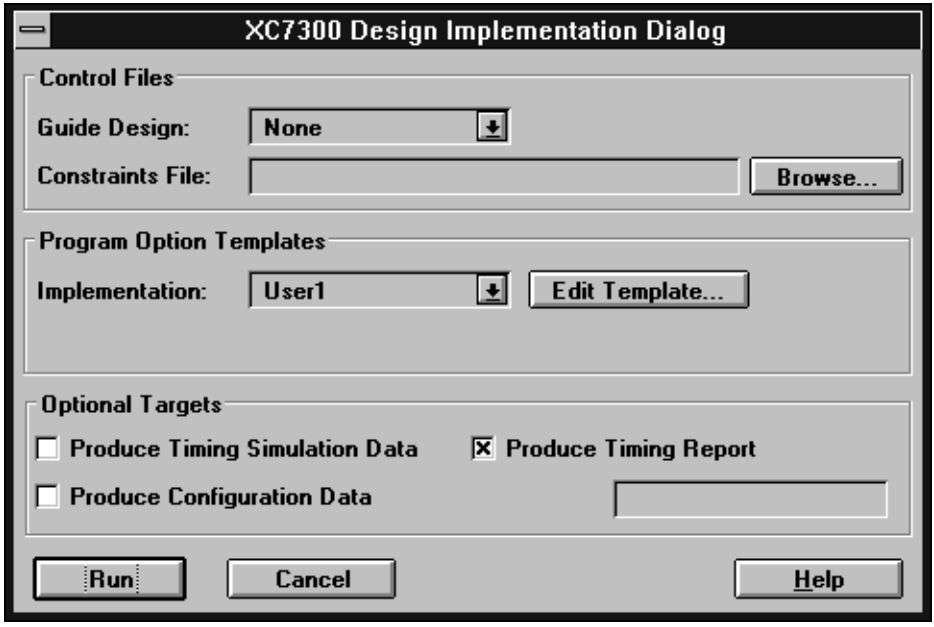


Figure 3-3 XC7300 Design Implementation Dialog Window

4. Select Edit Template which brings up the XC7000 Implementation Template Window (for Fitting), as follows:

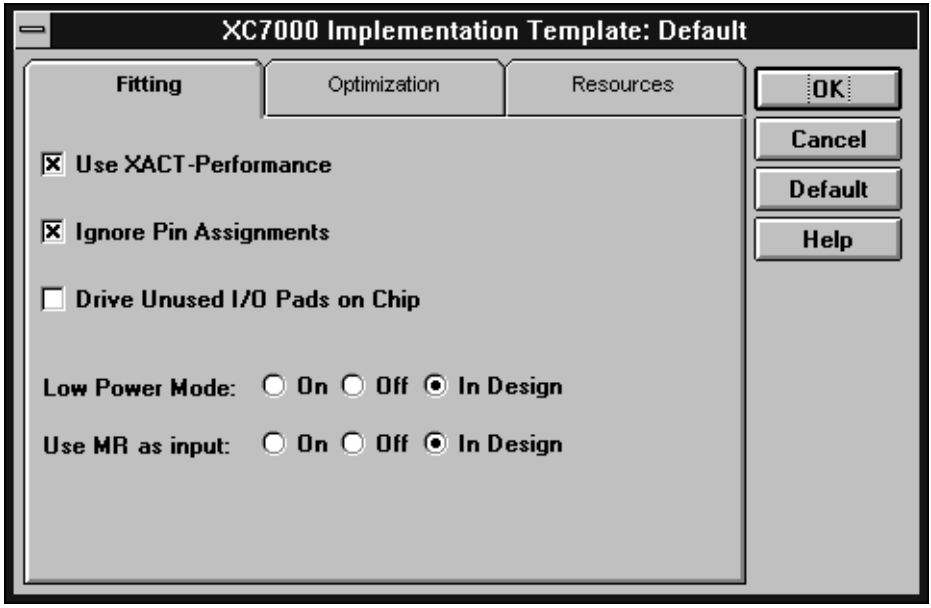


Figure 3-4 XC7000 Implementation Template — Fitting

5. Select Optimization, which brings up the optimization window, as follows:

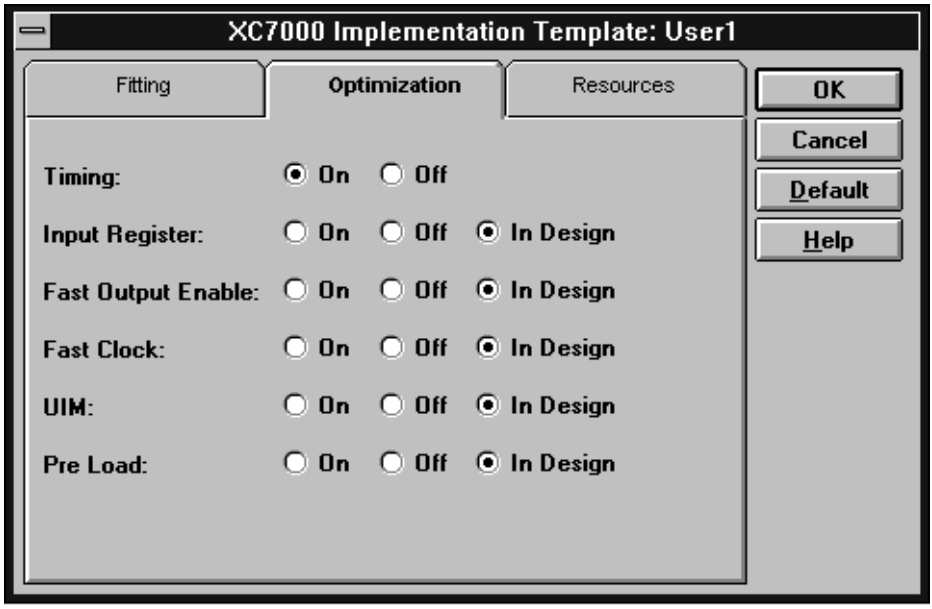


Figure 3-5 XC7300 Implementation Template — Optimization

From here you can control the global design optimization options which are set as follows:

- On — The option is activated for the whole design and corresponding schematic attributes, behavioral commands, or VHDL commands are ignored.
- Off — The option is turned off for the whole design and corresponding schematic attributes, behavioral commands, or VHDL commands are ignored.
- In Design — The option is controlled exclusively by schematic attributes, behavioral commands, or VHDL commands. If the option is not specified in the design, it defaults to On.

The Optimization options are described as follows:

Timing

The Timing option determines if the fitter will try to optimize your design to achieve the best possible timing for your critical paths. If this option is off, the fitter will try to achieve the best possible density, without regard for timing.

This timing optimization is performed independently from the timing-driven optimization of XACT Performance, which was described in Chapter 2.

Input Register

The Input Register option controls register optimization for the entire design. Input Register ON is the default.

Input register optimization reduces the number of macrocells in a design by moving simple FD registers connected to IBUFs into a pad register (provided that the IBUF has no other fanouts). The clock that controls the register must be a FastCLK or an input that can be assigned to a FastCLK pin.

Fast Output Enable

The Fast Output Enable option controls FOE (Fast Output Enable) optimization for the entire design. Fast Output Enable ON is the default.

FOE optimization changes a product-term 3-state signal to an FOE global control signal. Like FastCLK assignment, this reduces the number of UIM inputs and product terms required by each Function Block. FOE optimization generally applies only to BUFE, OBUFE, or 3-state PLD outputs driving an OBUF.

Fast Clock

The Fast Clock option controls FastCLK optimization for the entire design. Fast Clock ON is the default.

FastCLK optimization changes a product-term clock to a FastCLK global signal, which reduces the number of UIM inputs and product terms required by each Function Block.

UIM

The UIM option controls the placement of the Boolean functions into the UIM. The software automatically places all possible product terms into the UIM to achieve the maximum possible density.

Pre Load

The Pre Load option allows the XEPLD software to select register preload values to match the values supported by specified device resources such as FFBs and input registers. The default (Pre Load: ON) allows the XEPLD software to map your design most efficiently, using the device resources most suited to your design.

If you specify Pre Load: OFF, it may prevent registers from mapping into FFBs, and attempting to force such a register into an FFB (using the F attribute or through pin assignment) results in an error.

Note: The software always follows any preload values explicitly defined in your design. For example, the Pre Load Off option specifies a preload value of zero for all registers that do not have values explicitly defined.

Schematic — Design Optimization Control

Use the following attributes to control the automatic optimization of your schematic design.

OPT=OFF|ON|UIM

The OPT=OFF component attribute inhibits logic optimization of all macrocells used by the symbol to which it is attached. OPT=ON overrides the LOGIC_OPT=OFF global attribute for individual symbols. This attribute has no effect on any symbol that contains no macrocell logic, such as an I/O buffer.

Logic optimization collapses logic forward into succeeding macrocells, where possible, to increase the available logic resources.

The OPT=UIM component attribute forces the placement of the Boolean function, to which it is attached, into the UIM. The software automatically places all possible product terms into the UIM to achieve the maximum possible density. However, by using this attribute, you can force specific functions (AND, NAND, OR, NOR, invert) into the UIM, if needed.

LOGIC_OPT=OFF

The LOGIC_OPT=OFF global attribute inhibits Boolean logic optimization for the whole design. LOGIC_OPT=ON is the default.

You can override the LOGIC_OPT=OFF setting for individual symbols by using the OPT=ON attribute.

Logic optimization collapses logic forward into succeeding macrocells, where possible, to increase the available logic resources.

UIM_OPT=OFF

The UIM_OPT=OFF global attribute inhibits UIM optimization for the entire design. UIM_OPT=ON is the default.

UIM optimization extracts Boolean expressions and inverters out of macrocell logic functions and moves them into the UIM, which reduces the use of Function Block resources.

MINIM=OFF

The MINIM=OFF attribute tells the fitter to disable logic minimization for the attached component. For example, if you have MINIMIZE=ON, you need to use the MINIM=OFF attribute if you want to specify redundant logic in a portion of your design to avoid a potential race condition such as when designing combinational feedback loops and latches.

If you have global minimization turned off (MINIMIZE=OFF) then you do not need to use the MINIM=OFF attribute on specific logic.

MINIMIZE=OFF

The MINIMIZE=OFF global attribute inhibits logic minimization for the whole design. MINIMIZE=ON is the default.

Minimization eliminates any redundant or non-effective logic through Boolean minimization.

FOE_OPT=OFF

The FOE_OPT=OFF global attribute inhibits Fast Output Enable optimization for the entire design. FOE_OPT=ON is the default.

FOE optimization changes a product-term tri-state signal to an FOE global control signal. Like FastCLK assignment, this reduces the number of UIM inputs and product terms required by each Function Block. FOE optimization generally applies only to BUFE, OBUFE, or tri-state PLD outputs driving an OBUF.

CLOCK_OPT=OFF

The CLOCK_OPT=OFF global attribute inhibits FastCLK optimization for the entire design. CLOCK_OPT=ON is the default.

FastCLK optimization changes a product-term clock to a FastCLK global signal, which reduces the number of UIM inputs and product terms required by each Function Block.

REG_OPT=OFF

The REG_OPT=OFF global attribute inhibits input register optimization for the entire design. REG_OPT=ON is the default.

Input register optimization reduces the number of macrocells in a design by moving simple FD registers connected to IBUFs into a pad register (provided that the IBUF has no other fanouts). The clock that controls the register must be a FastCLK or an input that can be assigned to a FastCLK pin.

PRELOAD_OPT=OFF

The PRELOAD_OPT=OFF global attribute allows the XEPLD software to select register preload values to match the values supported by specified device resources such as FFBs and input registers. The default (PRELOAD_OPT=ON) allows the XEPLD software to map your design most efficiently, using the device resources most suited to the elements of your design.

If you specify PRELOAD_OPT=OFF, it may prevent registers from mapping into FFBs, and attempting to force such a register into an

FFB (using the F attribute or through pin assignment) results in an error.

Note: The software always follows any preload values explicitly defined in your design. For example, the PRELOAD_OPT=OFF option specifies a preload value of zero for all registers that do not have values explicitly defined.

Behavioral — Design Optimization Control

Use the following commands to control optimization in behavioral designs.

LOGIC_OPT

The LOGIC_OPT statement inhibits the automatic logic collapsing function of the XEPLD software. This can be used globally on all equations or explicitly on individual equations.

MINIMIZE

The MINIMIZE statement selectively enables or disables Boolean equation minimization.

OPTIONS

The OPTIONS statement inhibits various XEPLD optimization routines for your design.

See the Xilinx PLUSASM Reference for more information.

VHDL — Design Optimization Control

Use the following VHDL attributes to control design optimization.

NO_FOE

The global NO_FOE attribute controls the automatic use of global Fast Output Enable inputs. If the NO_FOE attribute is specified, it indicates that the software will not automatically assign 3-state control input signals to the global FOE device inputs. If NO_FOE is not specified, the software assigns the 3-state control signals in your design to the global FOE inputs of the device, if possible.

To specify that no 3-state control signals will automatically be assigned to the global FOE inputs, instantiate the NO_FOE component as follows:

```
U1: NO_FOE;
```

where U1 is any instance name.

NO_FCLK

The global NO_FCLK attribute controls the automatic use of global FastClock inputs. If the NO_FCLK attribute is specified, it indicates that the software will not automatically assign clock signals to the global FastClock inputs. If NO_FCLK is not specified, the software will assign clock signals in your design to the dedicated FastCLK inputs of the device, if possible.

To specify that no clock signals will automatically be assigned to the global FastClock nets, instantiate the NO_FCLK component as follows:

```
U1: NO_FCLK;
```

where U1 is any instance name.

NO_IFD

The global NO_IFD attribute controls the automatic usage of input pad registers. If the NO_IFD attribute is specified, it indicates that the software will not automatically use the registers in the input pads. If NO_IFD is not specified, the software will assign registers in your design to the input pads whenever possible, to reduce macrocell resource requirements.

To specify that registers will not be automatically placed into the input pads, instantiate the NO_IFD component as follows:

```
U1: NO_IFD;
```

where U1 is any instance name.

OPT_OFF

The OPT_OFF signal attribute inhibits the software from optimizing the cell that drives the connected signal.

To specify that a signal is to remain as a macrocell output, use:

```
U1: OPT_OFF port map (signal_name);
```

where U1 is any instance name.

OPT_UIM

The OPT_UIM signal attribute forces the software to place the specified AND gate into the UIM. It can only be connected to a signal that originates from an AND gate.

To specify that an AND gate is to be implemented in the UIM, instantiate the OPT_UIM component as follows:

```
U1: UIM_OPT port map (signal_name);
```

where U1 is any instance name.

Note: The optimization of AND gates into the UIM is done automatically by the fitter whenever possible.

Power Management

You can control the power consumption and speed of each component in your design. If you select low power, your speed is decreased. If you select high speed, power consumption is increased. These options can be applied to individual components by using Schematic attributes, behavioral commands, or VHDL commands.

Design Manager — Power Management

To access the Design Manager global Low Power Mode option, do the following:

1. Open the DESIGN MANAGER window.
2. Select Design from the tool bar.
3. Select Implement from the Design menu, which brings up the XC7300 Design Implementation Dialog Window, as shown in Figure 3-6.

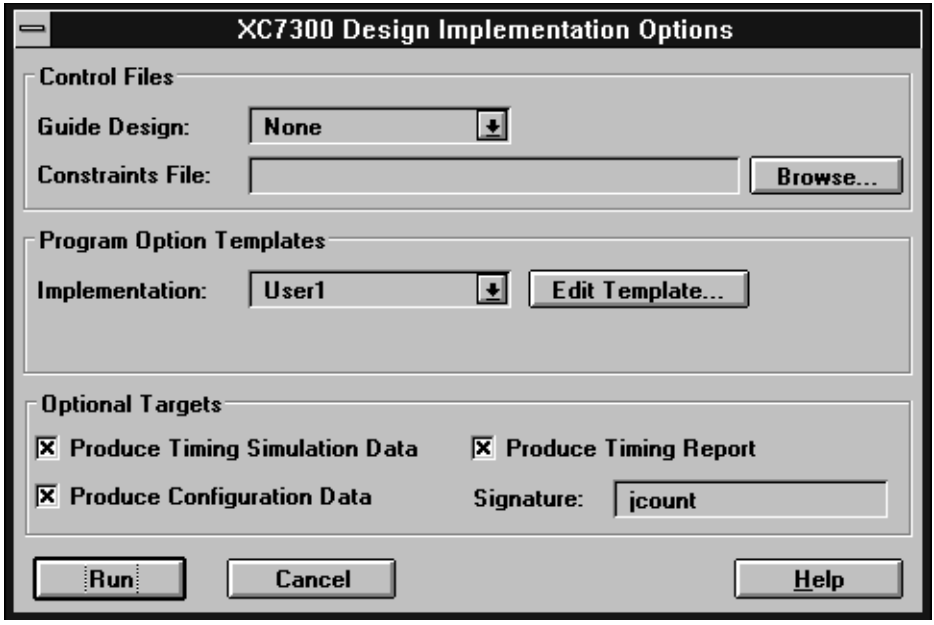


Figure 3-6 XC7300 Design Implementation Dialog Window

4. Select Edit Template which brings up the XC7000 Implementation Template Window for Fitting, as follows:

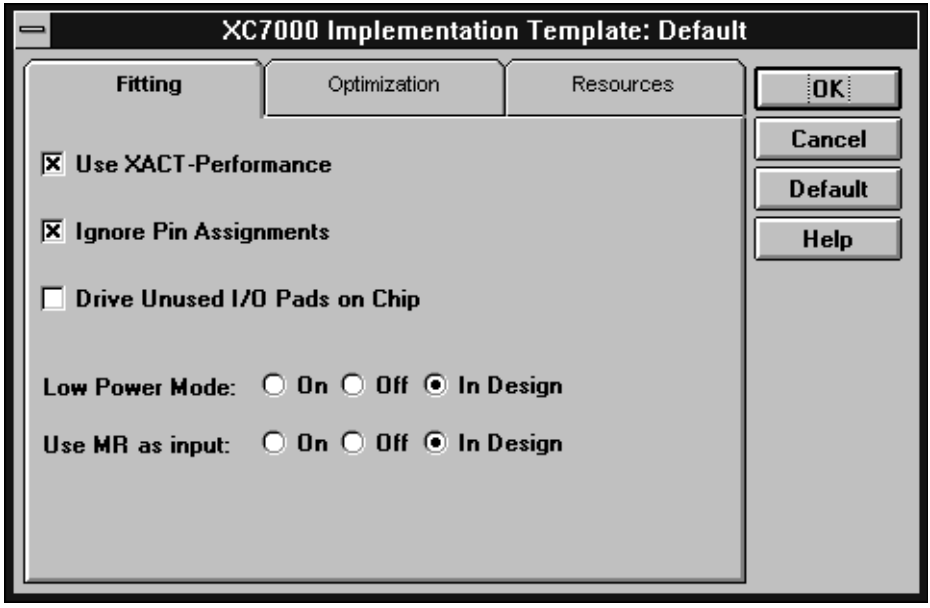


Figure 3-7 XC7000 Implementation Template — Fitting

From here you can control the Low Power Mode option which is set as follows:

- On — Low power is activated for the whole design and schematic attributes, behavioral commands, or VHDL commands for power management are ignored.
- Off — High speed is activated for the whole design corresponding schematic attributes, behavioral commands, or VHDL commands for power management are ignored.
- In Design — Device power is controlled exclusively by schematic attributes, behavioral commands, or VHDL commands. If power management options are not specified in the design, it defaults to Off.

Schematic — Power Management

Use the following schematic attribute to control device power consumption.

LOWPWR=ON | OFF | ALL

The LOWPWR attribute controls the power consumption for individual components (ON | OFF) or globally for all components (ALL). The default is LOWPWR=OFF (high speed) for all macrocells used in the design unless otherwise specified.

Use LOWPWR=ALL to make low power (slower speed) the global power setting.

To control the power setting of individual components, use LOWPWR=ON (if the default is LOWPWR=OFF) or LOWPWR=OFF (if the global attribute has been set to LOWPWR=ALL). This attribute is ignored if assigned to a symbol that uses no macrocells, such as an inverter, Boolean functions that are optimized into the UIM, and input registers. This attribute is valid only for XC7300 designs.

Behavioral — Power Management

Use the following command to control device power consumption in behavioral designs.

PWR

The PWR statement specifies whether macrocells implementing the specified equations have LOW-power or standard-power (STD) operation.

See the Xilinx PLUSASM Reference for more information.

VHDL — Power Management

Use the following VHDL global attribute to control the device power consumption.

LOWPWR

The LOWPWR attribute controls the macrocell power usage. If the LOWPWR attribute is specified it indicates that all macrocells have

low power operation. If LOWPWR is not specified, the macrocells have standard power operation.

To specify low power operation for all macrocells, instantiate the LOWPWR component as follows:

```
U1: LOWPWR;
```

where U1 is any instance name.

Slewrate Control

You can control the slew rate of your the output buffers in you design by using schematic attributes, behavioral commands, or VHDL commands only.

Schematic — Slewrate Control

Use the following global attribute to control the speed of your output buffers for XC73144 designs.

FAST

The Fast attribute causes the output buffers to switch as fast as possible.

Note: The output buffer slewrate defaults to slow for the XC73144; for all other devices the output slew rate is always fast.

Behavioral — Slewrate Control

Use the following behavioral command to control the speed of your output buffers for XC73144 designs.

FAST

The Fast command causes the output buffers to switch as fast as possible.

VHDL — Slewrate Control

Use the following signal attribute (for Synopsys only) to control the slew rate of the output buffers in your VHDL design.

set_pad_type

The SET_PAD_TYPE attribute, when used with the -slewrate option, controls the output buffer slew rate. The options are:

- *HIGH* — Slows the output signal transition time and thus reduces internal ground-bounce noise.
- *NONE* — Speeds up the output signal transition time. (The default is for fast transition times.)

The output buffers (`IBUF`, `OBUF`, and `IOBUFE`) default to fast transition outputs. However, in order to reduce possible ground-bounce problems, it is recommended that you use the fast transition default only for those output signals that require maximum speed.

To set all outputs for slow slew rate, use the following command:

```
set_pad_type -slewrate HIGH all_outputs ()
```

Use this command after specifying the `set_port_is_pad` command and before implementing the insert_pads command.

After you have globally changed all outputs to the HIGH option (for slow signal transition) you can set any individual output for fast signal transition by using the following command:

`set_pad_type -slewrate NONE` *port_name*

If you need fast transition time on most of your outputs, you can leave the default slew rate as NONE and slow only the selected outputs, by using the following command:

`set_pad_type -slewrate HIGH` *port_name*

Note: Synopsys and Xilinx define the slew rate using opposite terms. The Synopsys HIGH attribute translates to a SLOW Xilinx slew rate. The Synopsys NONE attribute translates to a FAST Xilinx slew rate.

Initial State Control

This section describes how to control the initial state of the registers in your design so that it will power-up into a known state.

Schematic — Initial State Control

Use the following attributes to control the initial states of registers in your schematic design.

INIT=R | S

The INIT=R (for a 0 value) or INIT=S (for a 1 value) component attributes specify the preload value of the registered component to which they are attached.

The INIT attributes are always obeyed by the fitter, regardless of the PRELOAD_OPT attribute, but are ignored in functional simulation.

Note: You cannot change the preload value of an input register to 0 using the INIT=R attribute because input registers physically do not support preload to the 0 state. Also, if you specify INIT=R to control preload values, it prevents registers from mapping into FFBs, and attempting to force such a register into an FFB (using the F attribute or through pin assignment) results in an error.

PRELOAD_OPT

The PRELOAD_OPT=OFF global attribute allows the XEPLD software to select register preload values to match the values supported by specified device resources such as FFBs and input registers. The default (PRELOAD_OPT=ON) allows the XEPLD software to map your design most efficiently, using the device resources most suited to the elements of your design.

If you specify PRELOAD_OPT=OFF, it may prevent registers from mapping into FFBs, and attempting to force such a register into an FFB (using the F attribute or through pin assignment) results in an error.

Note: The software always follows any preload values explicitly defined in your design. For example, the PRELOAD_OPT=OFF option specifies a preload value of zero for all registers that do not have values explicitly defined.

Behavioral — Initial State Control

Use the following command to control the initial states of registers in behavioral design.

signal_name.PRLD

The .PRLD extension specifies the initial logic value to be preloaded into the macrocell register during power-up (VCC=1, GND=0). The syntax is:

signal_name`.PRLD = VCC | GND`

See the Xilinx PLUSASM Reference for more detailed information.

VHDL — Initial State Control

Use the following VHDL attributes to control the initial states of registers in your design.

PRELOAD

The global PRELOAD attribute controls the use of default initial state values for registers in your design. If PRELOAD is specified, the software uses the default initial states for each register as shown in the library specifications. If the PRELOAD attribute is not specified, it indicates that the software may change the initial states of registers (unless explicitly specified) if the change allows a more efficient implementation.

To prevent the software from changing the initial states of registers in your design, instantiate the PRELOAD component as follows:

```
U1: PRELOAD;
```

where U1 is any instance name.

INIT=R | S

The INIT=R (for a 0 value) or INIT=S (for a 1 value) component attributes specify the preload value of the registered component to which they are attached.

The INIT attributes are always obeyed by the fitter, regardless of the PRELOAD_OPT attribute.

Note: You cannot change the preload value of an input register to 0 using the INIT=R attribute because input registers physically do not support preload to the 0 state. Also, if you specify INIT=R to control preload values, it prevents registers from mapping into FFBs, and attempting to force such a register into an FFB (using the F attribute or through pin assignment) results in an error.

set_attribute

This attribute is used to specify the initial (power up) state of registers for Synopsys designs only.

The format is:

```
set_attribute instance_name
     fpga_xilinx_init_state -type string state
```

where:

- *instance_name* is the name of a register instance
- *state* is either S (set) or R (reset)

For example:

```
set_attribute “QOUT_reg<2>”
     fpga_xilinx_init_state -type string S
```

You can also use the Synopsys **find** function to specify a set of instance names using wildcards, for example:

```
set_attribute find (cell QOUT*)
     fpga_xilinx_init_state -type string S
```

Resource Reservation for Design Iteration

This section shows you how to reserve macrocells for design iteration and for use in multi-chip partitioning.

Design Manager — Resource Reservation

To access the Design Manager Resources options, do the following:

1. Open the DESIGN MANAGER window.
2. Select Design from the tool bar.
3. Select Implement from the Design menu, which brings up the XC7300 Design Implementation Dialog Window, as follows:

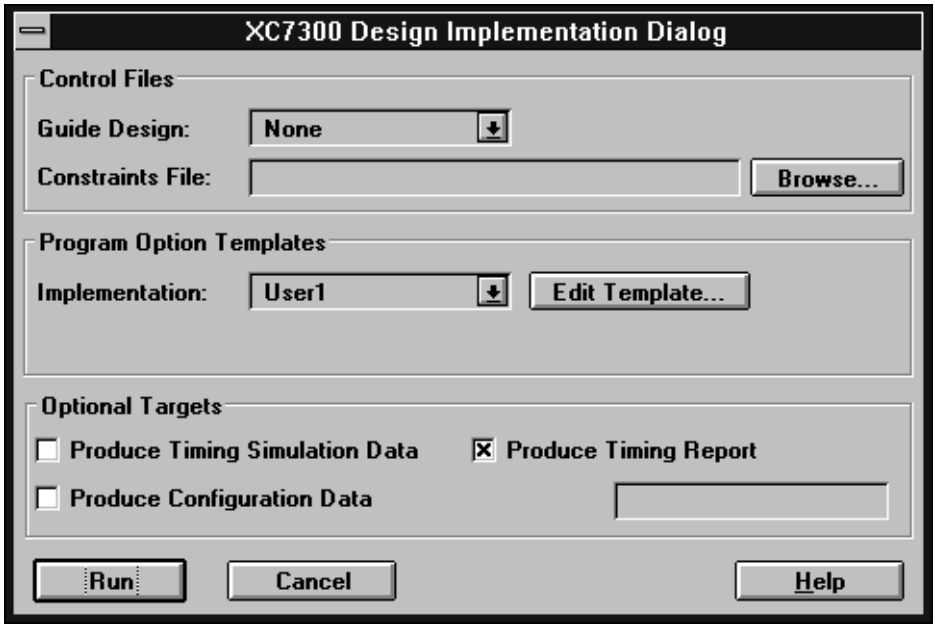


Figure 3-8 XC7300 Design Implementation Dialog Window

4. Select Edit Template which brings up the XC7000 Implementation Window for Fitting, as follows:

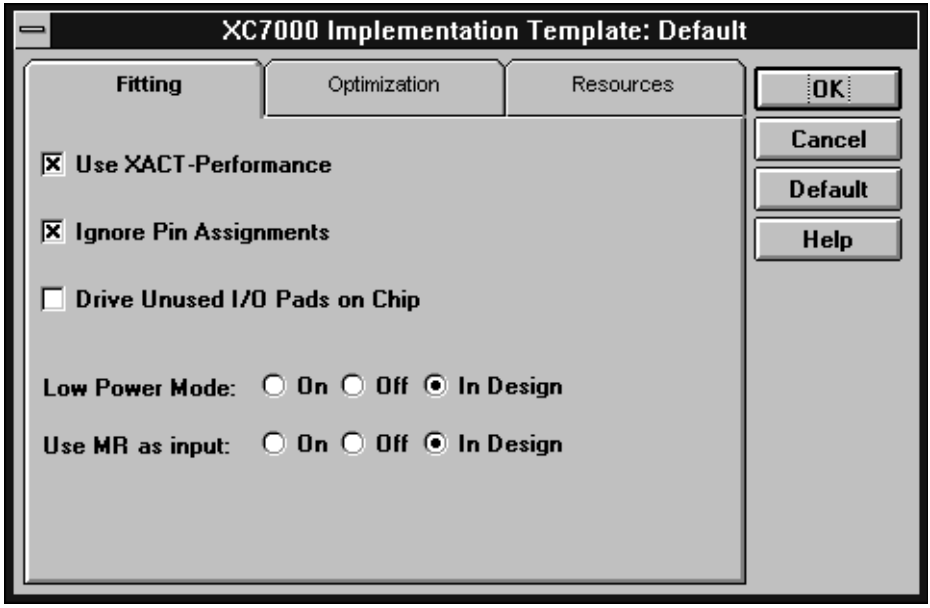


Figure 3-9 XC7000 Implementation Template — Fitting

5. Select Resources which brings up the following window:

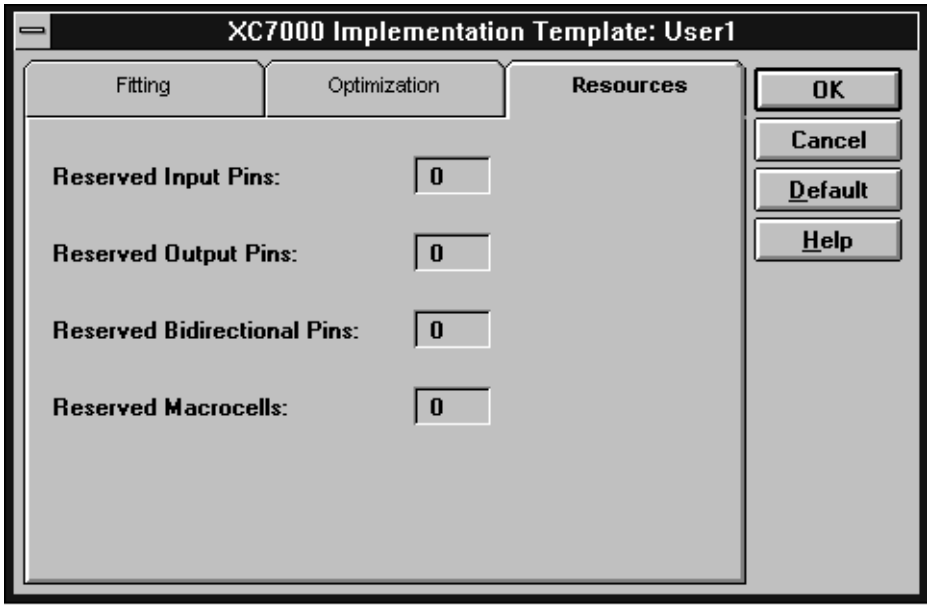


Figure 3-10 Resources Dialog Window

From here you can control the amount of device resources that will be left unused in each device. If your design uses more than one device, these resource reservations are applied to each device; the resource reservations are not divided among the total number of devices used. The resource reservation options are:

Reserved Input Pins

This specifies the total number of reserved input pins including Fast Input pins that are reserved in each device.

Reserved Output Pins

This specifies the total number of output pins reserved in each device including those connected to Fast Function Blocks and those connected to High Density Function Blocks.

Reserved Bidirectional Pins

This specifies the total number of I/O pins reserved in each device including those connected to Fast Function Blocks and those connected to High Density Function Blocks.

Reserved Macrocells

This specifies the total number of macrocells reserved in each device including those contained in Fast Function Blocks and those contained in High Density Function Blocks.

Chapter 4

Controlling Design Timing

You can use Timing Specifications (T-Specs) to specify the maximum allowable delay between groups of components in your design. The software then places and routes your design to achieve the timing defined by these specifications. This Xilinx EPLD timing-driven optimization is called XACT-Performance.

T-Specs can be applied directly to a schematic design or they can be specified in a constraints file for behavioral and VHDL designs. You can then globally enable or disable T-Specs through Design Manager menu options.

Design Manager — Timing Control

To enable global timing-driven optimization (XACT-Performance), do the following:

1. Open the DESIGN MANAGER window.
2. Select Design from the menu bar.
3. Select Implement from the Implementation menu, which brings up the XC7300 Design Implementation Dialog Window as follows:

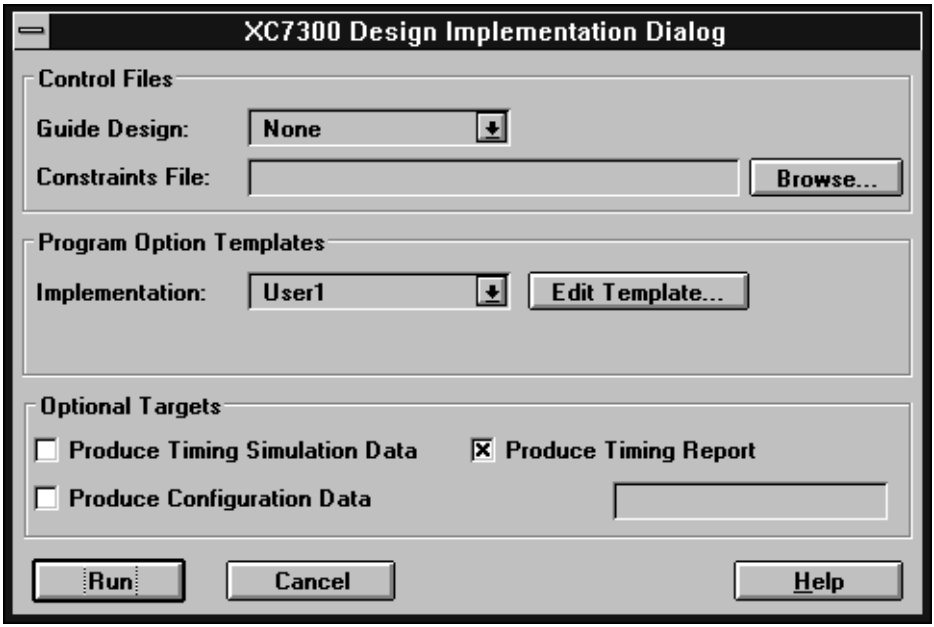


Figure 4-1 XC7300 Design Implementation Dialog Window

4. Select Edit Template, which brings up the XC7300 Implementation Template Window for Fitting as follows:

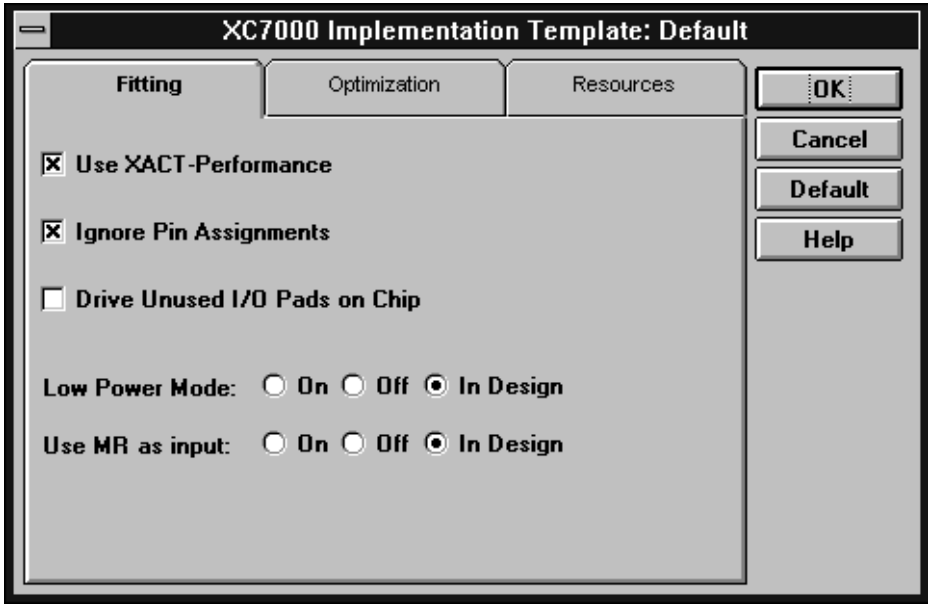


Figure 4-2 XC7300 Implementation Template — Fitting

5. If you place an X on Use XACT Performance the software will use your T-Spec information (if it exists) when fitting your design. If you do not select this option, the software will ignore all T-Spec information.

Schematic — Timing Control

This section shows you how to specify timing control information in schematic designs.

Defining Timing Specifications

Timing specifications are placed on your schematic using a TIMESPEC primitive that contains the Time Spec Attribute Definitions, which control the timing for paths between defined groups of components.

The TIMESPEC Primitive

The TIMESPEC primitive, as illustrated in Figure 4-3, is 30 characters wide and contains TS attribute definitions. Each TIMESPEC primitive can hold up to eight TS attribute definitions. If you want to include more than eight TS attribute definitions, you can use multiple TIMESPEC primitives in your schematic.

Note: Though the TIMESPEC primitive is only 30 characters wide, you can create TS attribute definitions of any length by continuing on the next line.

TIMESPEC
TS01=FROM:FFS:TO:PADS=25

Figure 4-3 TIMESPEC Primitive

How you add a TIMESPEC primitive to your schematic depends on your specific schematic-entry software. Refer to the appropriate Xilinx Interface User Guide for step-by-step instructions.

TIMESPEC Attribute Syntax

The TIMESPEC attribute definitions specify the maximum delay between groups of components. They begin with the letters "TS" and a unique identifier that can consist of letters, numbers, or the underscore character (_). The From-To statement specifies the timing requirements between specific end points. The full syntax is shown as follows:

TS*identifier***=FROM:***group1***:TO:***group2***=***delay

The parameters *group1* and *group2* can be any of the following:

- Predefined groups consisting of FFS, LATCHES, or PADS, which are discussed in the "Using Predefined Groups" section.

- Previously created TNM identifiers, which are introduced in the "Creating Arbitrary Groups Using TNMs" section.
- Groups defined in TIMEGRP symbols, which are introduced in the "Creating New Groups from Existing Groups" section.

The *delay* parameter defines the maximum delay for the attribute, using nanoseconds as the default unit of measurement. Other units of measurement such as MHZ may also be used; see the section "Specifying Time Delay Units" later in this chapter.

Note: Keywords, such as FROM and TO, appear in this document in upper case; however, you can enter them in the TIMESPEC primitive in either upper or lower case.

The following examples show typical TIMESPEC attribute definitions:

```
TS01=FROM:FFS:TO:FFS=30
TS_OTHER=FROM:PADS:TO:FFS=25
TS_THAT=FROM:PADS:TO:LATCHES=35
```

Note: Latches refer to input latches only.

OrCAD Users — The TIMESPEC primitive does not exist in the Xilinx OrCAD library. For details on how to use Time Specs in an OrCad design, refer to the *OrCAD Interface User Guide*.

Mentor Graphics Users — The term *attribute* in this chapter is equivalent to *property* as used in the Mentor Graphics environment.

Defining Timing Path End Points

Specify the start and end points of your timing paths using one of the following methods:

- Refer to a predefined group by specifying one of the corresponding keywords — FFS, PADS, LATCHES.
- Create arbitrary groups within a predefined group by tagging symbols with TNM (pronounced *tee-name*) attributes.
- Create groups that are combinations of existing groups by using TIMEGRP symbols.
- Create groups by pattern matching on signal names.

The following sections discuss each method in detail.

Using Predefined Groups

You can refer to a group of flip-flops, input latches, or I/O pads, by using the corresponding keywords:

Keyword	Group
FFS	Macrocell or IOB flip-flops
LATCHES	input latches only; not latches built from function generators
PADS	input/output pads

These predefined groups represent all symbols of that type. For example the following TS Attribute means that the delay between any two flip-flops must be no greater than 30 ns.

```
TS01=FROM:FFS:TO:FFS=30
```

And the following TS attribute means that the delay between any I/O pad and any flip-flop must be no greater than 25ns.

```
TS_OTHER=FROM:PADS:TO:FFS=25
```

To create groups that are more specific, see the next section, "Creating Arbitrary Groups Using TNMs."

Creating Arbitrary Groups Using TNMs

A TNM (T-Name or timing name) is a flag that you place directly on your schematic to tag specific pads, flip-flops, or input latches. All symbols tagged with the TNM identifier are considered a group. Place TNM attributes directly on your schematic using the following syntax:

TNM=*identifier*

where *identifier* is a value that consists of any combination of letters, numbers, or underscores. It is usually a good idea to keep the TNM short for convenience and clarity. For example:

```
TNM=grp_1
```

Warning: Do not use reserved words, such as FFS, LATCHES, or PADS, for TNM identifiers.

You can specify as many groups of end points as you need.

You can tag groups of end points by placing TNM identifiers on:

- primitive symbols
- macro symbols
- signals or pins

The method you choose depends on how the path end points are related in your design.

Placing TNMs on Primitive Symbols

You can group individual logic primitives explicitly by flagging each symbol, as illustrated in the following figure:

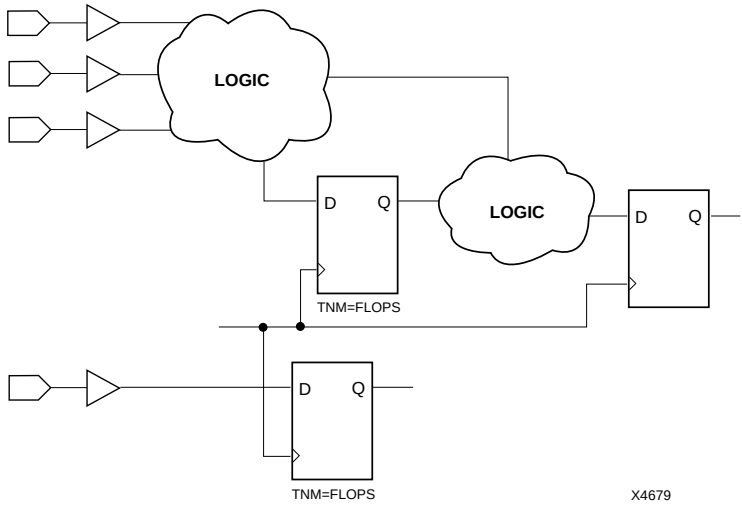


Figure 4-4 Placing a TNM on Primitive Symbols

In Figure 4-4, the flip-flops tagged with the TNM form a group called "FLOPS." The untagged flip-flop is not part of the group.

Note: You cannot use TNMs to group instances of incompatible symbols; for example, it is incorrect to tag a pad and a flip-flop with the same TNM. The only compatible predefined groups are flip-flops and input latches, on which you can use the same TNM.

Place only *one* TNM on each symbol. If you want to assign more than one identifier to the same symbol, include all identifiers on the right side of the equal sign (=) separated by a semicolon (;), as follows:

```
TNM=joe;fred
```

Placing TNMs on Macro Symbols

When a macro contains only one symbol type, you can place a TNM directly on the macro. If the macro contains only flip-flops, all the flip-flips are identified by the given TNM.

When a macro contains more than one symbol type, use the TNM identifier in conjunction with one of the predefined groups: FFS, PADS, or LATCHES as indicated by the following syntax examples:

TNM=FFS:*identifier*
TNM=LATCHES:*identifier*
TNM=PADS:*identifier*

If you want to place an identifier on more than one symbol type, separate each symbol type and identifier with a semicolon (;) as illustrated by the following examples:

```
TNM=FFS:FLOPS_1;PADS:OPADS_6
TNM=RAMS:MEMS;LATCHES:INLATS
```

Placing TNMs on Signals or Pins to Group Flip-Flops

You can easily group flip-flops by flagging a common input signal, typically either a clock signal or an enable signal. If you attach a TNM to a signal or load pin, that TNM applies to all flip-flops and/or input latches that are reached through the signal or pin. The software traces forward on that path, through any number of gates or buffers, until it reaches a flip-flop or input latch and adds that element to the specified TNM group. This mechanism is called forward tracing.

Placing a TNM on a signal is equivalent to placing that TNM attribute on every load pin of the signal. Use pin TNM attributes when you need finer control.

Figure 4-5 illustrates the use of a TNM on a net that traces forward to create a group of flip-flops. The TNM traces forward to the first two flip-flops, which form a group called FLOPS. The bottom flip-flop is not part of the group FLOPS.

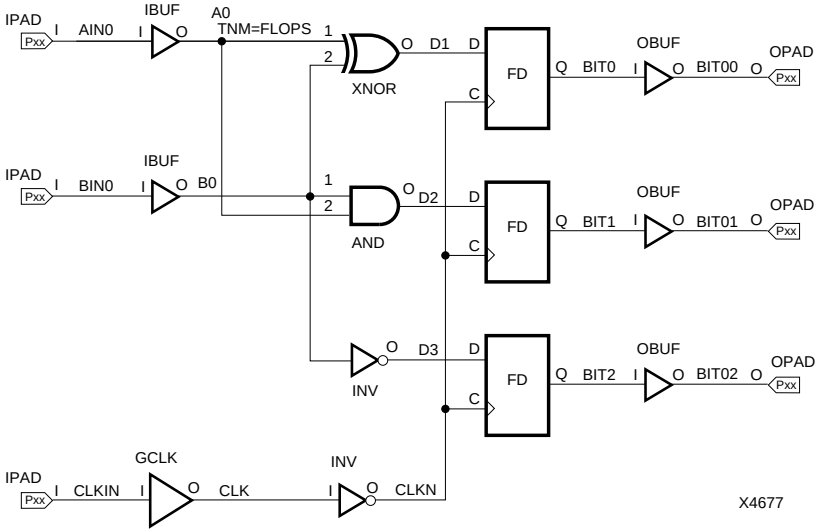


Figure 4-5 Placing a TNM on Signals to Group Flip-Flops

Figure 4-6 illustrates placing a TNM on a clock pin, which traces forward to all three flip-flops and forms the group Q_FLOPS:

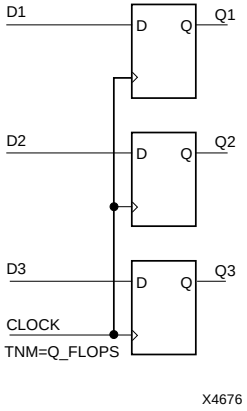


Figure 4-6 Placing a TNM on a Clock Pin to Group Flip-Flops

Creating New Groups from Existing Groups

In addition to naming groups using the TNM identifier, you can also define groups in terms of other groups. You can create a group that is a combination of existing groups by defining a TIMEGRP attribute as follows:

newgroup=*existing_grp1*:*existing_grp2* [:*existing_grp3* . . .]

where *newgroup* is a newly created group that consists of existing groups which were created via TNMs, predefined groups, or other TIMEGRP attributes.

Mentor Graphics Users — You must specify a leading equal sign (=) when defining TIMEGRP attributes, for example, =*newgroup*. The preceding equal sign lets Mentor know that this is a user-defined attribute. Refer to the *Mentor Interface User Guide* for more information.

The TIMEGRP Primitive

TIMEGRP attributes reside in the TIMEGRP primitive, as illustrated in Figure 4-7. After you create a TIMEGRP attribute definition within a TIMEGRP primitive, you can use it in the TIMESPEC primitive. Each TIMEGRP primitive can hold up to eight group definitions and you can use multiple TIMEGRP primitives.

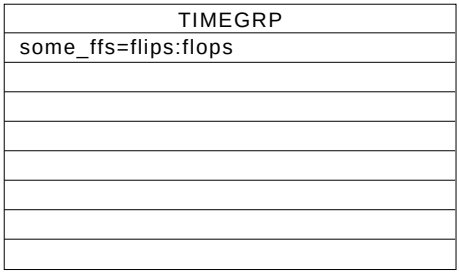


Figure 4-7 TIMEGRP Primitive

You can place TIMEGRP attributes either in the TIMEGRP primitive on the schematic as discussed in this section or in the constraints (CST) file.

You can use TIMEGRP attributes to create groups using the following methods:

- Combining multiple groups into one
- Creating groups by exclusion

The following sections discuss each method in detail.

Combining Multiple Groups into One

You can define a group by combining other groups. The following syntax example illustrates the simple combining of two groups:

```
big_group=small_group:medium_group
```

In this syntax example, small_group and medium_group are existing groups defined using a TNM or TIMEGRP attribute. Within the TIMEGRP primitive, TIMEGRP attributes can be listed in any order; that is, you can create a TIMEGRP attribute that references another TIMEGRP attribute that appears after the initial definition.

Note: A circular definition, as shown below, causes an error.

```
many_ffs=ffs1:ffs2
ffs1=many_ffs:ffs3
```

Creating Groups by Exclusion

You can define a group that includes all elements of one group except the elements that belong to another group, as illustrated by the following syntax examples:

group1=*group2*:**EXCEPT**:*group3*

where *group1* represents the group being defined; *group2* and *group3* can be a valid TNM, predefined group, or TIMEGRP attribute.

As illustrated by the following example, you can specify multiple groups to include or exclude when creating the new group.

group1=*group2*:*group3*:**EXCEPT**:*group4*:*group5*

Note: Do not use reserved words, such as FFS, PADS, RISING, FALLING, or EXCEPT, as group names or TNMs.

Creating Groups by Pattern Matching

You can use wildcard characters to define groups of symbols whose associated signal names match a specific pattern as follows:

group=predefined_group(pattern)

where *predefined_group* can only be one of the following predefined groups — FFS, LATCHES, or PADS. A *pattern* is any string of characters used in conjunction with one or more wildcard characters.

Note: When specifying a signal name, you must use its full hierarchical path name so the software can find the signal in the flattened design.

For flip-flops, and input latches, specify the output signal name. For pads, specify the external signal name.

In XACT-Performance, any place you specify a predefined group, you can specify a predefined group qualified by a pattern as follows:

TS*identifier***=FROM:***group(pattern)***:TO:***group(pattern)***=***delay*

The following example illustrates creating a group that includes the flip-flops that source signals whose names begin with $1I3/FRED:

```
group1=ffs($1I3/FRED*)
```

The following example illustrates a group that excludes certain flip-flops whose output signal names match the specified pattern:

```
group_23=ffs:EXCEPT:ffs(a*)
```

In this example, group_23 includes all flip-flops except those whose output signal names begin with the letter “a.”

Using Wildcards to Specify Signal Names

The wildcard characters “*” and “?”, enable you to select a group of symbols whose output signal names match a specific string or pattern. The asterisk (*) represents any character string. The question mark (?) indicates a single character.

For example, DATA* indicates any signal name that begins with “DATA,” such as DATA1, DATA22, DATABASE, and so on. The string NUMBER? specifies any signal names that begin with “NUMBER” and end with one single character, for example, NUMBER1 and NUMBERS, but not NUMBER12.

You can also specify more than one wildcard character. For example, *OR? specifies any signal names that begin with any series of characters followed by "OR" and end with any one character such as NOR1, OR2, and MULTIPLEXOR5.

Multiple Specifications Applied to the Same Path

When you apply more than one From-To specification to the paths between a given pair of end points, the software analyzes all of them. In this case, the software chooses the fastest one, which might not be the specification you want. For example, suppose you have the following timing requirements:

```
TS_ALL=FROM:PADS:TO:FFS:=30
TS_SLOWER=FROM:PADS:TO:SLOW_FFS=40
```

For TS_SLOWER to control the paths to SLOW_FFS, it would have to disable TS_ALL on these paths; however, no specification can disable another. Therefore, the TS_SLOWER paths are specified faster than intended. You can avoid this problem by either of two methods:

- Make sure your broad specifications are always the slowest ones. You could create a FAST_FFS group instead of a SLOW_FFS group, as follows:

  ```
  TS_ALL=FROM:PADS:TO:FFS=40
  TS_FASTER=FROM:PADS:TO:FAST_FFS=30
  ```

- Explicitly exclude slower paths from faster, more general specifications by using a TIMEGRP EXCEPT statement, as explained in the previous section.

  Using this method, you could create a TIMEGRP attribute to define the FAST_FFS group as a complement of the SLOW_FFS group as follows:

  ```
  FAST_FFS=FFS:EXCEPT:SLOW_FFS
  ```

  In the TIMESPEC primitive, you can define two non-overlapping specifications:

  ```
  TS_FASTER=FROM:PADS:TO:FAST_FFS=30
  TS_SLOWER=FROM:PADS:TO:SLOW_FFS=40
  ```

Ignoring Selected Paths

Timing paths that are never used during time-critical operations do not require path analysis and any timing requirement placed on these paths will unnecessarily restrict the software and may result in failure to meet the overall timing requirements. You can use IGNORE to disable any paths that you do not need to control by using the following syntax within the TIMESPEC primitive:

TS*id***=IGNORE**

Attaching this to a net or load pin causes the timing analysis software to ignore any paths that include the net or load pin and this makes your design easier to route.

Breaking Combinational Loops

The software cannot perform timing path analysis on combinational loops and therefore it automatically ignores certain connections in order to break the loops. By using IGNORE, you can direct the software to ignore specific nets or load pins, consequently controlling how loops are broken.

Specifying Relative TS Attribute Delays

Instead of specifying time or frequency in a TS attribute definition, you can specify timing as a multiple or division of another TS attribute. This is useful in a system where all clocks are derived from a master clock; in this situation, changing the timing specification for the master clock changes the specification for all clocks in the system.

Use the following syntax to specify a relative TS attribute delay:

TS*identifier***=***specification***:***reference_TS_attribute***[*,/]***number*

where *number* can be either a whole number or a decimal. The *specification* can be any From-To statement as illustrated by the following examples:

```
FROM:PADS:TO:PADS
FROM:group1:TO:group2
FROM:tnm_identifier:TO:FFS
FROM:LATCHES:TO:group1
```

Use "*" to represent multiplication and "/" to represent division. The specification type of the reference TS attribute does not need to be the same as the TS attribute being defined. However, it must not be specified in terms of AUTO or IGNORE.

Examples of specifying relative TS attributes are described as follows. In these cases, assume that the reference attributes were specified as delays (not frequencies)

`TS08=FROM:FFS:TO:PADS= TS05*10`	The paths between flip-flops and pads are placed and routed so their delay is at most 10 times the delay specified in the TS05 attribute.
`TS1=FROM:PADS:TO:PADS= TS07/8`	The paths between input and output pads are placed and routed so their delay is at most one-eighth the delay specified in the TS07 attribute.

Note: When a reference attribute is specified as a frequency, a multiple represents a faster specification; a division represents a slower specification.

You can also specify relative TS attributes in terms of other relative TS attributes. The following example provides an illustration.

```
TS09=FROM:FFS:TO:FFS=50
TS10=FROM:FFS:TO:PADS=TS09*2
TS11=FROM:PADS:TO:PADS=TS10*4
```

Specifying Time Delay Units

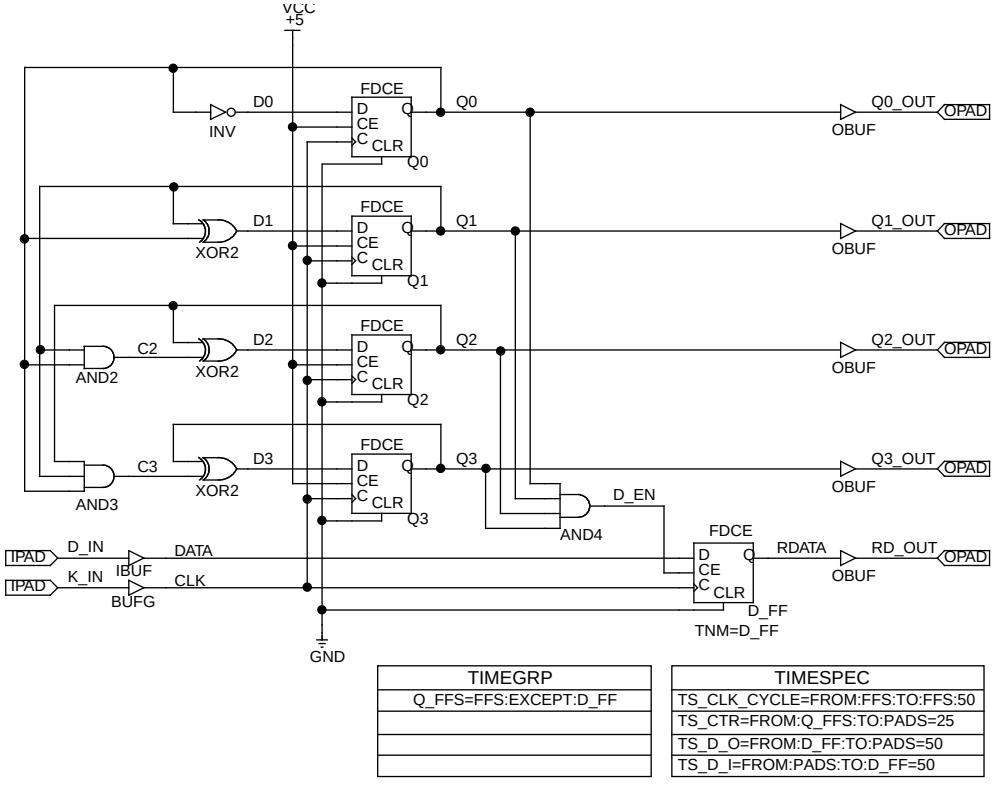
Nanoseconds are the default units for specifying delay times in TS attributes. However, after specifying the maximum delay or minimum frequency numerically, you can enter the unit of measure by specifying the following:

- NS for nanoseconds
- MHZ for megahertz
- US for microseconds
- KHZ for kilohertz

The software converts all units to nanoseconds and rounds them to 0.1 ns accuracy.

Example Schematic Using TNMs and TIMEGRPs

TNM identifiers define symbols or groups of symbols that are used in timing specifications. They can also define other groups. Figure 4-8 shows an example of a TNM attribute attached to an individual symbol. In this circuit, the flip-flop D_FF has the attribute TNM=D_FF attached to it.



TIMEGRP
Q_FFS=FFS:EXCEPT:D_FF

TIMESPEC
TS_CLK_CYCLE=FROM:FFS:TO:FFS:50
TS_CTR=FROM:Q_FFS:TO:PADS=25
TS_D_O=FROM:D_FF:TO:PADS=50
TS_D_I=FROM:PADS:TO:D_FF=50

Figure 4-8 Using TNMs and TIMEGRPs in a Schematic

The TIMEGRP symbol contains an attribute that defines a group of flip-flops called Q_FFS, which includes all flip-flops in the schematic except the one labeled D_FF. You can then use the group Q_FFS to create timing specifications in the TIMESPEC primitive. The flip-flop D_FF has its clock enable driven at 1/16th of the clock frequency; therefore, its flip-flop to pad and pad to flip-flop timing specifications are longer than the flip-flop to pad specifications in the Q_FFS group.

Timing Control Syntax Summary

The following sections summarize the XACT-Performance syntax.

TNM Attributes

The following table lists the syntax used when creating TNMs, which you enter directly on the primitive symbol, macro symbol, signal, or load pin.

Flag Type	TNM Attribute Syntax
Symbol Macro symbol	**TNM**=*group1* [;*group2* . . .] **TNM**=*predefined_group*:*group1* [;*predefined_group*:*group2* . . .]
Signal	**TNM**=*group*
Load pin	**TNM**=*group*

TIMEGRP Attributes

The following lists the syntax used within the TIMEGRP primitive.

Group Type	TIMEGRP Attribute Syntax
Combine	*new_group*=*group1*:*group2* [:*group3* . . .]
Exclude	*new_group*=*group1*: [:*group2* . . .]:**EXCEPT**:*group3*: [:*group4* . . .]

TIMESPEC Attributes

The following lists the syntax used for parameters that define TS attributes, which reside in the TIMESPEC primitive.

Spec Type	TIMESPEC Attribute Syntax	TS Flag Required?
From-To	**TS***id*=**FROM:***group*:**TO:***group*=*delay*	No
Ignore	**TS***id*=**IGNORE**	Yes
Auto	**TS***id*=*path_type*:**AUTO**	Yes

Timing Control using a Timing Constraints File

For behavioral and VHDL designs you can use a Constraints file (.CST) to specify timing requirements. The TIMESPEC and TIMEGRP constraints that apply to EPLD designs are described here.

Note: You can also use a constraints file with schematic designs. The Constraints file will override any T-Specs within the schematic.

Constraints File Syntax Conventions

The following conventions are used for .CST file entries:

- Lower-case words are case sensitive and must be lower case.
- [0-9] means a range of numbers between 0 and 9, inclusive.
- [a-z] means a range of characters between AA and Z, inclusive.
- % means “can be composed of.”
- | indicates alternatives, of which only one must be selected.
- { } means that you must choose one of the enclosed items.
- [ ] means that the enclosed items are optional, or that the enclosed items represent a range of possible values.
- *italics* represent variables.

TIMESPEC Constraint Statement Syntax

Use the following syntax for creating EPLD TIMESPEC statements:

TIMESPEC="*timespec_line*";

where *timespec_line* is composed of the following elements:

- *timespec_line* : := *tsIdentifier=timespec_statement*

  Example: from:FFS:to:FFS=10

- *tsIdentifier* : := TS*label*

  Example: TSmux_23

- *timespec_statement* : := {*default_spec* | *delay_spec* | *link* | ignore}

  Example: from:fred:to:joe:=15

- *tlabel* : := *alphanum* [*alphanum*...]

  Example: mux_3

- *alphanum* : := {[a-z] | [A-Z] | [0-9] | _}

  Example: Test_Mux_57

- *from_to* : := from:*group*:to:*group=max_delay*

  Example: from G_1:to:G_8=15

- *dmax_delay* : := {*requirement* | auto}

  Example: auto

- thi : := *float_number*

  Example: 5.5

- *group* : := {*tlabel* | *pattern* | *whole_class*}

  Example: mux_3

- *whole_class* : := {ffs | pads | latches}

  Example: pads

- *max_delay* : := {*requirement* | auto}

  Example: max_delay : := 10.1 mhz

- *pattern* : := *whole_class* (*signame* {:*signame*})

  Example: input*
- *signame* : := {*alphanum* | * | ?} [{*alphanum* | * | ?} ...]

  Example: signame : := input*
- *requirement* : := {*float_number* [*unit*] | *reference*}

  Example: requirement : := 1.5 us
- *unit* : := {ns | us | mhz | khz}

  Example: unit : := mhz
- *reference* : := {*tsIdentifier operator float_number}*

  Example: reference : := TSa_5 * 10.3
- *operator* : := {* | /} (***note:*** the * is a multiplier, not a wildcard)

  Example operator : := /
- *float_number* : := *number* [.*number*]

  Example: 45

TIMEGRP Constraint Statement Syntax

Use the following syntax for creating EPLD TIMEGRP statements:

`TIMEGRP="`*`timegrp_line`*`";`

where *timegrp_line* is composed of the following elements, in addition to those described in the previous section:

- *timegrp_line* : := *tlabel=derived_group*

  Example: mux_3=Test_mux_45:except:ff_23
- *derived_group* : := *compound* | *difference*

  Example: mux_3:Test_mux_45
- *compound* : := *group* [:*group*:*group* ...]

  Example: mux_3:Test_mux_45
- *difference* : := *compound*:except:*compound*

  Example: Test_mux_45:except:Reg_4

Creating a Constraints File

You can create a constraints file by using an ascii text editor. Save the file as *file_name*.CST in your design directory. An example of a constraints file is shown as follows:

```
TIMESPEC=”TS02=FROM:FF_1:TO:FF_4=20”;
TIMESPEC=”TS08=FROM:FFS:TO:PADS=TS05*10”;
TIMESPEC=”TS59=FROM:G_1A:TO:G_8C=15”;
TIMESPEC=”TS62=FROM:LATCHES:TO:PADS=25”;
TIMESPEC=”TS65=FROM:PADS:TO:PADS=18”;
TIMESPEC=”TS73=FROM:FFS(a*):TO:FFS(b*)=20”;
TIMEGRP=”many_ffs=ffs1:ffs2”;
TIMEGRP=”group1=ffs(ctr*)”;
TIMEGRP=”group2=group1:except:ffs(ctr0)”;
TIMEGRP=”ff1=FFS(a*)”;
```

Figure 4-9 Example .CST File

Chapter 5

Timing Analysis

After you have fitted your design, you can use the Timing Analyzer to find and report the timing of paths through the device. You can select specific paths or general types of paths to examine and you can choose from several report formats.

If you included time specification attributes in your design (also called TimeSpecs or T-Specs), you can use the Timing Analyzer to determine whether the XEPLD fitter was able to meet these specifications.

Timing Analysis Procedure

The typical procedure for using the Timing Analyzer is as follows:

1. Make sure the design you want to analyze is the current design in the Design Manager, then open the Timing Analyzer.
2. Choose the kind of report you want from the Analyzer menu.
3. Use the report window commands to view the report in more detail, save the report, and print the report.
4. (Optional) If having all the paths in your design reported is more information than you need, select filters from the Path Filters menu to determine which types of paths will be analyzed and reported. You can "filter out" paths you are not interested in. Then repeat step 2.
5. (Optional) Select options from the Options menu to fine-tune the analysis even more. Then repeat step 2.

These steps are described in more detail in the following sections.

Opening the Timing Analyzer

Open the Timing Analyzer by double clicking on the Timing Analysis icon in the Tools subwindow of the main Design Manager window. Figure 5-1 shows the icon.

Figure 5-1 Timing Analyzer Icon

The Timing Analysis window appears. The design that is loaded into the Timing Analyzer is the current design established in the Design Manager. To load a different design into the Timing Analyzer, exit to the Design Manager and load a different design from there.

Timing Analysis Window Features

The Timing Analysis window is shown in Figure 5-2.

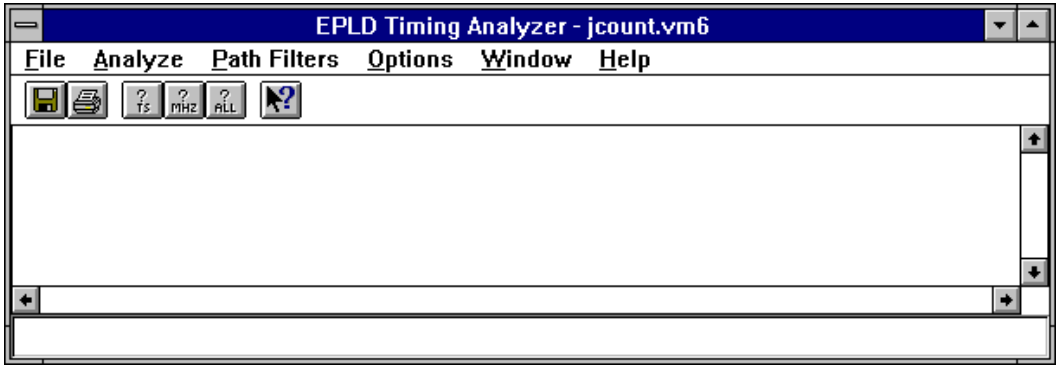


Figure 5-2 Timing Analysis Window

This Window controls the display of report windows generated by commands on the Analyze menu. Because the report windows can be moved and sized independently of the main Timing Analyzer window, the Cascade and Tile commands arrange report windows below and to the right of the main window rather than within it.

However, when you collapse report windows, their icons appear in the main Timing Analyzer window.

Generating Reports

To generate a report, select one of the commands on the Analyze menu; each of these commands generates a different timing report. If you have previously chosen path filters or other options, the report reflects these choices. After a report is generated, it appears in a separate window from which it can be saved and printed.

Double click on an icon in the main window to expand the report window. The selected report can be saved, printed, deleted, or closed. You can also save and print from the main window file menu by selecting the icon.

Standard Report Header

All Timing Analyzer reports have a header that summarizes the path filters and options that affect the report. Here is an example header for a Performance Summary report:

```
                Performance Summary Report
                --------------------------

Design:     jcount
Device:     XC7336-5PC44
Program:    Timing Report Generator Version 6.0
Date:       Mon Apr 17 13:54:42 1995

Timing Analysis Options:

From: CLR CE C $1N5.Q.FFB.REG.Q $1N8.Q.FFB.REG.Q $1N7.Q.FFB.REG.Q
      $1N6.Q.FFB.REG.Q
  To: Q0 Q1 Q2 Q3 $1N5.Q.FFB.D $1N16 $1N5.Q.FFB.RSTF $1N8.Q.FFB.D
      $1N8.Q.FFB.RSTF $1N7.Q.FFB.D $1N7.Q.FFB.RSTF $1N6.Q.FFB.D $1N6.Q.FFB.RSTF

Speed Grade:                                -5

Selected Path Types:

External
--------
Pad to Pad -                                Pad to pad combinatorial propagation
                                            delay. Paths through PRE/CLR register
                                            inputs are not reported.



Clock Pad to Output Pad -                   Clock pad(global or product term
                                            clock) to output pad propagation
                                            delay.
```

```
Setup to Clock at Pad -                 Setup time of data at PAD to clock at
                                        PAD. This parameter is only reported
                                        for global clocks and product term
                                        clocks driven directly from input
                                        pads

Internal
--------
Clock to Setup -                        Register to register cycle time.

Pad to Setup -                          Data path delay from PAD to register
                                        data input. Includes register tSU.

Clock to Pad -                          Delay from register clock input to
                                        output pad.

Path Ending at Clock Pin of FFs -       Clock path delay from clock
                                        pad(global or product term) to
                                        register clock input.

Other Options:
Maximum Number of Paths:                1000
Maximum Number of Paths per TimeSpec:   1000
Report Delays Less Than:                1000.0ns
```

Performance to Timespecs Report

In the Performance to Timespecs report, timing is analyzed according to how well it matches the T-Specs you included in your design. Make sure the Report Paths Failing TimeSpec command on the Path Filters menu is ON. The format of this report is exactly the same as that of the Detailed Path Report, except that only paths failing to meet the time specifications assigned to them are listed.

Note: The Performance to Timespecs report shows the source and destination for all paths with T-Specs; the filter for sources and destinations does not apply.

Performance Summary Report

In the Performance Summary report, general indexes of overall performance are reported, for example worst-case pad-to-pad delay, pad-to-setup, clock-to-pad, clock-to-setup, and maximum clock speed. The Select Path Types command partially determines the contents of this report. Here is an example report without the header:

```
Performance Summary:

Clock net 'C' path delays:

Worst case clock pad to output pad delay  :          4.5ns (1 macrocell levels)
(Includes an external input margin of 0.0ns.)
(Includes an external output margin of 0.0ns.)
Clock Pad 'C' to Output Pad 'Q0'                                    (Fast Clock)

Clock to Setup path delay                 :          5.0ns (1 macrocell levels)
Clock to Q, net '$1N5.Q.FFB.REG.Q' to DFF Setup(D) at '$1N5.Q.FFB.D' (Fast Clock)
Target FF drives output net '$1N5.Q'

Clock to Pad path delay                   :          3.0ns (1 macrocell levels)
(Includes an external output margin of 0.0ns.)
Clock to Q, net '$1N5.Q.FFB.REG.Q' to Pad 'Q0'                      (Fast Clock)

Pad to Setup path delay                   :          8.5ns (0 macrocell levels)
(Includes an external input margin of 0.0ns.)
Pad 'CE' to DFF Setup(D) at '$1N5.Q.FFB.D'                          (Fast Clock)
Target FF drives output net '$1N5.Q'

Pad Ending at Clock Pin path delay        :          1.5ns (0 macrocell levels)
(Includes an external input margin of 0.0ns.)
Clock Pad 'C' to DFF Clock Pin at '$1N16'                           (Fast Clock)

Setup to Clock at the Pad                 :          7.0ns (0 macrocell levels)
Data signal 'CE' to DFF D input Pin at '$1N5.Q.FFB.D'
Clock pad 'C' to DFF Clock Pin at '$1N16'                           (Fast Clock)

                        Minimum Clock Period: 5.0ns
                    Maximum Internal Clock Speed: 200.0Mhz
                           (Limited by Cycle Time)

                Estimated Maximum External Clock Speed: 142.8Mhz
                     (Limited by Setup to Clock at the Pad)

--------------------------------------------------------------------------------
                      Clock Pad to Output Pad Delays(nsec)

\ Clock     C
  \
    \
      \
        \
          \
            \
Output \------

Q0         4.5
Q1         4.5
```

```
Q2        4.5
Q3        4.5

------------------------------------------------------------------------------
                       Setup to Clock at Pad Delays(nsec)

\ Clock              C
 \
  \
   \
    \
     \
      \
       \
        \
         \
          \
           \
            \
             \
 Data         \------

$1N5.Q.FFB.REG.Q   3.5
$1N6.Q.FFB.REG.Q   3.5
$1N7.Q.FFB.REG.Q   3.5
$1N8.Q.FFB.REG.Q   3.5
C                  5.0
CE                 7.0

------------------------------------------------------------------------------
                       Register to Register Delays(nsec)
                                  (Clock: C)

\ From               $     $     $     $
 \                   1     1     1     1
  \                  N     N     N     N
   \                 5     6     7     8
    \                .     .     .     .
     \               Q     Q     Q     Q
      \              .     .     .     .
       \             F     F     F     F
        \            F     F     F     F
         \           B     B     B     B
          \          .     .     .     .
           \         R     R     R     R
            \        E     E     E     E
             \       G     G     G     G
              \      .     .     .     .
               \     Q     Q     Q     Q
  To            \------------------------

$1N5.Q.FFB.D       5.0                 5.0
$1N6.Q.FFB.D       5.0   5.0
$1N7.Q.FFB.D             5.0   5.0
$1N8.Q.FFB.D                   5.0   5.0
```

Detailed Path Report

In the Detailed Path report, the timing of every path in the design is analyzed in detail, with a breakdown of each segment in each path. The Sort On field of the Report Options command determines the order in which paths are listed.

This report tends to be long, even for fairly simple designs. You may want to use path filters, described in the "Applying Path Filters" section of this chapter, to eliminate report paths not of interest to you.

Here is an example report that has been edited for brevity, without the header:

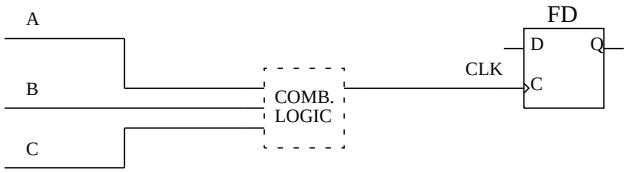
```
Report Only Longest Paths between Points: True

Output will be sorted by increasing path delays.
Logical Path                  Delay Type               Delay        Cumulative
------------                  ----------               -----        ----------
From: C                       -                        :  0.0ns        (0.0ns)
Thru: C/BUF0.OUT              MARGIN                   :  0.0ns        (0.0ns)
  To: $1N16                   FCLK BUFFER + FB.DFF.FCLK:  1.5ns        (1.5ns)

Source clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: $1N5.Q                  FFB.DFF.tCO              :  1.0ns        (1.0ns)
Thru: Q0/OLDNODE              OUTPUT BUFFER            :  2.0ns        (3.0ns)
  To: Q0                      MARGIN                   :  0.0ns        (3.0ns)

Source clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: $1N6.Q                  FFB.DFF.tCO              :  1.0ns        (1.0ns)
Thru: Q1/OLDNODE              OUTPUT BUFFER            :  2.0ns        (3.0ns)
  To: Q1                      MARGIN                   :  0.0ns        (3.0ns)

Source clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: $1N7.Q                  FFB.DFF.tCO              :  1.0ns        (1.0ns)
Thru: Q2/OLDNODE              OUTPUT BUFFER            :  2.0ns        (3.0ns)
  To: Q2                      MARGIN                   :  0.0ns        (3.0ns)

Source clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: $1N8.Q                  FFB.DFF.tCO              :  1.0ns        (1.0ns)
Thru: Q3/OLDNODE              OUTPUT BUFFER            :  2.0ns        (3.0ns)
  To: Q3                      MARGIN                   :  0.0ns        (3.0ns)

...

From: C                       -                        :  0.0ns        (0.0ns)
Thru: C/BUF0.OUT              MARGIN                   :  0.0ns        (0.0ns)
Thru: $1N16                   FCLK BUFFER              :  1.5ns        (1.5ns)
Thru: $1N5.Q                  FFB                      :  1.0ns        (2.5ns)
Thru: Q0/OLDNODE              OUTPUT BUFFER            :  2.0ns        (4.5ns)
  To: Q0                      MARGIN                   :  0.0ns        (4.5ns)
```

```
Source clock net: $1N16
Worst case clock delay from origin C is 1.5ns
Destination clock net: $1N16
Worst case clock delay from origin C is 1.5ns
Source and destination clock skew: 0.0ns
From: $1N5.Q.FFB.FBK          -                      :  1.0ns        (1.0ns)
  To: $1N5.Q.FFB.D            FFB.DFF.D              :  4.0ns        (5.0ns)

...

Destination clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: C                       -                      :  0.0ns        (0.0ns)
Thru: C/BUF0.OUT              MARGIN                 :  0.0ns        (0.0ns)
Thru: $1N16                   FCLK BUFFER            :  1.5ns        (1.5ns)
Thru: $1N5.Q.FFB.FBK          FFB                    :  1.0ns        (2.5ns)
  To: $1N6.Q.FFB.D            FFB.DFF.D              :  4.0ns        (6.5ns)

...

Destination clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: C                       -                      :  0.0ns        (0.0ns)
Thru: C/BUF0.OUT              MARGIN                 :  0.0ns        (0.0ns)
Thru: $1N16                   FCLK BUFFER            :  1.5ns        (1.5ns)
Thru: $1N7.Q.FFB.FBK          FFB                    :  1.0ns        (2.5ns)
  To: $1N7.Q.FFB.D            FFB.DFF.D              :  4.0ns        (6.5ns)

...

From: CLR                     -                      :  0.0ns        (0.0ns)
Thru: CLR/BUF0.OUT            MARGIN                 :  0.0ns        (0.0ns)
Thru: $1N12                   INPUT BUFFER           :  1.5ns        (1.5ns)
  To: $1N6.Q.FFB.RSTF         UIM + FFB.DFF.S        :  6.5ns        (8.0ns)

Destination clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: CE                      -                      :  0.0ns        (0.0ns)
Thru: CE/BUF0.OUT             MARGIN                 :  0.0ns        (0.0ns)
Thru: $1N20                   INPUT BUFFER           :  1.5ns        (1.5ns)
  To: $1N6.Q.FFB.D            UIM + FFB.DFF.D        :  7.0ns        (8.5ns)

...

Destination clock net: $1N16
Worst case clock delay from origin C is 1.5ns
From: CE                      -                      :  0.0ns        (0.0ns)
Thru: CE/BUF0.OUT             MARGIN                 :  0.0ns        (0.0ns)
Thru: $1N20                   INPUT BUFFER           :  1.5ns        (1.5ns)
  To: $1N8.Q.FFB.D            UIM + FFB.DFF.D        :  7.0ns        (8.5ns)
```

Check for Asynchronous Logic

The Check for Asynchronous Logic command lists locations of signals that are possibly asynchronous and the product-term clock signals that control them. Two types of paths are considered possibly asynchronous:

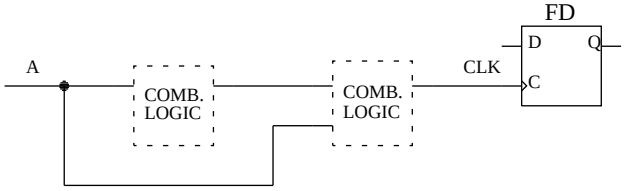
- A clock pin with multiple origins, as in the following example:



In this case, the following statement appears in the report:

```
clock pin CLK has multiple origins A, B, C:
possible asynchronous.
```

- A clock pin with multiple paths leading to it, as in the following example:



In this case, the following statement appears in the report:

```
clock pin CLK has multiple paths to its origin
A: possible asynchronous.
```

Show Clocks

The Show Clocks command lists the clock signals in the design.

Show Settings

The Show Settings command lists the Path Filter and Options settings you chose during this session of the Timing Analyzer.

Report Window Commands

After you select any command on the Analyze menu, the requested report appears in a new window. For example, here is a Performance Summary report window.

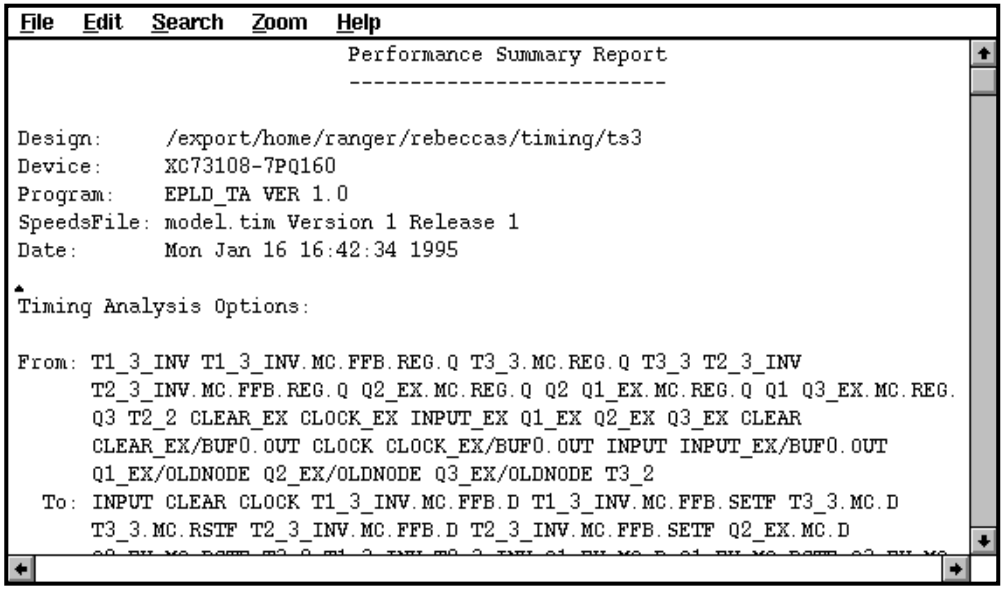


Figure 5-3 Report Window

The Window menu in the main Timing Analyzer window controls the display of report windows generated by commands on the Analyze menu. Because the report windows can be moved and sized independently of the main Timing Analyzer window, the Cascade and Tile commands arrange report windows below and to the right of the main window rather than within it.

However, when you collapse report windows, their icons appear in the main Timing Analyzer window.

Collapsing the main window collapses the report windows as well. Restoring the main window also restores the report windows.

Each report window has its own set of menu commands, which allow you to view the report in more detail, save the report, and print the report. The report window commands are as follows:

- File-Save As — Opens a file browser window and lets you save the report file.
- File-Print — Sends the report to the default printer. No dialog box is presented.
- Edit-Copy — Saves the selected text to the clipboard so you can copy it into another text file or document.
- Edit-Select All — Allows you to select all the text in the report at once. To copy the entire report, use this command followed by the Copy command.
- Edit-Word Wrap — When this option is on (the default), lines too long to fit in the window wrap to the next line. When this option is off, lines do not wrap regardless of length. Turning word wrap off is useful for viewing tables.
- Search-Find — Presents a dialog box that allows you to search for text. You can specify case sensitivity or whole word only matching, and you can search forward or backward.
- Search-Find Next — Searches for the text you specified last time you used the Find command.
- Zoom-Zoom In — Expands the text size and shows a smaller area of the report.
- Zoom-Zoom Out — Shrinks the text size and shows a larger area of the report.
- Zoom-Zoom Normal — Changes the text to its original size.

To close the report window, select the File-Exit command.

Applying Path Filters (Optional)

The Path Filters menu offers the following choices, which enable you to eliminate paths from the report and examine only the kinds of paths of greatest interest to you. By default, if you do not use any filters, all paths are included in whichever report you choose.

The Options menu commands, which are described in the "Selecting Additional Options" section of this chapter, also refine the contents of the reports you select.

Report Paths Failing Timespec

Use the Report Paths Failing TimeSpec command to view timing information for signals that do not meet your timing specifications. This command includes in the report paths that are slower than the time specification you assigned.

When this command is on, a check appears to the left of the command in the menu. The default setting is off.

Report Paths with No Timespec

Use the Report Paths with No TimeSpec command to view timing information for all signals, including paths having no assigned time specification.

When this command is on, a check appears to the left of the command in the menu. The default setting is On.

Ignore Timespecs

The Ignore TimeSpecs command deletes selected T-Specs from the report. Use this command to filter out signals that are not timing-critical. By default, all T-Specs are included.

When you select this command, the Ignore TimeSpec window appears as shown in Figure 5-4.

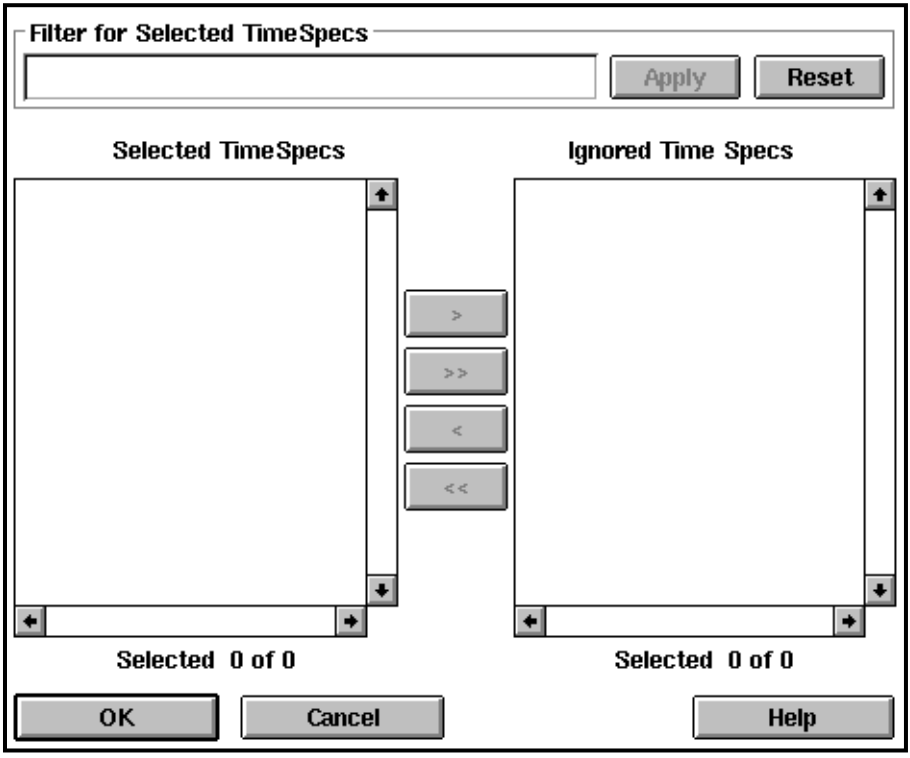


Figure 5-4 Ignore TimeSpec Window

Follow these steps to delete specific T-Specs from the report:

1. Type a name, partial name, or name with a wildcard in the Filter for Selected Timespecs field to select a subset of the T-Specs in the design. The ? is a single-character wildcard; the * is a multiple-

character wildcard. Select the Apply button to apply the filter. To select all T-Specs, leave this field blank or select the Reset button followed by the >> button. This filtering step is optional.

Note: The Reset button clears the current filter and includes all the T-Specs in the Selected TimeSpec list except the T-Specs you have already moved to the Ignored TimeSpec list.

2. To move a T-Specs into the Ignored TimeSpec list, select the T-Specs from the Selected TimeSpec list and select the > button. To move a all T-Specs into the Ignored TimeSpec list, use the >> button.

3. To move a T-Specs back into the Selected TimeSpec list (so it is NOT ignored), select the T-Specs from the Ignored TimeSpec list and select the < button. To move all T-Specs back into the Selected TimeSpec list, use the << button.

Note: After you move T-Specs to the Ignored TimeSpec list, the T-Specs remain highlighted so you can undo the move with a single mouse click on the < button.

4. When your Selected TimeSpec list and Ignored TimeSpec list are organized as you want them, select the OK button.

Select Sources and Select Destinations

The Select Sources and Select Destinations commands include paths having the sources and destinations you specify in the report. These sources and destinations can be nets, flip-flops, pads, clocks, or macrocells. The following figure shows the source and destination of a typical pad to setup path.

Figure 5-5 Source and Destination

By default, all pads and registers are included as sources and destinations. You can add sources or destinations to the included list or select sources or destinations you want to be removed from the included list.

When you select this command, the Select Sources or Select Destinations window appears. These two windows have exactly the same functions; the only difference is that Select Sources lists design elements that can be beginnings of paths and Select Destinations lists design elements that can be ends of paths.

Note: The Performance to Timespecs report shows the source and destination for all paths with T-Specs; the filter for sources and destinations does not apply.

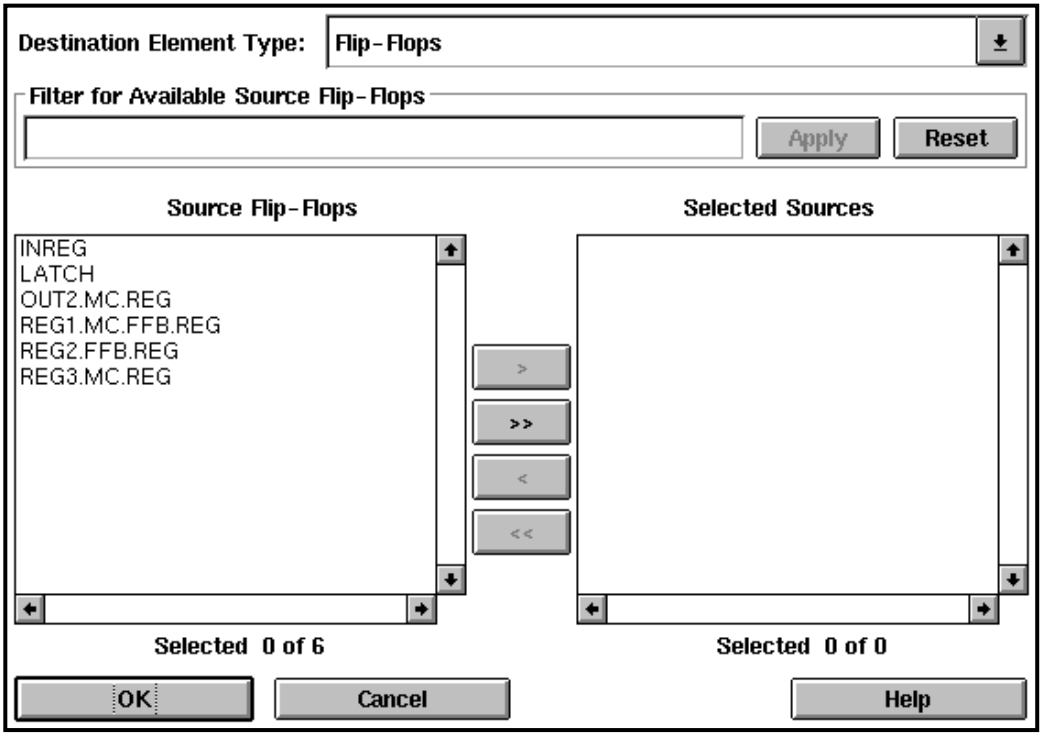


Figure 5-6 Select Sources Window

Follow these steps:

1. Select the type of source or destination element you are interested in from the Source Element Type or Destination Element Type field at the top of the window. Choices are All, Flip-flops, Pads, Clocks, Nets, or Macrocells. The All option, which is the default, means all pads and flip-flops, not all elements.

Note: If you select All, this text appears in the Source Elements or Destination Elements list: `*** All Sources***` or `***All Destinations***`. If you add this text to the Selected Sources or Selected Destinations list using the > button, you cannot add additional elements to the Selected Sources or Selected Destinations list. To add other types of elements, first select the All text in the Selected Sources or Selected Destinations list and click on the < button, then add the elements you want.

2. Type a name, partial name, or name with a wildcard in the Filter for Available Elements field to select a subset of the elements in the design to be excluded. The ? is a single-character wildcard; the * is a multiple-character wildcard. Select the Apply button to apply the filter. To select all elements, leave this field blank or select the Reset button followed by the >> button.

Note: The Reset button clears the current filter and includes all the elements in the Available Elements list except the elements you have already moved to the Source Elements or Destination Elements list.

3. The Source Elements or Destination Elements list contains the elements that will be included in the report. To remove an element from this list, select it and select the < button. The element moves to the list on the left.

   The > button moves an element from the list on the left back to the Source Elements or Destination Elements list. The >> and << buttons move all elements in one list to the other list.

Note: After you move elements to the Source Elements or Destination Elements list, the elements remain highlighted so you can undo the move with a single mouse click on the < button.

4. Select OK when your sources or destinations list is finished.

Select Path Types

The Select Path Types command specifies the types of paths to include in the report, for example, “clock to setup.” By default, all path types are listed.

When you select this command, the Select Path Type window appears.

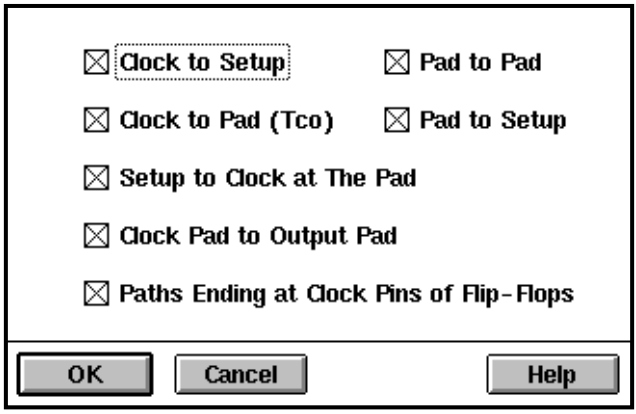


Figure 5-7 Select Path Type Window

If you wish to eliminate some path types from the report, click on them to remove the X’s from their boxes.

The path types are defined as follows:

- Clock to Setup — Register to register cycle time, including clock to output and setup delay.

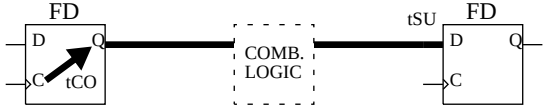


Figure 5-8 Clock to Setup Path

- Pad to Pad — Combinational pad to pad delay.

Figure 5-9 Pad to Pad Path

- Clock to Pad — Delay from the register clock input to the output pad.

Figure 5-10 Clock to Pad Path

- Pad to Setup — Data path delay from the pad to the register data input. Includes the register setup time.

Figure 5-11 Pad to Setup Path

- Setup to Clock at the Pad — Setup time of data at the pad-to-clock at the pad. This path type includes only global clocks and product term clocks driven directly from input pads. If the data input is signal A and the clock input is signal CLK, the timing calculation for this type of path is as follows:

  Max(A to D) – Min(CLK to C)

  Max and Min are maximum and minimum propagation delays through the combinational logic.

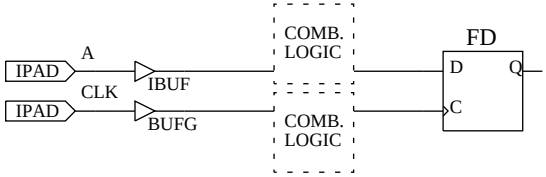


Figure 5-12 Setup to Clock at the Pad Path

- Clock Pad to Output Pad — Clock pad to output pad propagation delay. The clock can be a global or product term clock.

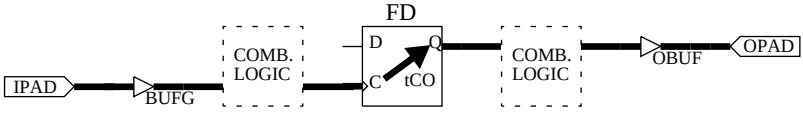


Figure 5-13 Clock Pad to Output Pad Path

- Paths Ending at Clock Pins of Flip-Flops — Delay from clock pad to register clock input. The clock can be a global or product term clock.

Figure 5-14 Path Ending at the Clock Pin of a Flip-Flop

Include or Exclude Paths with Nets

The Include and Exclude Paths with Nets commands are not opposites and can be used together.

The Include Paths with Nets command includes in the report only the paths that pass through at least one of the nets you specify; all other paths are excluded. By default all nets are included, so all paths are included in the report.

The Exclude Paths with Nets command excludes from the report the paths that pass through at least one of the nets you specify. Paths not

passing through any of the selected nets are not necessarily included, however. By default no nets are excluded.

You can use both of these commands together. First use Include Paths with Nets to select the set of paths you are interested in, then use Exclude Paths with Nets to refine this set of paths. A path that passes through both an included net and an excluded net is excluded.

When you select one of these commands, the Include Paths with Nets or Exclude Paths with Nets window appears. These two windows have the same layout of lists and buttons. You select paths to include or exclude in the same way; the only difference between these two commands is the results.

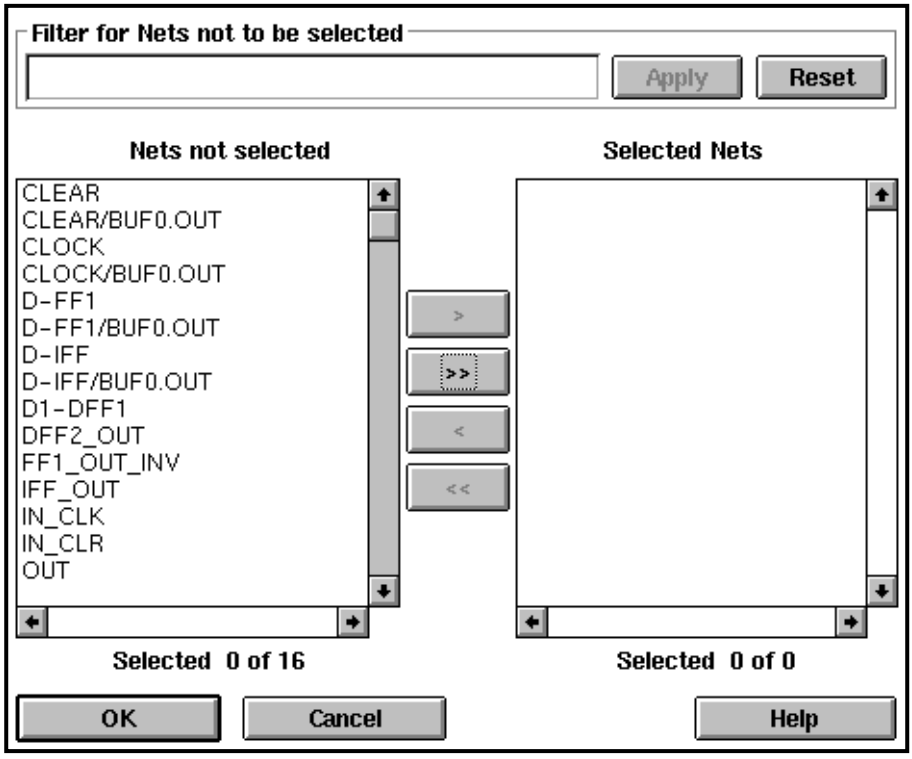


Figure 5-15 Include Path with Nets Window

Follow these steps:

1. Type a name, partial name, or name with a wildcard in the Filter for Nets not to be selected field to select a subset of the nets in the design. The ? is a single-character wildcard; the * is a multiple-character wildcard. Select the Apply button to apply the filter. To select all nets, leave this field blank or select the Reset button followed by the > button.

Note: The Reset button clears the current filter and includes all the elements in the Nets not selected list except the elements you have already moved to the Selected Nets list.

2. Select nets from the Nets not selected list and move them to the Selected Nets list by selecting the > button.

   To move a net back to the Nets not selected list, use the < button. The >> and << buttons move all nets in a list to the other list.

Note: After you move elements to the Nets not selected or Selected Nets list, the elements remain highlighted so you can undo the move with a single mouse click on the < button.

3. Select OK when your list of nets is as you want it.

Break Logic Loops

The Break Logic Loops command specifies feedback loops to be broken in the report. When you select this command, the Break Logic Loops window appears.

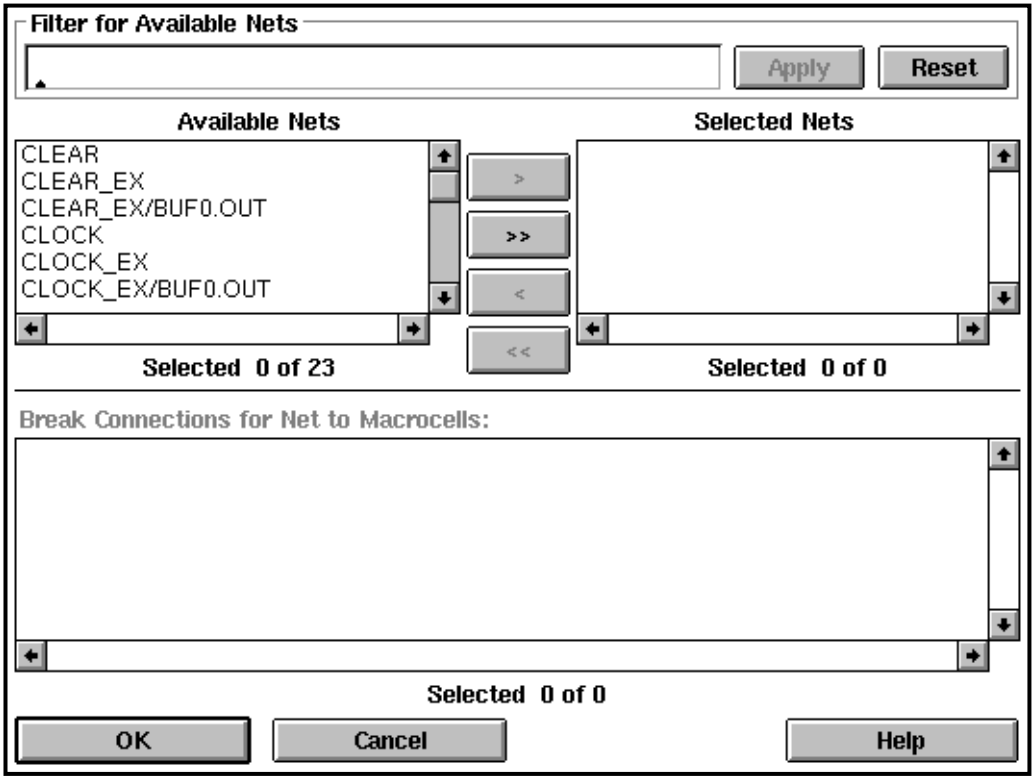


Figure 5-16 Break Logic Loops Window

Follow these steps:

1. Type a name, partial name, or name with a wildcard in the Filter for Available Nets field to select a subset of the nets in the design. The ? is a single-character wildcard; the * is a multiple-character wildcard. Select the Apply button to apply the filter. To select all nets, leave this field blank or select the Reset button followed by the > button.

Note: The Reset button clears the current filter and includes all the elements in the Nets not selected list except the elements you have already moved to the Selected Nets list.

2. Select nets from the Available Nets list and move them to the Selected Nets list by selecting the > button.

   To move a net back to the Nets not selected list, use the < button. The >> and << buttons move all nets in a list to the other list.

Note: After you move elements to the Nets not selected or Selected Nets list, the elements remain highlighted so you can undo the move with a single mouse click on the < button.

3. After you select a net, the macrocells fed by that net appear in the Break Connections for Net to Macrocells list with check marks on the left. Select the macrocell(s) to which the net does not feed back to remove the checks. If you remove all macrocells, the net is removed from the Selected Nets list.
4. Select another net and repeat the previous step.
5. Select OK when your list of nets and macrocells indicating feedback loops is as you want it.

Note: To specify a feedback path for a net, you must specify (leave checked) only the macrocell(s) to which the net feeds back. If you specify all macrocells fed by the net, all load pins of that net are broken, including those that are not feedback paths. It is a good idea to display your design file in another window so you can see the feedback paths and macrocells clearly.

For example, in the following figure, A is a signal that feeds back to Macrocell 1. If you break all the macrocells fed by signal A, the load pin of A that goes to Macrocell 2 is broken in addition to the signal that goes to Macrocell 1.

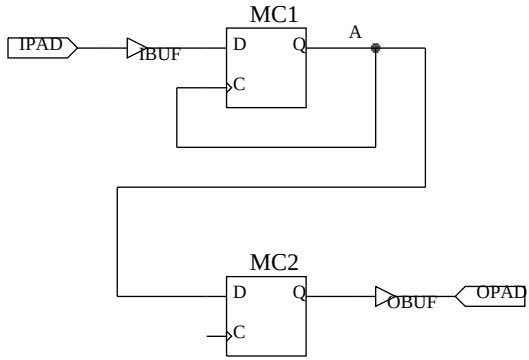


Figure 5-17 Break Logic Loops Example

Reset Path Filters

The Reset Path Filters command restores default path filters. In general, all paths are included in the report by default. More specifically, default filters are as follows:

- Report Paths Failing TimeSpec is OFF.
- Report Paths with No TimeSpec is OFF.
- Ignore TimeSpecs has no T-Specs selected, which means all paths are included.
- Select Sources includes all pads and flip flops.
- Select Destinations includes all pads and flip flops.
- Select Path Types includes all path types.
- Include Paths with Nets includes all nets.
- Exclude Paths with Nets includes no nets.
- Break Logic Loops includes no feedback paths, which means all feedback paths are included.

Selecting Additional Options (Optional)

The commands on the Options menu allow you to fine-tune your analysis.

Set Speed Grade

The Set Speed Grade command re-evaluates the timing based on a selected speed grade. When you select this command, the Set Speed Grade window appears.

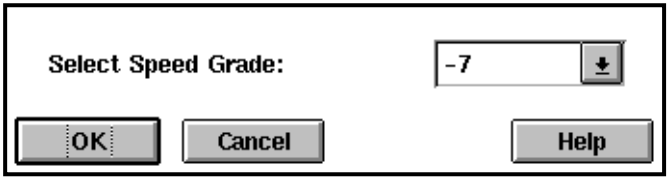


Figure 5-18 Set Speed Grade Window

The speed grades you can select are different for each XC7000 device.

Set Delay Margins for I/O

The Set Delay Margins for I/O command adds the specified delays to pads to simulate delays due to lines on the PC board.

When you select this command, the Set Delay Margins for I/O window appears.

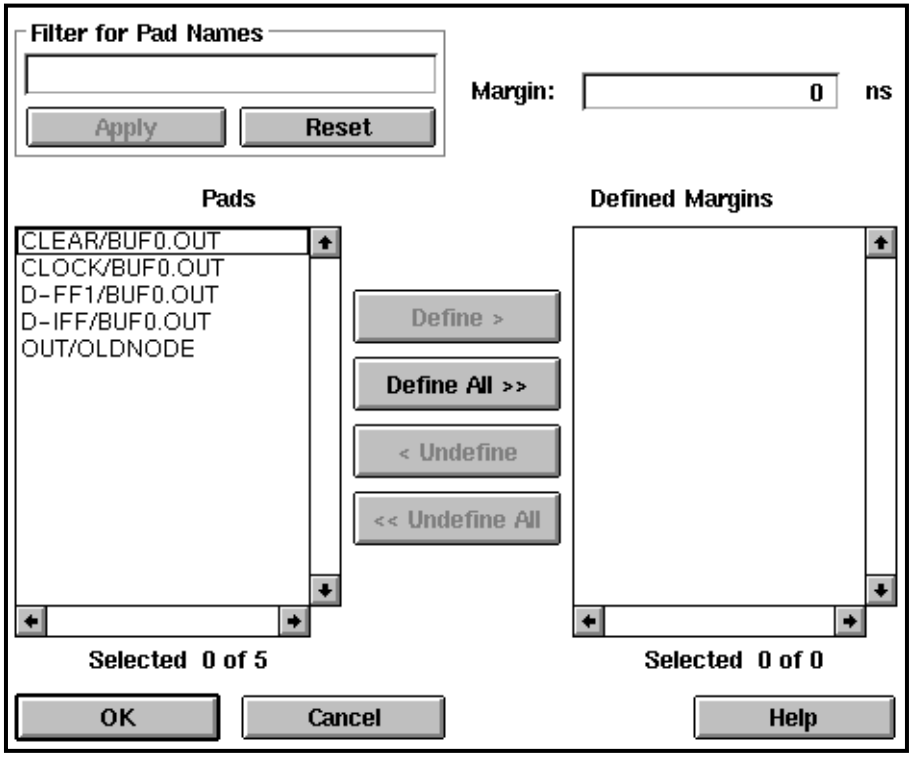


Figure 5-19 Set Delay Margins for I/O Window

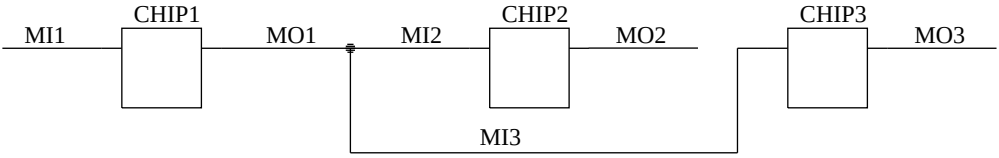
Follow these steps:

1. Type a name, partial name, or name with a wildcard in the Filter for Pad Names field to select a subset of the pads in the design. The ? is a single-character wildcard; the * is a multiple-character wildcard. Select the Apply button to apply the filter. To select all pads, leave this field blank or select the Reset button.
2. In the Margin field, type the number of nanoseconds additional delay you wish to add to the selected pads. If you enter an invalid value such as a negative number, an error message is displayed.
3. Select pads from the Pads list and transfer them to the Defined Margins list by selecting the Define > button. The delay you specified in step 2 applies to the pads in the Defined Margins list.

To remove a margin from a pad, select the pad in the Defined Margins list and then select the < Undefine button. The Define All >> and << Undefine All buttons transfer all pads in one list to the other list.

4. To apply a different delay to a different set of pads, repeat steps 1 through 3 with a different value in the Margin field.
5. Select OK when you have finished assigning delays to pads.

The following figure shows input, output, and interchip delays.



Interchip delays:

CHIP1 to CHIP2 = MO1 + MI2
CHIP1 to CHIP3 = MO1 + MI3

Figure 5-20 Margin Delays

Report Options

The Report Options command gives you additional filters for report information. When you select this command, the Report Options window appears, with the default values shown. Blank fields indicate that an infinite number are allowed. If you enter an invalid value such as a negative number, a warning is displayed.

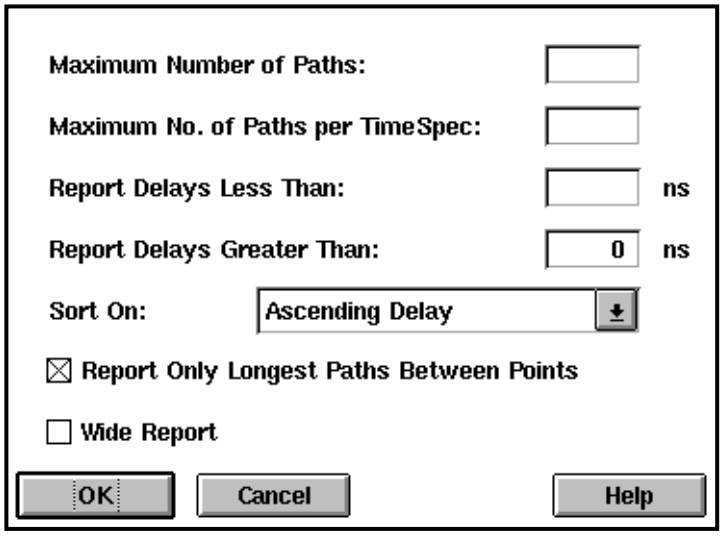


Figure 5-21 Report Options Window

The options are:

- *Maximum Number of Paths* — The maximum number of paths that will appear in the report.
- *Maximum No. of Paths per TimeSpec* — The maximum number of paths that will be reported for a specified T-Spec.
- *Report Delays Less Than* — Only those delays that are shorter than this specification are reported.
- *Report Delays Greater Than* — Only those delays that are longer than this specification are reported.
- *Sort On* — Determines how paths are listed in the report as follows:
  - Ascending Delay — Lists paths in order of their delay, with the shortest delay first and the longest last.
  - Descending Delay — Lists paths in order of their delay, with the longest delay first and the shortest last.

- Source Net — Lists paths alphabetically by the net at which they start.
- Destination Net — Lists paths alphabetically by the net at which they end.
- Source Clock Net — Lists paths alphabetically by the clock net at which they start.
- Destination Clock Net — Lists paths alphabetically by the clock net at which they end.

Chapter 6

Report Formats

The XEPLD software creates a variety of report files in the design directory that provide you with information about the state of your design. The reports described in this chapter are as follows:

- Resource report (*design_name*.res).
- Mapping report (*design_name*.map).
- Pin Out report (*design_name*.pin).
- Fitting report (*design_name*.par).
- Timing Report (*design_name*.tim).
- Timespec Report (*design_name*.tsp).
- PLUSASM Assembler Log report (*pld_name*.lga).
- EQN file (*design_name*.eqn).
- Log file (*design_name*.log).

The report formats are described in the following sections.

Viewing Reports

You can use the Report Browser to view most reports. You can view those reports which are not displayed in the Report Browser with any ASCII text editor.

To use the Report Browser, follow these steps:

1. Select the Report Browser command from the Utilities menu. The Report Browser window appears as shown in Figure 6-1.

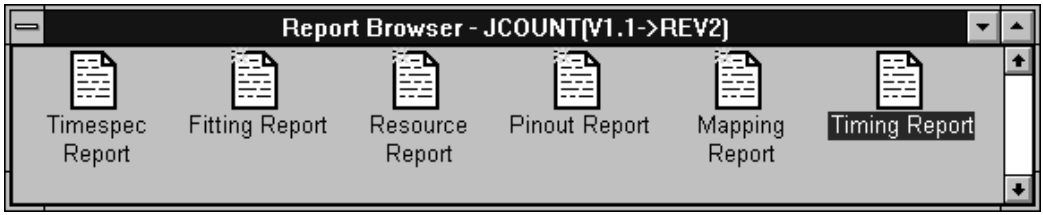


Figure 6-1 Report Browser

The yellow stars on the report icons indicate that the reports have not yet been read. After you open the report, the yellow star 5disappears.

2. Double click on the icon for the report of your choice. The report appears in its own window. Figure 6-2 shows an example report window.

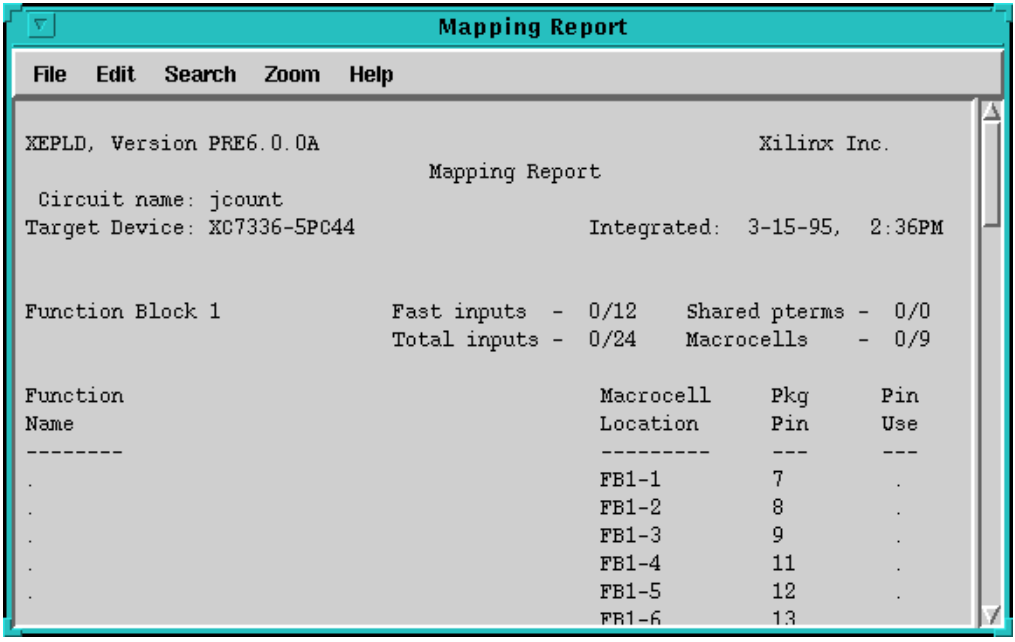


Figure 6-2 Example Report Window

3. Use the commands in the report window to examine your report. The commands are:
   - File-Save As — Opens a file browser window and lets you save the report file.
   - File-Print — Sends the report to the default printer. No dialog box is presented.
   - Edit-Copy — Saves the selected text to the clipboard so you can copy it into another text file or document.
   - Edit-Select All — Allows you to select all the text in the report at once. To copy the entire report, use this command followed by the Copy command.
   - Edit-Word Wrap — When this option is on (the default), lines too long to fit in the window wrap to the next line. When this option is off, lines do not wrap regardless of length. Turning word wrap off is useful for viewing tables.
   - Search-Find — Presents a dialog box that allows you to search for text. You can specify case sensitivity or whole word only matching, and you can search forward or backward.
   - Search-Find Next — Searches for the text you specified last time you used the Find command.
   - Zoom-Zoom In — Expands the text size and shows a smaller area of the report.
   - Zoom-Zoom Out — Shrinks the text size and shows a larger area of the report.
   - Zoom-Zoom Normal — Changes the text to its original size.
4. To close the report window, select the File-Exit command.

The Resource Report

The Resource report (*design_name*.res) is the first report you should examine to determine the results of the fitting process. This report lists the used and remaining numbers of Function Blocks, Macrocells, and pins of each type.

Logic Resources

The Logic Resources section shows the number of used and remaining Function Blocks and Macrocells.

Required Pin Resources

The Required column of the Pin Resources section shows the number of input, output (excluding global control/output), I/O, FastClk, FOE, and CE pins required by the design.

Used Pin Resources

The Used columns show how many of each type of physical device pin are occupied to satisfy the I/O signals in your design; note that output-only or input-only signals in your design may occupy I/O pins on the device.

Remaining Pin Resources

The Remaining columns show how many pins are still available on the device.

Fast Inputs and Outputs

The Fast Input/Fast Output section lists how many fast inputs and fast outputs the design requires and how many of each are available on the device.

Example Resource Report

The following is a resource report example.

```
XEPLD, Version 5.0                                          Xilinx Inc.
                              Resource Report
 Circuit name: UARTPALC
Target Device: XC7354-10PC68                Integrated: 11-10-93,  1:37PM

LOGIC RESOURCES

                  Used       Remaining
Function Blocks     4          2
Macrocells         30         24
```

```
PIN RESOURCES:

Type    Required  --------Used-------------  --------Remaining---------
                  I  O  I/O Fclk Foe Cen Tot  I  O  I/O Fclk Foe Cen Tot
Inputs    3       3        0                 3  5       34                39
Outputs  12          0     8    0   2   2  12     0    34    2   0   0  36
I/Os      0          0                      0          34                34
Fclks     1                    1            1                2            2
Foes      0                        0        0                    0        0
Cens      0                            0    0                        0    0
         ---     --- --   ---  --- --- --- ---
          16       3  0    8    1   2   2  16

Note:The design requires 0 pins with Fast Input capability.
     This device has 11 pins with Fast Input capability.
     The design requires 0 pins with Fast Output capability.
     This device has 0 FO and 15 I/FO remaining from original 0 FO and 18
I/FO.

 End of Resource Report
```

The Mapping Report

The Mapping report (*design_name*.map) lists the contents of all the macrocells and Function Blocks in the target device. For each Function Block in the device, a header area lists the total number of FastInputs, total inputs, shared product terms, and macrocells used by the design, and the total number available for the Function Block. Beneath each header is a list indicating how each macrocell in the Function Block is used. The list contains the following columns and sections.

Function Name

This column shows the names of the signals in the design. For a behavioral design, the function name is the signal name.

For a schematic design, the function name is in the following format:

instance_name:*pin_name*

The instance name is the internal component name. The pin name is the name of the signal that the component pin drives.

Macrocell Location

This column shows the Function Block and macrocell addresses where the component output pins or equation output signals are placed. The format of this column is: FB *function_block – macrocell*. For example, FB4-3 signifies Function Block 4, macrocell 3.

If you have specified low power on some macrocells, an L will appear on these macrocells in this report.

If a macrocell is using the macrocell carry from the previous macrocell, a C will appear on the macrocell in this report.

Pkg Pin

This column shows the pin numbers of the device output or I/O pins that can be driven by the macrocells. A dot in this column signifies that the macrocell is buried and cannot drive a device pin.

Pin Use

This column shows how the corresponding device pins are used by the design (I, O, I/O, or dot). An "I" indicates that the pin is not used for macrocell output but is used by an input port in the design to receive some other type of signal.

Power Estimation (7300 Family Only)

At the end of this report is an estimate of the power consumption for low power and high power settings.

- The MCHP value is the number of used macrocells in used Function Blocks not designated as low power.
- The MCLP value is the number of used macrocells designated as low power plus the number of unused macrocells in used function blocks.
- Macrocells in unused Function Blocks do not consume power; therefore they are not counted in the power estimation calculation.

For more details about the power estimation calculation, see the Xilinx EPLD data sheets.

Example Mapping Report

The following is a mapping report example.

```
XEPLD, Version 5.0                                          Xilinx Inc.
                              Mapping Report
 Circuit name: UARTPALC
Target Device: XC7354-10PC68                 Integrated: 11- 5-93, 12:23PM

Function Block 1            Fast inputs  -  0/12    Shared pterms -  0/0
                            Total inputs - 10/24    Macrocells    -  3/9

Function                                    Macrocell     Pkg      Pin
Name                                        Location      Pin      Use
--------                                    ---------     ---      ---
C2                                          FB1-1         4         .
START                                       FB1-2         12        .
PAR                                         FB1-3         13        .
.                                           FB1-4         15        .
.                                           FB1-5         17        .
.                                           FB1-6         19        .
.                                           FB1-7         21        .
.                                           FB1-8         22        .
.                                           FB1-9         23        .
...
Function Block 6            Fast inputs  -  0/3     Shared pterms -  1/12
                            UIM inputs   - 15/21    Macrocells    -  9/9

Function                                    Macrocell     Pkg      Pin
Name                                        Location      Pin      Use
--------                                    ---------     ---      ---
BITCLK                                      FB6-1         27        .
D4                                          FB6-2         28        .
D3                                          FB6-3         8         I
DOUT2                                       FB6-4         9         O
BYTECLK                                     FB6-5         10        .
D2                                          FB6-6         29        .
C0                                          FB6-7         6         .
C1                                          FB6-8         24        .
C5                                          FB6-9         26        .

Power estimation parameters: MCLP =   6, MCHP =  30.
MCLP represents macrocells in low power mode and
MCHP represents macrocells in high power mode.
Macrocells in unused function blocks are not considered.
MCLP and MCHP are used to estimate chip power consumption.
Consult the proper data sheets for this calculation.

 End of Mapping Report
```

The Pinout Report

The Pinout report (*design_name*.pin) lists all pins on the target device, how they are used, and the names of their associated signals.

Pkg Pin

This column shows all the pin numbers of the target EPLD device, in numerical order, or in alphanumeric order in the case of BPGA/PGA packages.

Pin Type

This column shows whether the pin is physically input-only (I), output-only (O), bidirectional (I/O), power (VCC, VSS), or special-purpose (MR, CLK, CEN, FOE).

Pin Use

This column shows how the pin is used in your design (I, O, I/O, and so on; see "Pin Use Legend"). For unused input and I/O pins, this column lists "tie" to remind you to tie the unused inputs to ground on your board. "NC" is shown for unused output-only pins indicating the pin should be left unconnected. If you select the "drive unused I/O pins" implementation option (in the default template), an (O) appears in this column whenever an instance is mapped to the macrocell but the output is not used.

Pin Name

This column shows the name of the external signal mapped to the pin.

Pin Use Legend

This section lists the codes for the types of pins shown in the Pin Use column.

Example Pinout Report

The following is a pinout report example.

```
XEPLD, Version 5.0                                          Xilinx Inc.
                                Pin-List Report
 Circuit name: UARTPALC
Target Device: XC7354-10PC68                Integrated: 11- 5-93, 12:23PM

Pkg  Pin   Pin    Pin
Pin  Type  Use    Name
---  ----  ---    ----
1    MR
2     I    tie    (unused)
3     I    tie    (unused)
4    I/O   tie    (unused)
5     I    tie    (unused)
6    I/O   tie    (unused)
7    VSS
8    CLK    I     X4CLK
9    CLK    O     DOUT2
10   CLK   tie    (unused)
...
58   I/O   tie    (unused)
59   VCC
60   CEN    O     DOUT1
61   CEN    O     DOUT0
62   FOE    O     FRAMING
63   VCC
64   FOE    O     PARITY
65    I     I     RD
66   I/O   tie    (unused)
67    I     I     CS
68    I    tie    (unused)

Pin Use Legend:

I     - input
O     - output
I/O   - input/output
I-L   - input uses latch
I-R   - input uses register
I/O-L - input/output uses latch
I/O-R - input/output uses register
NC    - not connected/not available
tie   - unused pin must be tied to VCC or GND
(O)   - unused pin attached to used macrocell

 End of Pin-List Report
```

The Fitting Report

The Fitting report (*design_name*.par) shows the allocation of Function Block resources. This report helps you to understand how the resources available on the device were filled by the equations in your design. It can provide useful information for optimizing or modifying your design.

The Fitting report lists all partitions created to implement the design, and for each partition lists the outputs and Function Block resources. It also shows the input signals used by each partition and each output. Use this report when trying to increase the design density or when modifying a design with a frozen pinout.

Summary

The summary contains information that applies to all partitions.

Part Name

Partition names are Function Block names. If the number of partitions in the design exceeds the number of Function Blocks in the device, the extra partitions will be named OVERFLOW_*n*, starting with OVERFLOW_0.

If you see extra partitions even though your design has fewer partitions than the device, it means some of the partitions in your design cannot use the available function blocks. For example, a partition that requires output pins cannot use a Function Block that has no pins.

Number of Outputs

This column shows the number of macrocells used in the partition.

Number of Input Lines Used

This column shows the combined number of pin and UIM inputs going into the partition. Fast Function Blocks allow up to 24 inputs. High Density Function Blocks allow up to 21 inputs.

Signal Inputs

This column shows the number of inputs that would be necessary if no UIM ANDing were performed.

Number of Shared Pt

This column shows the combined number of shared product terms used in the partition. High Density Function Blocks have 12 shared product terms available. Fast Function Blocks have none.

O/IO Used

This column shows output pins and I/O pins used by each partition.

O/IO Avail

This column shows the total number of output pins and I/O pins available to each partition. The partitioner only counts usable outputs after considering pin assignment and control pin assignment. If you use a FastCLK signal as a control input, you cannot use the attached macrocell for an output pin. If you assign an input signal to an I/O pin, you cannot use the attached macrocell for an output pin.

Size Factor

This column estimates the Function Block resources used (from 0 to 9) based on the number of inputs, outputs, and shared product terms.

Inputs Used by Each Partition

This cross-reference table identifies the specific inputs feeding into each partition. Indexing numbers (top and bottom) for the inputs are listed at the end of the report. The Xs are inputs entering the function via an input line; the @s are ANDed UIM inputs.

Partition Listing

There is one partition listing for each partition in the design. Each listing contains information that applies to one partition only.

Signals Used

This list displays all the pin or UIM inputs to the partition.

Anded UIM Inputs Used

This list displays all UIM equations that are inputs to the partition.

Inputs Used by Each Output Table

This cross reference table identifies the specific inputs used by each output macrocell in the equation file. The indexing numbers for inputs are defined in the input name list at the end of the report.

MC No

This column shows the macrocell number of the Function Block.

Output Name

This column identifies the output signal name mapped to the corresponding macrocell.

Pin Req

This column indicates that the output requires no pin (-), an output pin (O), or an I/O pin (I/O).

Pin Avl

This column indicates that the pin available to the output is no pin (-), an output pin (O), an I/O pin (I/O), a FastCLK pin (FCLK), or a Fast Output Enable pin (FOE).

Sh Pt

This column indicates the combined number of shared D1 and D2 product terms used by each output (whether or not any of the product terms are actually shared with any other output).

Input Listing

The names of all UIM inputs in the design appear in this list. At the end are all the UIM equations. The numbers correspond to the columns of the “Inputs Used” tables.

Example Fitting Report

The following is a fitting report example.

```
XEPLD, Version 6.0                                          Xilinx Inc.
                          Partitioning Report
 Circuit name: UARTPALC
Target Device: XC7354-10PC68                  Integrated: 11- 5-93, 12:23PM

Part        #  of    # of Input  Signal   # of        O/IO    O/IO   Size
Name        Outputs  Lines Used  Inputs   Shared Pt   Req     Avail  Factor
FB1           3         10         10        0         0/0     0/9     4
FB2           0          0          0        0         0/0     0/9     0
FB3           9         19         19        0         6/0     4/3     9
FB4           9         15         15        0         5/0     0/6     9
FB5           0          0          0        0         0/0     0/9     0
FB6           9         15         15        1         1/0     2/6     9
             ---                                       ---     ---    ---
             30                                       12/0     6/42   31

Part                            Inputs Used by Each Partition
Name
                               |----+----1----+----2----+----3
FB1                             X.......X.......XXXXXXXX..
FB2                             ..........................
FB3                             X.XXXXXXXX.....XXXXXXXXX.
FB4                             ..XX....X...XXXXXXXXX.X.XX
FB5                             ..........................
FB6                             ..XX...XX.XXX...XXXXXXX.X.
                               |----+----1----+----2----+----3

+++++++++++++++++++++++++++++++  FB1  +++++++++++++++++++++++++++++++++
                            Signals Used:
  1: SDIN              19: C2                22: C5
  9: BITCLK            20: C3                23: START
 17: C0                21: C4                24: PAR
 18: C1

MC Output          Pin  Pin  Sh
No Name            Req  Avl  Pt
                               |----+----1----+----2----+----3
1  C2              -    I/O  0  ................XXX...X...
2  START           -    I/O  0  X..............XXXXXXX...
3  PAR             -    I/O  0  X.......X.............XX..
4  Unused               I/O     ..........................
5  Unused               I/O     ..........................
6  Unused               I/O     ..........................
7  Unused               I/O     ..........................
8  Unused               I/O     ..........................
9  Unused               I/O     ..........................
                               |----+----1----+----2----+----3
...
```

```
+++++++++++++++++++++++++++++++++  FB6  +++++++++++++++++++++++++++++++++
                            Signals Used:
  3: RD                 12: D2                20: C3
  4: CS                 13: D3                21: C4
  8: READY              17: C0                22: C5
  9: BITCLK             18: C1                23: START
 11: D1                 19: C2                25: BYTECLK

MC Output         Pin  Pin  Sh
No Name           Req  Avl  Pt
                              |----+----1----+----2----+----3
1  BITCLK         -    I/O  0  .................XX....X...
2  D4             -    I/O  0  ........X...X.............
3  D3             -    FCLK 0  ........X..X..............
4  DOUT2          O    FCLK 0  ..XX.......X............X.
5  BYTECLK        -    FCLK 0  .......X........XXXXXX....
6  D2             -    I/O  0  ........X.X...............
7  C0             -    I/O  0  .................X.....X...
8  C1             -    I/O  0  .................XX....X...
9  C5             -    I/O  1  .................XXXXXXX...
                              |----+----1----+----2----+----3

All outputs placed in a partition.

In   Input Name
No

  1   SDIN
  2   X4CLK
  3   RD
  4   CS
  5   FRAMING
  6   PARITY
  7   OVERUN
  8   READY
  9   BITCLK
 10   D0
...
 16   D6
 17   C0
...
 22   C5
 23   START
 24   PAR
 25   BYTECLK
 26   D7

 End of Partitioning Report
```

The Timing Report

Use this report (*design_name*.tim) to verify your design timing. This report shows the calculated worst-case timing based on the physical implementation of your design. General indexes of overall performance are reported; for example, worst-case pad-to-pad delay, pad to setup, clock to pad, clock to setup, and maximum clock speed. This report is the same as the Performance Summary report generated by the EPLD timing analyzer.

Example Timing Report

The following is a timing report example.

```
                Performance Summary Report
                --------------------------

Design:     /export/home/ranger/rebeccas/timing/allpaths
Device:     XC73144-7PQ160
Program:    EPLD_TA VER 1.0
SpeedsFile: model.tim Version 1 Release 1
Date:       Thu Feb  2 13:48:32 1995

Timing Analysis Options:

From: All
  To: All

Speed Grade                                 : -7
Selected Path Types                         : Clock to Setup
                                              Pad to Pad
                                              Clock to Pad
                                              Pad to Setup
                                              Setup to Clock at Pad
                                              Clock Pad to Output Pad
                                              Path Ending at Clock Pin of FFs

Other Options:
Maximum Number of Paths                     : 1000
Maximum Number of Paths per TimeSpec        : 0
Report Delays Less Than                     : 100.0ns
Report Delays Greater Than                  : 0.0ns

Performance Summary:

Worst case Pad to Pad path delay            : 31.8ns (4 macrocell levels)
(Includes an external input margin of 0.0ns.)
```

```
(Includes an external output margin of 0.0ns.)
Pad 'CAR2' to Pad 'OUT3.OUT.OUT'

Worst case clock pad to output pad delay    : 23.0ns (2 macrocell levels)
(Includes an external input margin of 0.0ns.)
(Includes an external output margin of 0.0ns.)
Clock Pad 'PCLK' to Output Pad 'OUT2.OUT.OUT'

Clock net 'PCLK' path delays:

Clock to Setup path delay                   : 12.0ns (1 macrocell levels)
Clock to Q, net 'REG3.MC.REG.Q' to DFF Setup(D) at 'OUT2.MC.ALU.F'
Target FF drives output net 'OUT2.MC.Q'

Clock to Pad path delay                     : 5.5ns (1 macrocell levels)
(Includes an external output margin of 0.0ns.)
Clock to Q, net 'OUT2.MC.REG.Q' to Pad 'OUT2.OUT.OUT'

Pad to Setup path delay                     : 29.5ns (2 macrocell levels)
(Includes an external input margin of 0.0ns.)
Pad 'PCLK' to DFF Setup(D) at 'OUT2.MC.ALU.F'
Target FF drives output net 'OUT2.MC.Q'



Pad Ending at Clock Pin path delay          : 17.5ns (1 macrocell levels)
(Includes an external input margin of 0.0ns.)
Clock Pad 'PCLK' to DFF Clock Pin at 'OUT2.MC.CLKF'

Setup to Clock at the Pad                   : 28.5ns (2 macrocell levels)
Data signal 'PCLK' to DFF D input Pin at 'OUT2.MC.ALU.F'
Clock pad 'PCLK_PIN' to DFF clock Pin at 'OUT2.MC.CLKF'

                          Minimum Clock Period: 29.5ns

                     Estimated Maximum Clock Speed: 33.9Mhz

Clock net 'FCLK1' path delays:

Clock to Setup path delay                   : 10.0ns (1 macrocell levels)
Clock to Q, net 'REG1.MC.FFB.REG.Q' to DFF Setup(D) at 'OUT1.MC.FFB.D'
Target FF drives output net 'OUT1.MC.Q'

Clock to Pad path delay                     : 13.5ns (1 macrocell levels)
(Includes an external output margin of 0.0ns.)
Clock to Q, net 'LATCH' to Pad 'INREG_OUT.OUT.OUT'

Pad to Setup path delay                     : 11.5ns (0 macrocell levels)
(Includes an external input margin of 0.0ns.)
```

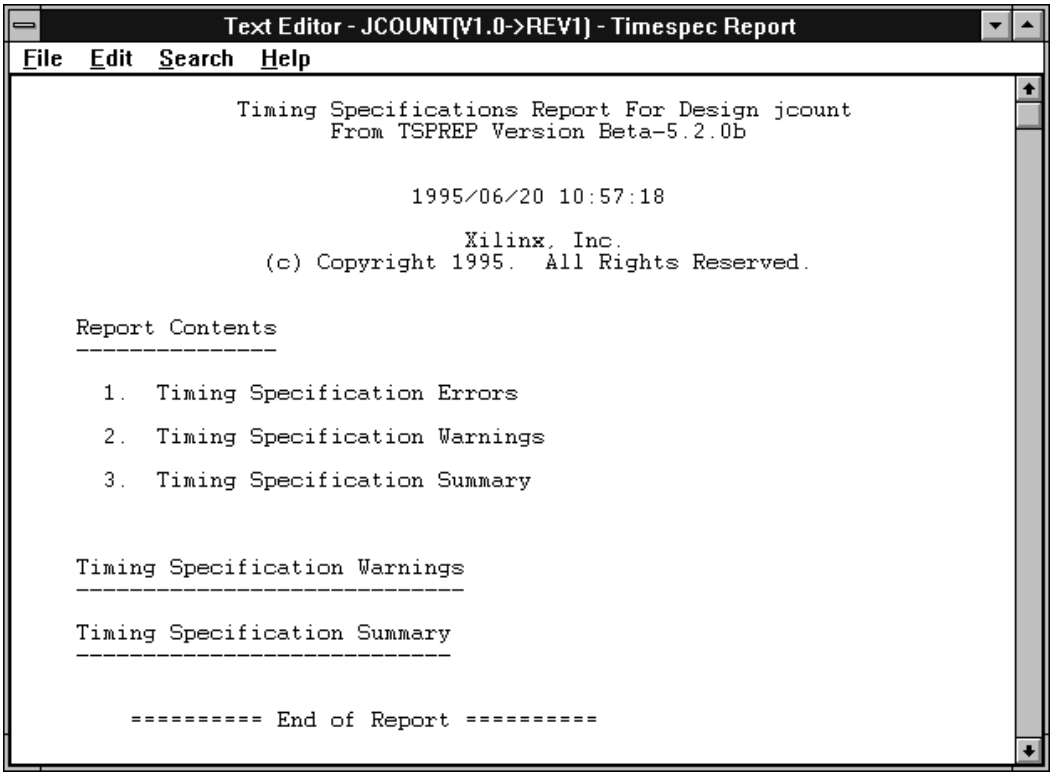
```
Pad 'HIN1' to DFF Setup(D) at 'OUT1.MC.FFB.D'
Target FF drives output net 'OUT1.MC.Q'

Pad Ending at Clock Pin path delay          : 1.5ns (0 macrocell levels)
(Includes an external input margin of 0.0ns.)
Clock Pad 'FCLK1' to DFF Clock Pin at 'REG1_CLK'

Setup to Clock at the Pad                   : 10.0ns (0 macrocell levels)
Data signal 'HIN1' to DFF D input Pin at 'OUT1.MC.FFB.D'
Clock pad 'FCLK1/BUF0.OUT' to DFF clock Pin at 'REG1_CLK'

                         Minimum Clock Period: 13.5ns

                    Estimated Maximum Clock Speed: 74.1Mhz
```

The Timespec Report

The Timespec report (*design_name*.tsp) tells you if you have made any errors in assigning TSPEC attributes in your design and lists the T-Specs that the fitter used.

Example Timespec Report

The following is a timespec report example. Vertical spacing has been adjusted. This example design ran without TSPEC errors, so there are no error or warning sections.



The EQN File

The Equation File (*design_name*.EQN) describes how your design was implemented on the EPLD device.

This file shows the following results of fitting the device:

- Where everything is placed in UIM functions
- FastCLK assignment
- FOE assignment
- Input register assignment
- Logic optimization results
- Logic minimizer results

- Equations assigned to a Fast Function Block with active-low polarity
- Export equations (Fast Function Blocks)
- Split equations

For information about how to use this report, see the *XEPLD Design Guide*.

Example Equation File

The following is an equation file example.

```
;;-------------------------------------------------------------------;;
; This is the .eqn file produced by the partitioner.  It shows     ;
; how your equations were implemented in order to best utilize the ;
; resources available on the chip.  Note that certain equations    ;
; are now active low so that they are eligible to be placed in      ;
; a fast function block.                                           ;
;                                                                  ;
; This design was compiled for the 733644PC                        ;
;;-------------------------------------------------------------------;;

PATTERN jcount.eqn

DATE  Wed Aug 02 09:30:21 1995

CHIP jcount XEPLD

MINIMIZE OFF
LOGIC_OPT OFF
OPTIONS OFF UIM_OPT REG_OPT CLOCK_OPT FOE_OPT

PARTITION FB3_1 Q0
PARTITION FB3_3 Q2 Q3
PARTITION FB3_9 Q1

INPUTPIN CLR CE
          PIN 28 42

OUTPUTPIN  Q0 Q1 Q2 Q3
          PIN 06 40 04 03

FASTCLOCK C
          PIN 05
```

```
EQUATIONS
 Q0 :=  CE */Q3
       + /CE * Q0    ; Fast function block
   Q0.CLKF =  C
   Q0.RSTF =  CLR

 Q1 :=  CE * Q0
       + /CE * Q1    ; Fast function block
   Q1.CLKF =  C
   Q1.RSTF =  CLR

 Q2 :=  CE * Q1
       + /CE * Q2    ; Fast function block
   Q2.CLKF =  C
   Q2.RSTF =  CLR

 Q3 :=  CE * Q2
       + /CE * Q3    ; Fast function block
   Q3.CLKF =  C
   Q3.RSTF =  CLR
```

Design Directories and Files

This appendix describes the flow of files through the Design Manager when a design is processed and the directories and files that comprise a Xilinx design.

File Flow

Figure A-1 shows the types of design files the Design Manager can accept and the types of simulation and programming files the Design Manager can produce. You can create, process, and then simulate a design using Viewlogic, Viewlogic Synthesis, OrCAD, Mentor Graphics, Synopsys, Xilinx ABEL, or PLUSASM.

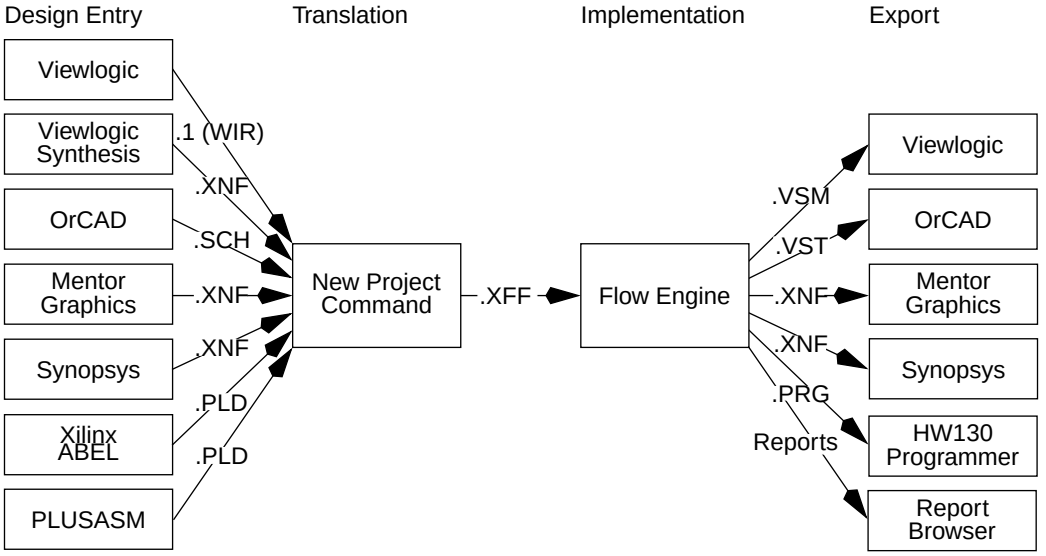


Figure A-1 Input and Output Files

For example, if you created a schematic in OrCAD named mydesign.sch, translating the design would produce a file called mydesign.xff, and implementing the design would produce a simulation file called mydesign.vst and a programming file called mydesign.prg. You would also see report files such as mydesign.pin (the Pinout report) and mydesign.map (the Mapping Report).

The xproject Directory Tree

When you use the New Project command to bring a new design into the Design Manager, the Design Manager creates a directory called xproject. By default, the xproject directory is below the directory that contains your design file.

All the design files that the Design Manager creates are in the xproject directory tree. Figure A-2 shows the structure of the xproject directory and the files contained in its subdirectories. Directories are in boldface; files are in plain text. The v*n_n*, and rev*n* directories represent multiple design versions and revisions.

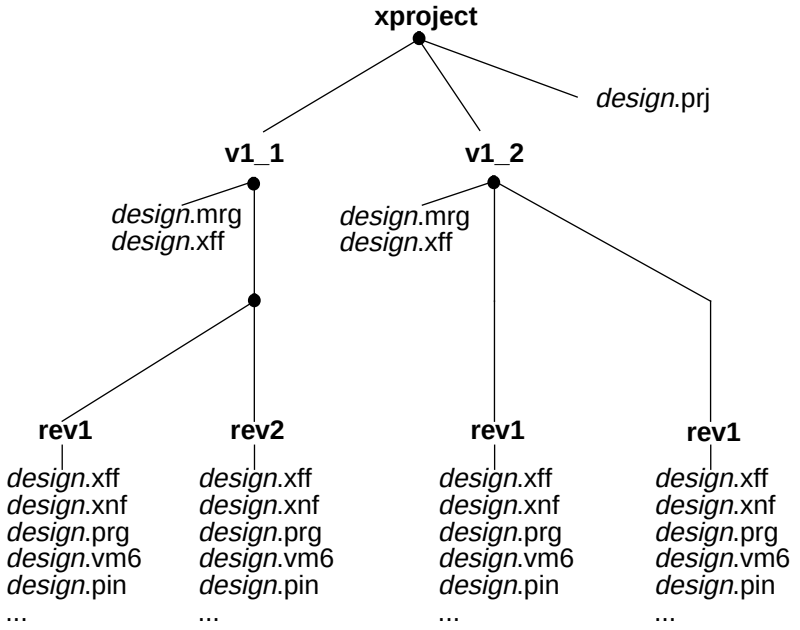


Figure A-2 The xproject Directory Tree

Design Files

Table A-1 lists some of the important files that are part of a Xilinx EPLD design. All the files in the table have the name *design_name*; they differ only in the extension. All these files are located in a revision (rev*n*) subdirectory of xproject unless otherwise stated.

Table A-1 EPLD Design Files

File Extension	Description
.1	Viewlogic schematic file, located in the Viewlogic project directory (not xproject).
.eqn	PLUSASM file showing the logic equations after optimization and fitting.
.err	Error log file.
.gyd	Guide file, which stores the frozen pinout and information for reproducing some logic optimizations.
.inf	OrCAD netlist file, located in the OrCAD project directory (not xproject).
.lga	PLUSASM Assembler Log Report (see the "Reports" chapter).
.lgc	Optimization report (see the "Reports" chapter).
.log	Log file.
.map	Mapping report (see the "Reports" chapter).
.mrg	XNFMerge report. May show errors in flattening the schematic hierarchy. Located in the version (ver*n*) directory.
.par	Fitting Report (see the "Reports" chapter).
.pin	Pin Out Report (see the "Reports" chapter).
.pld	PLUSASM file for a behavioral design or a behavioral module of a schematic design, usually located in the parent directory of xproject.
.prg	Intel HEX programming file produced by the XEPLD fitter.
.prj	Design manager project data file, located in the xproject directory.

.res	Resource report (see the “Reports” chapter).
.sch	OrCAD schematic file, located in the OrCAD project directory (not xproject).
.tim	Timing report (see the “Reports” chapter).
.tsp	Timespec report (see the “Reports” chapter).
.vm6	Design database file. The XEPLD software uses the information in this file to generate timing simulation files and programming files.
.vsm	Viewsim file for Viewlogic timing simulation, located in the Viewlogic project directory (not xproject).
.vst	VST file for OrCAD timing simulation, located in the OrCAD project directory (not xproject).
.xff	Xilinx Flattened Format file. The file that results when multiple .xnf files are merged into a single design file. Each version, implementation, and revision of the design has its own .xff file (see Figure B-2).
.xnf	Xilinx Netlist Format file. A design file is translated into this format when you create a new design project.

Index

M

N

O

P

R

S

Trademark Information

XILINX®, XACT, XC2064, XC3090, XC4005, and XC-DS501 are registered trademarks of Xilinx. All XC-prefix product designations, XACT-Floorplanner, XACT-Performance, XAPP, XAM, X-BLOX, X-BLOX plus, XChecker, XDM, XDS, XEPLD, XPP, XSI, BITA, Configurable Logic Cell, CLC, Dual Block, FastCLK, HardWire, LCA, Logic Cell, LogicProfessor, MicroVia, PLUSASM, SMARTswitch, UIM, VectorMaze, VersaBlock, VersaRing, and ZERO+ are trademarks of Xilinx. The Programmable Logic Company and The Programmable Gate Array Company are service marks of Xilinx.

IBM is a registered trademark and PC/AT, PC/XT, PS/2 and Micro Channel are trademarks of International Business Machines Corporation. DASH, Data I/O and FutureNet are registered trademarks and ABEL, ABEL-HDL and ABEL-PLA are trademarks of Data I/O Corporation. SimuCad and Silos are registered trademarks and P-Silos and P/C-Silos are trademarks of SimuCad Corporation. Microsoft is a registered trademark and MS-DOS is a trademark of Microsoft Corporation. Centronics is a registered trademark of Centronics Data Computer Corporation. PAL and PALASM are registered trademarks of Advanced Micro Devices, Inc. UNIX is a trademark of AT&T Technologies, Inc. CUPL, PROLINK, and MAKEPRG are trademarks of Logical Devices, Inc. Apollo and AEGIS are registered trademarks of Hewlett-Packard Corporation. Mentor and IDEA are registered trademarks and NETED, Design Architect, QuickSim, QuickSim II, and EXPAND are trademarks of Mentor Graphics, Inc. Sun is a registered trademark of Sun Microsystems, Inc. SCHEMA II+ and SCHEMA III are trademarks of Omation Corporation. OrCAD is a registered trademark of OrCAD Systems Corporation. Viewlogic, Viewsim, and Viewdraw are registered trademarks of Viewlogic Systems, Inc. CASE Technology is a trademark of CASE Technology, a division of the Teradyne Electronic Design Automation Group. DECstation is a trademark of Digital Equipment Corporation. Synopsys is a registered trademark of Synopsys, Inc. Verilog is a registered trademark of Cadence Design Systems, Inc.

Xilinx does not assume any liability arising out of the application or use of any product described or shown herein; nor does it convey any license under its patents, copyrights, or maskwork rights or any rights of others. Xilinx reserves the right to make changes, at any time, in order to improve reliability, function or design and to supply the best product possible. Xilinx will not assume responsibility for the use of any circuitry described herein other than circuitry entirely embodied in its products. Xilinx devices and products are protected under one or more of the following U.S. Patents: 4,642,487; 4,695,740; 4,706,216; 4,713,557; 4,746,822; 4,750,155; 4,758,985; 4,820,937; 4,821,233; 4,835,418; 4,853,626;

4,855,619; 4,855,669; 4,902,910; 4,940,909; 4,967,107; 5,012,135; 5,023,606; 5,028,821; 5,047,710; 5,068,603; 5,140,193; 5,148,390; 5,155,432; 5,166,858; 5,224,056; 5,243,238; 5,245,277; 5,267,187; 5,291,079; 5,295,090; 5,302,866; 5,319,252; 5,319,254; 5,321,704; 5,329,174; 5,329,181; 5,331,220; 5,331,226; 5,332,929; 5,337,255; 5,343,406; 5,349,248; 5,349,249; 5,349,250; 5,349,691; 5,357,153; 5,360,747; 5,361,229; 5,362,999; 5,365,125; 5,367,207; 5,386,154; 5,394,104; 5,399,924; 5,399,925; 5,410,189; 5,410,194; 5,414,377; RE 34,363, RE 34,444, and RE 34,808. Other U.S. and foreign patents pending. Xilinx, Inc. does not represent that devices shown or products described herein are free from patent infringement or from any other third party right. Xilinx assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction if such be made. Xilinx will not assume any liability for the accuracy or correctness of any engineering or software support or assistance provided to a user.

Xilinx products are not intended for use in life support appliances, devices, or systems. Use of a Xilinx product in such applications without the written consent of the appropriate Xilinx officer is prohibited.