Chapter 1

Watch Design — Implementation Tools Tutorial

This chapter contains the following sections.

- “Installing the Tutorial Files”
- “Step 1: Creating an Implementation Project”
- “Step 2: Specifying Options”
- “Step 3: Translating the Design”
- “Step 4: Mapping the Design”
- “Step 5: Using Timing Analysis to Evaluate Block Delays After Mapping”
- “Step 6: Placing and Routing the Design”
- “Step 7: Evaluating Post-Layout Timing”
- “Step 8: Creating Timing Simulation Data”
- “Step 9: Using the Flow Engine to Create Configuration Data”
- “Step 10: Using the PROM File Formatter”

Installing the Tutorial Files

This tutorial demonstrates the steps in the Alliance Series design implementation flow for the Watch design. Before beginning this tutorial, you should have already entered the Watch design into the design entry tool of your choice.

Note: This tutorial describes the basic design implementation flow. For more information on the Alliance design flow and implementation methodologies, see the online version of the Xilinx *Development System Reference Guide.*

Depending on your design entry tool, the input design is described by either a Xilinx netlist file (*.xnf, *.sxnf) or an EDIF file (*.sedif, *.edf, *.edn). This tutorial passes an input netlist from the entry tool to the Alliance design implementation tools, while incorporating placement constraints through a User Constraints File (UCF).

Before proceeding to Step 1 in the tutorial, create a working directory with the tutorial files as follows.

1. Create an empty working directory named Watch.
2. Copy the following files created with your design entry tool into the Watch directory.

File Name	Description
watch.*	Input netlist file
tenths.ngc	LogiBLOX implementation file
watch.ucf	User constraints file

Step 1: Creating an Implementation Project

The design implementation tools are organized under a single program called the Design Manager. The Design Manager helps you manage the design flow process by keeping track of design versions and the implementation revisions within each version. The Design Manager also provides access to the entire suite of Xilinx implementation tools needed to complete a design.

Use the following steps to create an implementation project.

1. Go to the directory containing your copy of the Watch design files.
2. On a workstation, enter the following to start the Design Manager.

   ```
   xilinx &
   ```

   You can also start the Design Manager as follows.

   ```
   dsgnmgr &
   ```

   On a PC, select the following to start the Design Manager.

   Start → Programs → Xilinx → Design Manager

3. When running the Design Manager for the first time, there are no projects available. To create a new implementation project for the tutorial design, select **`File`** → **`New Project`** to open the New Project dialog box as shown in the following figure.

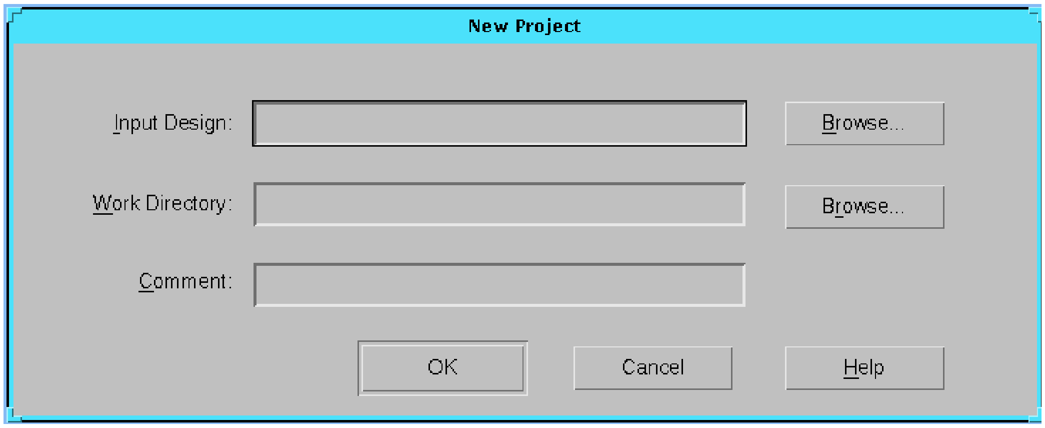


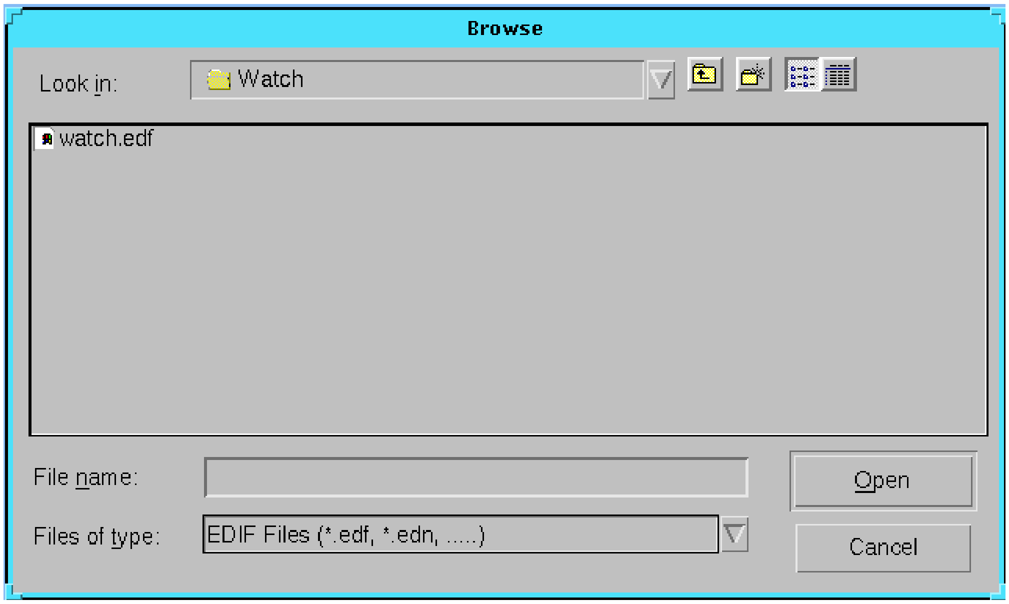
Figure 1-1 New Project Dialog Box

This dialog box contains the following fields.

Table 1-1 New Project Dialog Box Fields

Field	Description
Input Design	Top level netlist file containing the design definition
Work Directory	Directory used to store the implementation data created as the design is compiled
Comment	Enter any optional notation for the design in this field

4. Click **`Browse`** to the right of the Input Design field to specify the input design. The Browse dialog box appears as shown in the following figure.



Look in: Watch

Files of type: EDIF Files (*.edf, *.edn, ...)

Figure 1-2 Browse Dialog Box

5. Select the appropriate file type from the drop-down list in the Files of Type field. For the tutorial design shown in the "Browse Dialog Box" figure, EDIF is selected.

6. Select the Watch design file. The file name appears in the File Name field.

Note: You may need to change directories to the area containing the tutorial design files.

7. Click **Open**.

   The Browse dialog box closes and the New Project dialog box is updated to include the specified input netlist. By default, the Work Directory field is set to the directory containing the input design. If preferred, you can set this to another directory. Because the files were previously copied to the Watch directory, this directory is used for the implementation project and resulting output files.

8. In the Comment field, enter the following.

   ```
   -tutorial
   ```

9. Click **OK** to close the New Project dialog box and update the Design Manager with the specified project, as shown in the following figure.

Note: The toolbox buttons on the right side of the Design Manager are inactive at this point. To use these tools, you must first create a design version and an implementation revision.

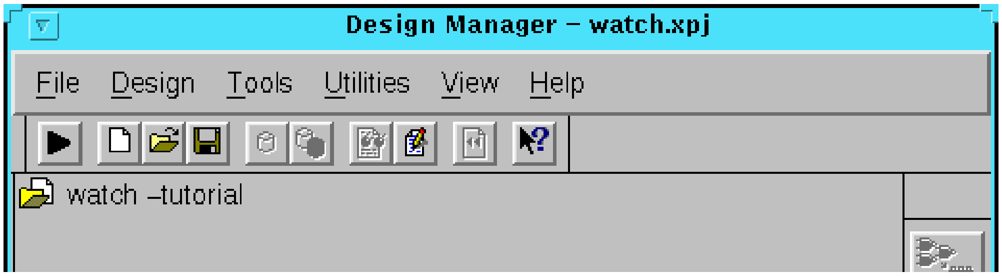


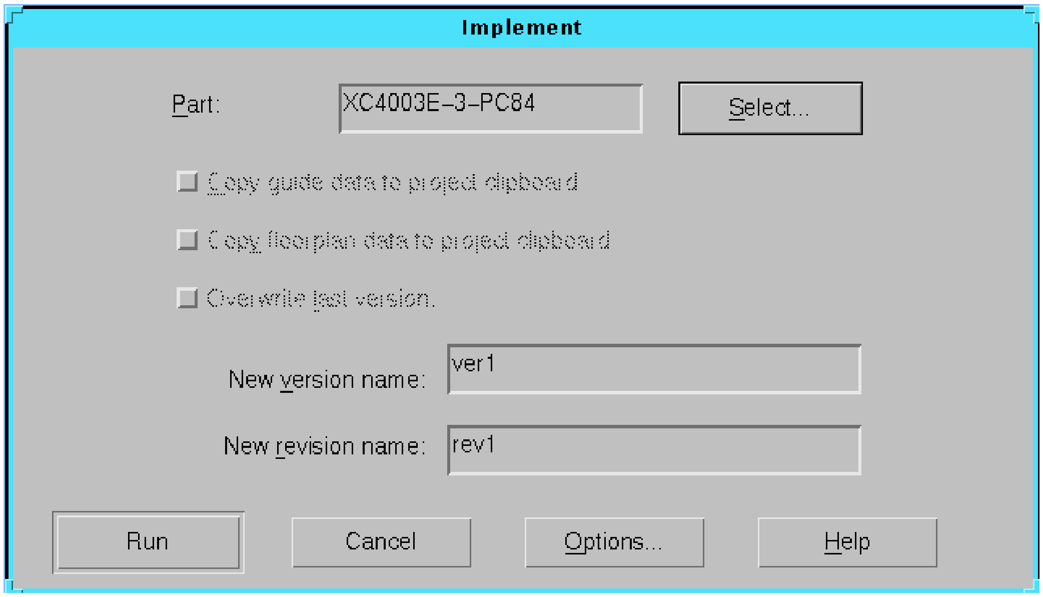
Figure 1-3 Watch Project in Design Manager

Step 2: Specifying Options

Each time a change is made to the input design, a new design version is created in the Design Manager. You can use the tools to create as many implementation revisions as you want for a design version. For example, if you try various implementation strategies on a netlist, several revisions are created for the single design version. By default, the Design Manager only keeps track of Xilinx-created files. The input design is not stored with the implementation data in the Xilinx implementation project area.

Use the following steps to specify options for the design.

1. Select **Design** → **Implement** to open the Implement dialog box as shown in the following figure. The New Version Name field shows ver1 as the default version, and the New Revision Name field shows rev1 as the default revision.



Implement

Part: XC4003E-3-PC84

□ Copy guide data to project clipboard

□ Copy floorplan data to project clipboard

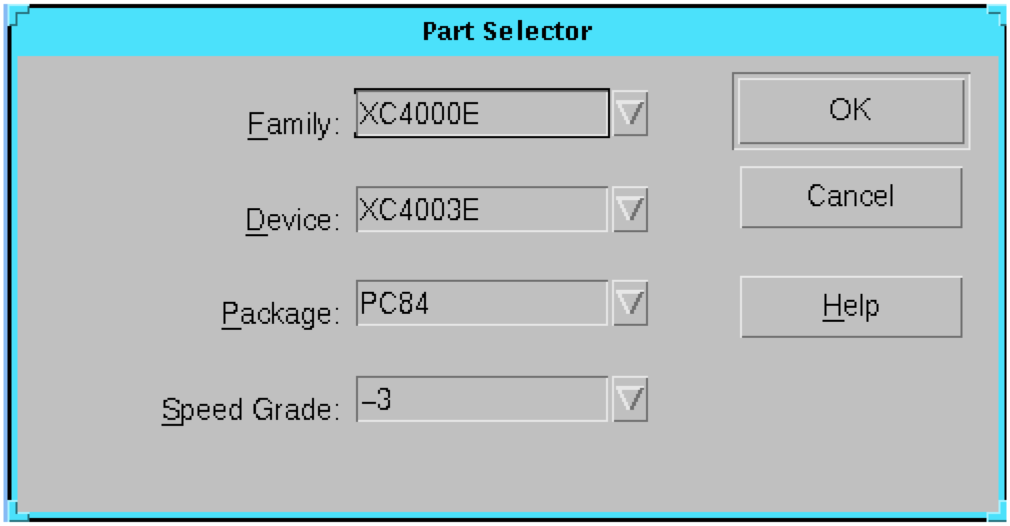
□ Overwrite last version.

New version name: ver1

New revision name: rev1

Figure 1-4 Implement Dialog Box

2. The Part field may contain a part number if you specified the target device in your design entry tool. If this field is empty, click **`Select`** to display the Part Selector dialog box.



Part Selector

Family: XC4000E

Device: XC4003E

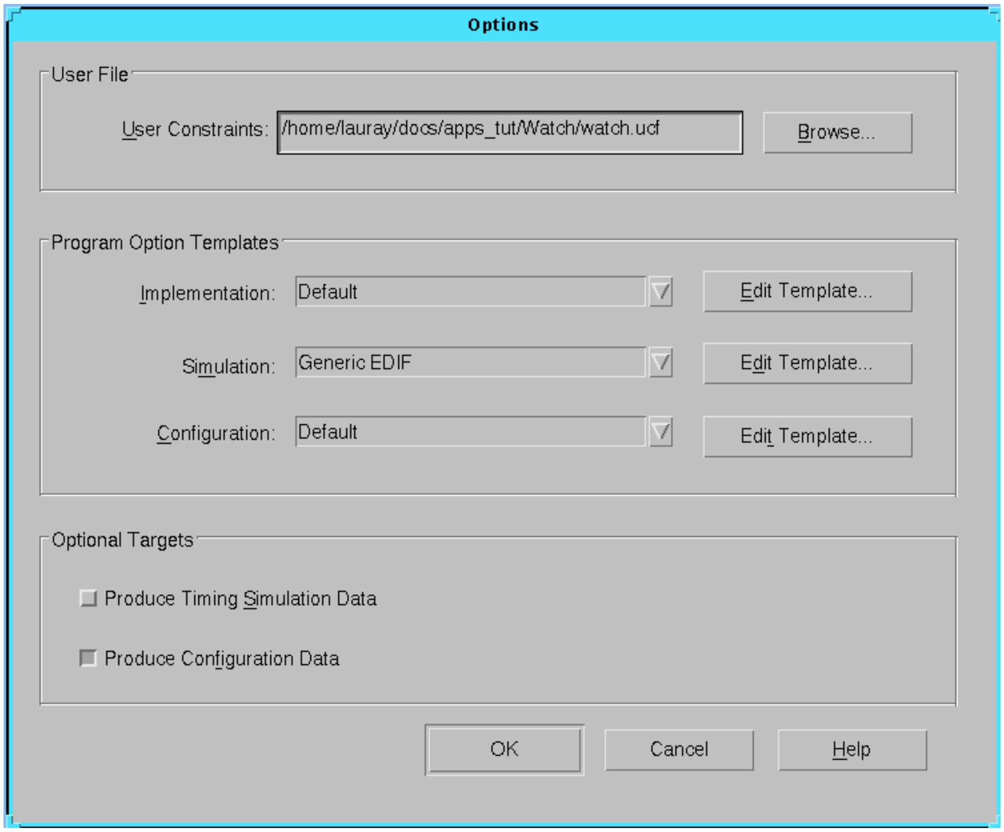
Package: PC84

Speed Grade: -3

Figure 1-5 Part Selector Dialog Box

3. Use the drop-down lists for the fields in the Part Selector dialog box to enter the Family, Device, Package, and Speed Grade for the design. This design targets an XC4003E-3-PC84. Click **`OK`**. The part number appears in the Part field in the Implement dialog box.

4. Select **`Options`** to open the Options dialog box.



User Constraints: /home/lauray/docs/apps_tut/Watch/watch.ucf

Implementation: Default

Simulation: Generic EDIF

Configuration: Default

☐ Produce Timing Simulation Data

☐ Produce Configuration Data

Figure 1-6 Options Dialog Box

You can specify a User Constraints File (UCF) and any optional processing targets in the Options dialog box. You can also access implementation, simulation, and configuration templates.

Templates are a convenient way of selecting several groups of options for a design implementation. The available options depend on the target device family. For example, you can use one template for a quick place and route and a different template for implementing a certain configuration option.

5. Verify that the entire path to the watch.ucf file is specified in the User Constraints field. When creating a project, if you include a user-generated UCF file in the same directory as your input design file, then the UCF file automatically appears in this field.

You can open the watch.ucf file with a text editor to view the constraints specified for this design. For a detailed explanation of constraints as well as examples with proper syntax, refer to the Xilinx *Libraries Guide.*

6. Select **`Edit Template`** next to the Implementation Program Option.

   The XC4000 Implementation Options dialog box is displayed. The implementation templates control how the software maps, places, routes, and optimizes a design.

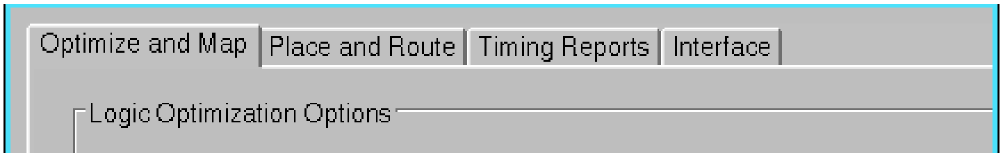


Figure 1-7 XC4000 Implementation Options Dialog Box

7. Select the **`Timing Reports`** tab.
8. Select **`Produce Logic Level Timing Report`**. The option to Produce Post Layout Timing Report should be selected by default. For both reports, select **`Report Paths Using Advanced Design Analysis (No Timing Constraints)`**.

   The timing reports generated after Map and PAR (place and route) are useful for evaluating design performance. They are analyzed in detail later in this tutorial.

9. Click **`OK`** to save the Implementation options. Click **`OK`** again to exit the Options dialog box.

Step 3: Translating the Design

The Design Manager manages the files created during the implementation process while the Flow Engine controls the implementation process itself. The programs run by the Flow Engine use the settings supplied by you in the various dialog boxes and templates. The Flow Engine gives you complete control over how a design is processed. Typically you should set all implementation options first and then run through the entire flow. However, to demonstrate the flexibility of the Flow Engine, and to provide you with a working knowledge of the tool, this tutorial proceeds through each step in the flow.

Based on the specified constraints and the selected report files, you can now implement the Watch design. The first step in design implementation is translation. During translation, NGDBuild performs the following functions.

- Converts input design netlists and writes results to a single merged NGD netlist. The merged netlist describes the logic in the design as well as any location and timing constraints.
- Performs timing specification and logical design rule checks
- Adds the User Constraints File (UCF) to the merged netlist.

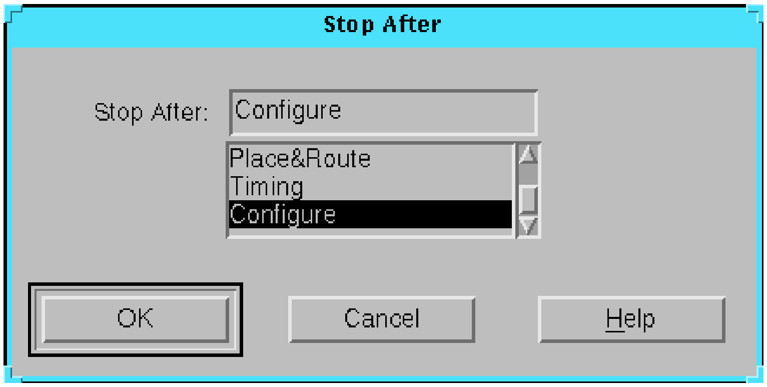
IMPORTANT: Read this Before Translating the Design — How to Stop the Design Processing

Before you start the Flow Engine and the translation of the tutorial design, review the following procedure for stopping the processing of the design after the translate step. Because the translate step for the tutorial design finishes quickly, you should be familiar with this procedure before you start the Flow Engine.

Setting a break point after the Translate step is useful when you want to perform a functional simulation of a design and copy the resulting *design*.ngd file to your working directory. After copying the *design*.ngd file, you can run the appropriate NGD2XXX program on the file to create functional simulation data. For more information on the NGD2XXX programs, see the appropriate chapter in the *Development System Reference Guide*.

Note: This same procedure can be used at any time in the Flow Engine to stop after any of the steps in the design flow.

1. In this tutorial, you want to stop processing the design after the Translate step. To do this, you must set a break point to stop the Flow Engine. To stop after the Translate step from within the Flow Engine, click the stop sign toolbar icon while Translate is running.
2. The Stop After dialog box is displayed with the default setting of Configure as shown in the following figure. The list box displays the break points appropriate for the current state of the design. Because the design is not processed at this point, all possible break points are listed.

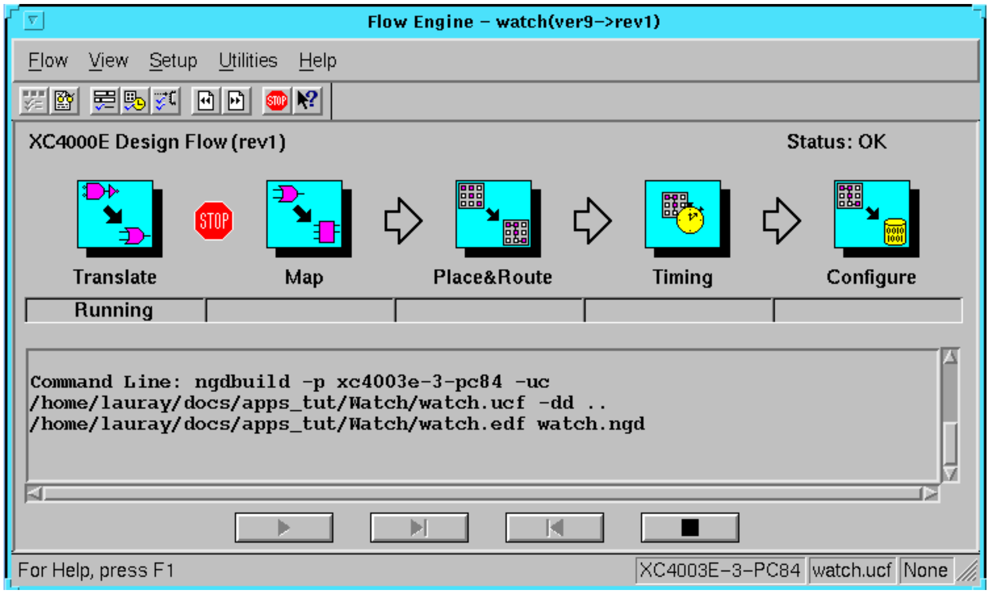


Stop After: Configure

Figure 1-8 Stop After Dialog Box

3. Select Translate in the list box and click **OK**. The stop sign is added to the design flow between the Translate and Map steps as shown in the following figure.

Note: The status bar at the bottom of the Flow Engine window is updated with the specified user constraints file (watch.ucf).



```
Command Line: ngdbuild -p xc4003e-3-pc84 -uc
/home/lauray/docs/apps_tut/Watch/watch.ucf -dd ..
/home/lauray/docs/apps_tut/Watch/watch.edf watch.ngd
```

Figure 1-9 Translating Design

Starting the Flow Engine and Translating the Design

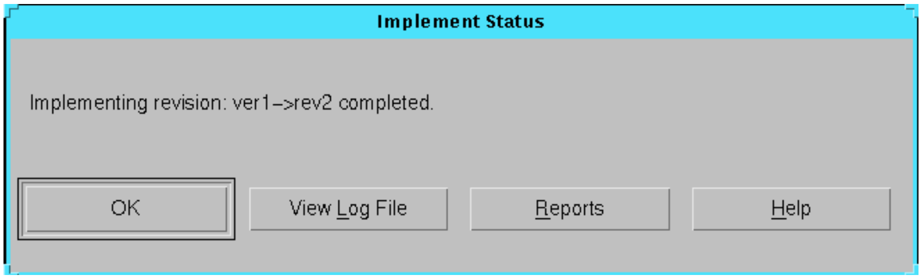
To translate the design, follow these steps.

1. Click **Run** in the Implement dialog box.

   This starts the Flow Engine. The steps in the design flow are graphically represented in the upper half of the Flow Engine window. The status of each stage is also shown. Refer to the “Translating Design” figure.

2. Perform the procedure previously described in the “IMPORTANT: Read this Before Translating the Design — How to Stop the Design Processing” section to stop the processing of the design.

3. After the Translate step is done, the Implement Status dialog box appears. Select **OK**.



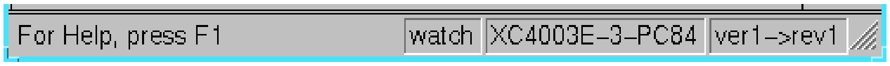
Implementing revision: ver1->rev2 completed.

Figure 1-10 Implement Status Dialog Box

4. The Design Manager window lists rev1 under the initial version of the watch project. The status of the revision is noted as (Translated, OK). *Translated* refers to the state of the design and is updated throughout the tutorial as the different compilation stages are completed. *OK* is the status of the current state and indicates no errors in the design processing.

Design Manager Status Bar

At the bottom of the Design Manager window is the status bar. The status bar lists the current project, target device, and currently selected version / revision pair. The left-hand portion of the status bar lists information on what is currently selected by the cursor.



For Help, press F1 | watch | XC4003E-3-PC84 | ver1->rev1

Figure 1-11 Design Manager Status Bar

Design Manager Toolbox

The toolbox, located on the right side of the Design Manager, becomes active with your first implementation revision. The icons in the toolbox are only active when a revision is selected. Icons in the toolbox (as shown in the following figure) represent the Flow Engine, Timing Analyzer, Floorplanner, PROM File Formatter, Hardware Debugger, and EPIC Design Editor.

Note: The toolbar has drag and drop capability.

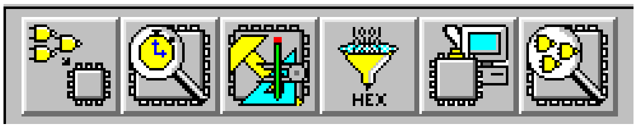


Figure 1-12 Design Manager Toolbox

Step 4: Mapping the Design

At this point, the input netlist has been translated and merged into a single design file. The design is now ready to be mapped into CLBs and IOBs. After mapping, the design is placed and routed. The final step in the design flow is the Configure step in which a configuration bitstream is created for downloading to a target device or for formatting into a PROM programming file.

Map performs the following functions.

- Allocates CLB and IOB resources for all basic logic elements in the design
- Processes all location and timing constraints, performs target device optimizations, and runs a design rule check on the resulting mapped netlist

Use the following steps to Map the tutorial design.

1. Click the Flow Engine icon in the Design Manager toolbox.

   The Flow Engine starts with the settings specified previously in this tutorial.

2. Select **`Flow`** → **`Step`** to start the Map process.

   Although a break point is not specified, the Flow Engine stops after the Map stage because a single step is executed. Refer to the “Mapping Design” figure.

Note: There are several ways to begin the implementation process. For example, you can select **`Flow`** → **Run**, or you can use the equivalent control buttons, as shown in the following figure.

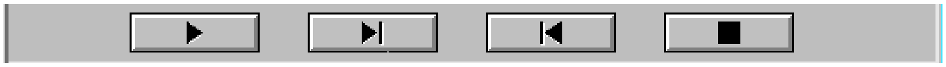
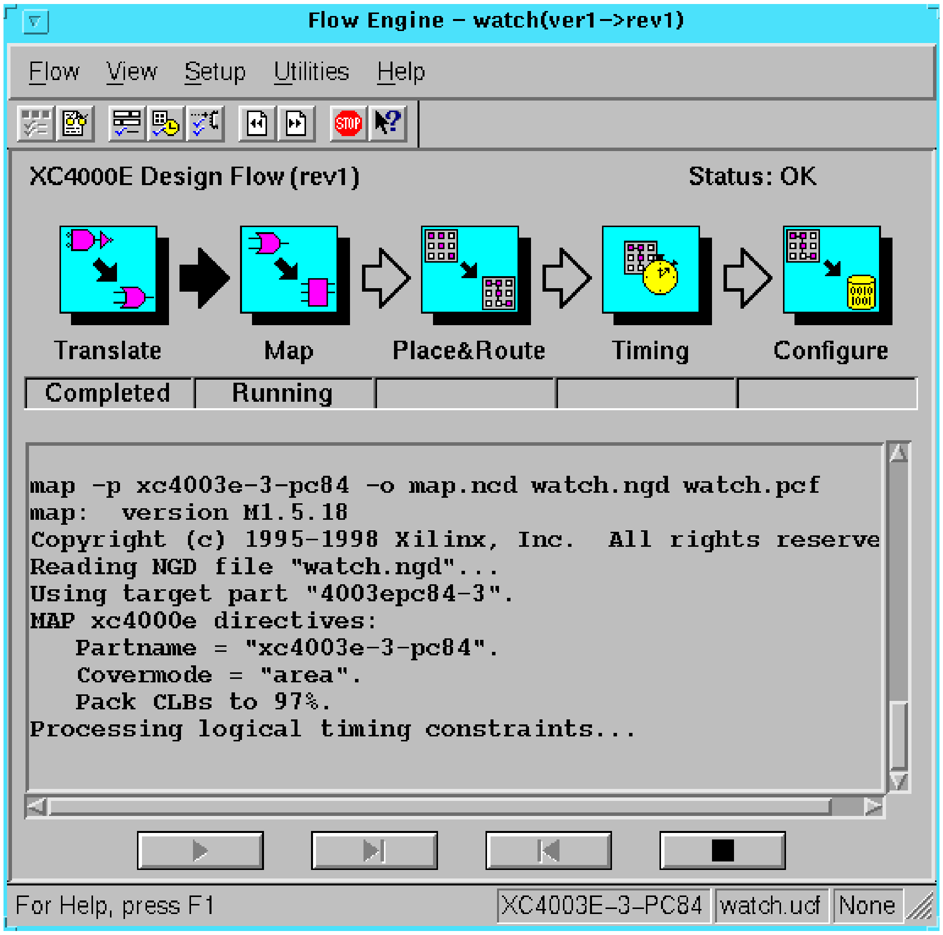


Figure 1-13 Control Panel

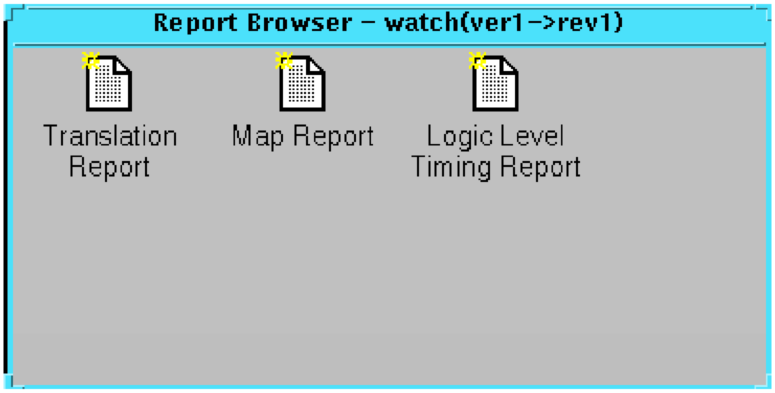


```
map -p xc4003e-3-pc84 -o map.ncd watch.ngd watch.pcf
map:  version M1.5.18
Copyright (c) 1995-1998 Xilinx, Inc.  All rights reserve
Reading NGD file "watch.ngd"...
Using target part "4003epc84-3".
MAP xc4000e directives:
   Partname = "xc4003e-3-pc84".
   Covermode = "area".
   Pack CLBs to 97%.
Processing logical timing constraints...
```

Figure 1-14 Mapping Design

3. While Map is running, you can review the currently available reports using the Report Browser by selecting **`Utilities`** → **`Report Browser.`**

   The Report Browser window appears with the Translation Report listed. The Map Report and Logic Level Timing Report files are created after Map is finished. New reports are denoted with a gold star in the upper left corner of the file icon.



Report Browser – watch(ver1->rev1)

Figure 1-15 Report Browser after Running Map

4. Double-click on a report to review its output. The following table lists the types of reports and descriptions.

Table 1-2 Report Browser Reports

Report	Description
Translation Report	Includes warning and error messages from the translation process.
Map Report	Includes information on how the target device resources are allocated, references to trimmed logic, and device utilization. For detailed information on the Map report, refer to the *Development System Reference Guide*.
Logic Level Timing Report	Provides a summary analysis of your timing constraints based on block delays and estimates of route delays. This report is produced after Map and prior to PAR (place and route).

At this point, in the Design Manger window, the current version/ revision should indicate the status (Mapped, OK). The design is now mapped to the target architecture. The next step is checking the design paths for excessive block delays.

Step 5: Using Timing Analysis to Evaluate Block Delays After Mapping

After the design is mapped, you can use the Logic Level Timing Report to evaluate the logical paths in the design. Because the design is not placed and routed yet, actual routing delay information is not available. The timing report describes the logical block delays and estimated routing delays. The net delays that are provided are based on an optimal distance between blocks (also referred to as *unplaced floors*).

Estimating Timing Goals with 50/50 Rule

You can get a preliminary idea of how realistic your timing goals are by evaluating a design after the map stage. A rough guideline (known as the 50/50 rule) specifies that the block delays in any single path make up approximately 50% of the total path delay after the design is routed. For example, a path with 10ns of block delay should meet a 20ns timing constraint after it is placed and routed. If your design is extremely dense, or if you are using an architecture with fewer routing resources, your net delays can be more than 50% of the total path delay.

Report Paths Using Advanced Design Analysis Option

Because timing constraints are not defined for this tutorial design, the Report Paths Using Advanced Design Analysis option is selected. This option forces the Logic Level Timing Report to perform a default period and path analysis. In this report, look at the minimum period attained and the percentage of block (logic) delay versus routing delay for each of the default calculations. The unplaced floors listed are estimates (indicated by the letter “e” next to the net delay) based on optimal placement of blocks.

To obtain an even more accurate assessment of timing requirements, you can place a design without routing it. The resulting placed floors are based on optimistic routing estimates, and provide a more realistic account of net delays compared to the unplaced floors estimates from the mapped design. The placed estimates calculate an absolute minimum routing for each placement. For more detailed information, refer to the *Development System Reference Guide.*

If you do not generate a Logical Level Timing Report, PAR still processes a design based on the relationship between the block delays, floors, and timing specifications for the design. For example, if a PERIOD constraint of 8 ns is specified for a path, and there are block delays of 7 ns and unplaced floor net delays of 3 ns, PAR stops and generates an error message. In this case, PAR fails because it determines that the total delay (10 ns) is greater than the constraint placed on the design (8 ns). You can use the Logic Level Timing Report to determine timing violations that may occur prior to running PAR.

Step 6: Placing and Routing the Design

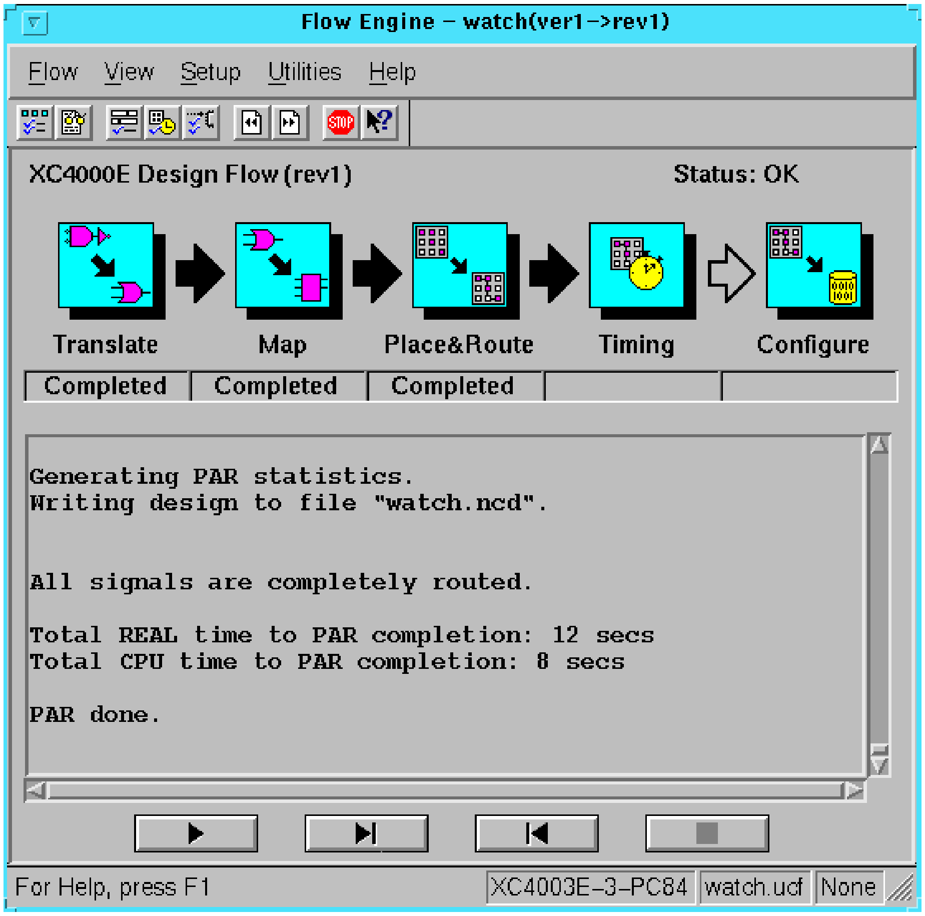
After the mapped design is evaluated to verify that block delays are reasonable given the design specifications, the design can be placed and routed. The Flow Engine can perform the following place and route algorithms.

- Timing Driven — run PAR with timing constraints specified from within the input netlist or from a constraints file
- Non-Timing Driven — run PAR and ignore all timing constraints

In this tutorial, non-timing driven placement and non-timing driven routing are automatically performed by PAR because there are no timing constraints specified for this design.

To place and route the design, perform the following steps.

1. In the Flow Engine window, select **`Flow`** → **`Step`** to run only the Place & Route step. The Status:OK message in the upper right corner of the Flow Engine indicates that no errors were generated by PAR at this point. Refer to the following figure.



```
Generating PAR statistics.
Writing design to file "watch.ncd".


All signals are completely routed.

Total REAL time to PAR completion:  12 secs
Total CPU time to PAR completion: 8 secs

PAR done.
```

Figure 1-16 Placing and Routing Design

2. Review the reports generated to make sure the place and route process finished as expected.

   The three new reports created in the Report Browser are the Place and Route Report, the Pad Report, and the Asynchronous Delay Report, as shown in the following figure and described in the following table.

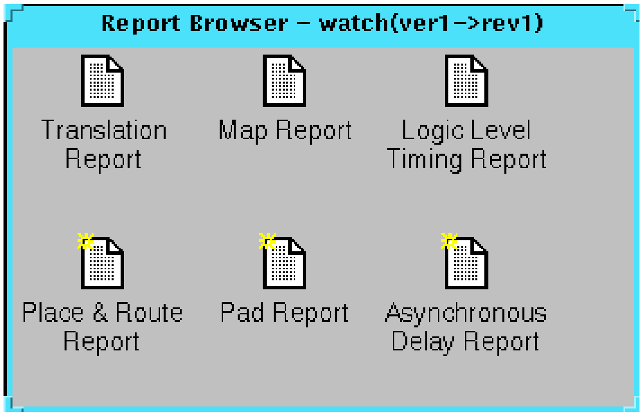

Report Browser – watch(ver1->rev1)
Translation Report
Map Report
Logic Level Timing Report
Place & Route Report
Pad Report
Asynchronous Delay Report

Figure 1-17 Reports Available After Place & Route

Table 1-3 Description of Reports Available After Place & Route

Report	Description
Place & Route Report	Provides a device utilization and delay summary. Use this report to verify that the design successfully routed and that all timing constraints were met.
Pad Report	Contains a report of the location of the device pins. Use this report to verify that pins locked down were placed in the correct location.
Asynchronous Delay Report	Lists all nets in the design and the delays of all loads on the net.

Note: In the Design Manager window, the status of the current version / revision is now (Routed, OK).

Step 7: Evaluating Post-Layout Timing

After the design is placed and routed, you can generate the Post Layout Timing Report to verify that the design meets your specified timing goals. This report evaluates the logical block delays and the routing delays. The net delays are now reported as actual routing delays after the place and route process.

Because the option to produce a Post-Layout Timing Report was previously selected, you can proceed with the timing step in the design flow.

1. In the Flow Engine, select **`Flow`** → **`Step`** to run the timing step.
2. After the timing step is completed, select **`Utilities`** → **`Report Browser.`**
3. Double-click on the Post Layout Timing Report to open it. The following is a summary of this report.
   - The minimum period value increased due to the actual routing delays.
   - After the Map step, logic delay contributed to about 80% of the minimum period attained. The post-layout report indicates that the logical delay value did not change much. However, the total unplaced floors estimate did change. Routing delay after PAR equals about 40% of the period; a true report of net delays after the place and route step.
   - The post-layout result does not necessarily follow the 50/50 rule previously described because the worst case path includes primarily component delays. After the design was mapped, block delays constituted about 80% of the period. After place and route, the majority of the worst case path is still made up of logic delay. Since total routing delay makes up only a small percentage of the total path delay, spread out across three nets, expecting this to be reduced any further is unrealistic. You can reduce excessive block delays and improve design performance by decreasing the number of logic levels in the design.

Step 8: Creating Timing Simulation Data

After the design is placed and routed and the timing is statically verified, the next step is to create timing simulation data. To create timing simulation data, perform the following steps in the Flow Engine.

1. Select **`Setup`** → **`Options`** to open the Options dialog box.

2. Select the simulator that corresponds to your design entry tool from the Simulation drop-down list in the Program Option Templates section of the dialog box.
3. Click on the Produce Timing Simulation Data option in the Optional Targets section of the dialog box.
4. Click **`OK`** to close the Options dialog box.
5. Select **`Flow`** → **`Step Back`** from the Flow Engine menu to back up to the Place & Route step.
6. Select **`Flow`** → **`Step`** to run the timing step.

   The timing step produces timing simulation data. In this tutorial, this stage is run again because the option to produce timing simulation data was not selected in the initial pass. For most designs, you will select all options at the beginning of the design processing, and it will not be necessary to repeat any steps.

   During the timing step, the Flow Engine runs the NGDAnno program to create a back-annotated NGD file. The NGD file is then used as input to one of the NGD2XXX programs to produce the preferred simulation file format. By default, the files created are named *time_sim*. To make it easy to find the output files for your third-party simulation environment, the files are automatically copied to your working directory.

Step 9: Using the Flow Engine to Create Configuration Data

The next step is creating configuration data. This step includes creating a bitstream for the target device by running the configure step in the Flow Engine, as follows.

Note: If the Flow Engine is not running, start it with the timed revision created in the previous step.

1. Select **`Setup`** → **`Options`** to open the Options dialog box.
2. Select Edit Template next to Configuration in the Program Option Templates section of the dialog box.

   The XC4000 Configuration Options dialog box appears. The configuration templates set options that define the initial configuration parameters, start-up sequence, readback capabilities, and other advanced features. In this tutorial, a configuration file is created that can be used for programming, verifying, and debugging XC4000E designs.

3. In the Configuration tab, verify that PullUp is selected next to the Done pin, and that the Perform CRC During Configuration option is selected.

4. Select the Readback tab, and verify that CCLK is selected as the readback clock.

5. Click **OK** to close the XC4000 Configuration Options dialog box.

6. Click **OK** to close the Options dialog box.

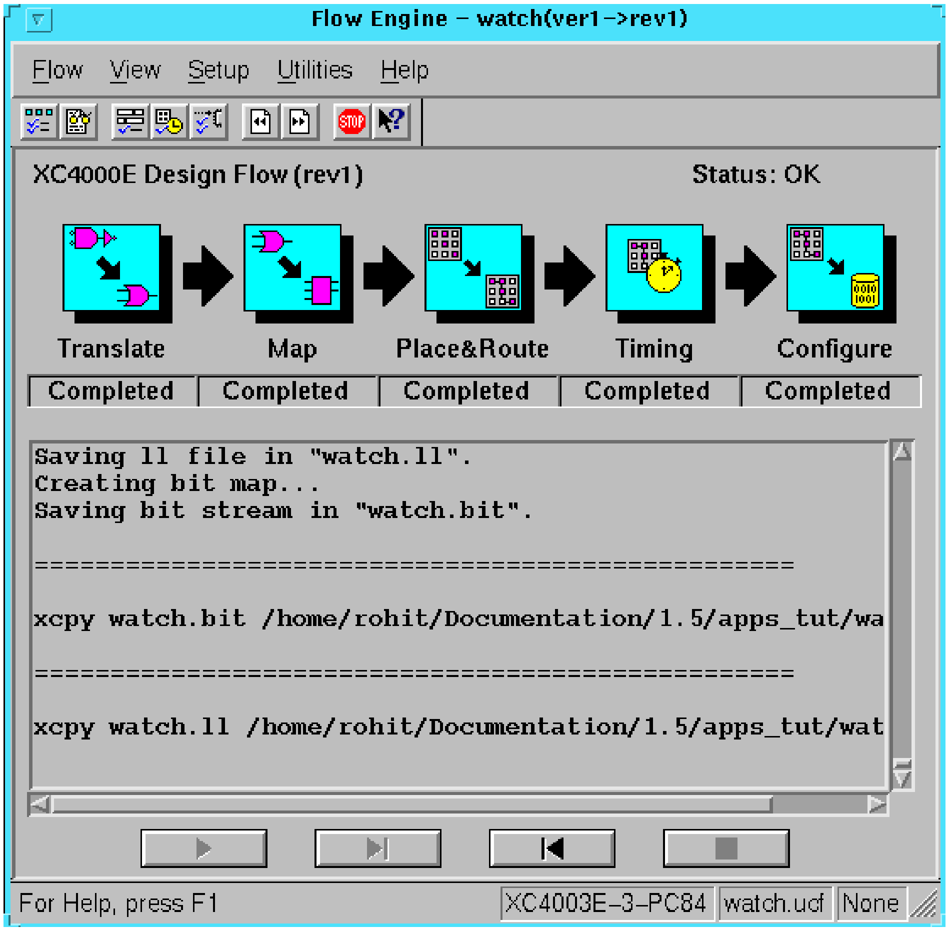
7. Select **Flow** → **Run** to run the configure step.

   The Flow Engine runs BitGen to create the configuration data. BitGen creates the *design_name*.bit and *design_name*.ll files (in this tutorial, the watch.bit and watch.ll files). The *design_name*.bit file is the actual configuration data. The *design_name*.ll file is the logical allocation file that is used during hardware debugging to determine the location of the probable points in the design. These files are automatically copied to your working directory. Verify that they are in this directory.

   The *design_name*.11 file is used to perform device readback with the Hardware Debugger tool. For more information on device readback, please refer to the *Watch Design- Hardware Verification Tutorial (ftp://ftp.xilinx.com/pub/documentation/M1.5_tutorials/wd_hwd_15.pdf)*

   For a listing of all avialable software tutorials, please take a look at the following URL: *http://www.xilinx.com/support/techsup/tutorials/*

   The following figure shows the Flow Engine window after the configure step is finished.



```
Saving ll file in "watch.ll".
Creating bit map...
Saving bit stream in "watch.bit".

==================================================

xcpy watch.bit /home/rohit/Documentation/1.5/apps_tut/wa

==================================================

xcpy watch.ll /home/rohit/Documentation/1.5/apps_tut/wat
```

Figure 1-18 Configuring Design

8. The Flow Engine saves the configuration options in the BitGen Report. Review the report using the Report Browser. Verify that the specified options were used when creating the configuration data.

9. Select **Flow** → **Close** to close the Flow Engine and the Report Browser.

Step 10: Using the PROM File Formatter

If you are going to program a single device using the Hardware Debugger, all you need is a *design*.bit file. If you are going to program several devices in a daisy chain configuration, or program your devices using a PROM, you must use the PROM File Formatter (PFF) to create a PROM file. The PROM File Formatter accepts any number of bitstreams and creates one or more PROM files containing one or more daisy chain configurations.

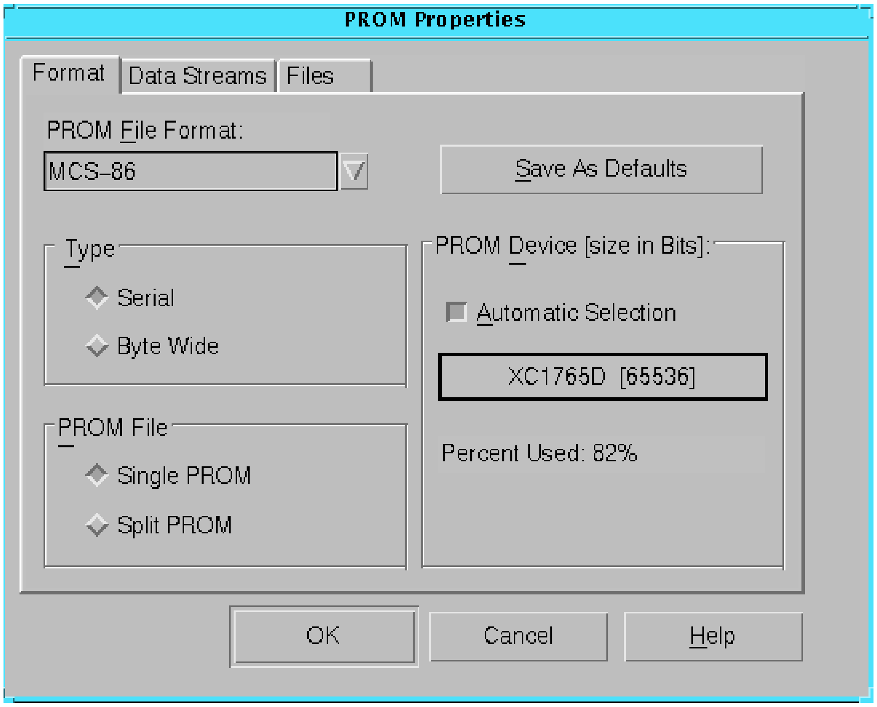
1. To start the PROM File Formatter, click the PROM File Formatter icon in the toolbox in the Design Manager.

   The PFF starts with a default PROM that matches the currently selected (configured) revision. At this point, you can add additional bitstreams to the daisy chain; create additional daisy chains; remove the current bitstream and start over; or immediately save the current PROM file configuration.

   The status bar at the bottom of the PFF window displays the PROM format, data format, current PROM size, and percentage of the selected PROM used by the current PROM configuration. The currently selected PROM is an XC1765D. 53,984 bits of data are required to hold the configuration bitstream for the XC4003E target device for this tutorial. The PFF determined that an XC1765D is the correct PROM because it can hold up to 65,536 configuration bits (or 82% full).

   The right half of the PFF window is a directory structure used for locating bitstreams. Only files with a .bit extension are shown in the list. For detailed information on using the PROM File Formatter to create daisy chains or complex PROM configurations, see the *PROM File Formatter Reference/User Guide*. This tutorial describes how to save the default PROM file.

2. Select **`File`** → **`PROM Properties`** to open the PROM Properties dialog box, shown in the following figure.



PROM File Format: MCS-86

Type
◆ Serial
◇ Byte Wide

PROM File
◆ Single PROM
◇ Split PROM

PROM Device [size in Bits]:
□ Automatic Selection
XC1765D [65536]

Percent Used: 82%

Figure 1-19 PROM Properties Dialog Box with Single PROM

3. Select the following options in this dialog box.

    - PROM File Format from the drop-down list
    - PROM Type
    - Number of PROMS used to hold the data

      If you have more data than space available in the PROM, you must split the data into several individual PROMs with the Split PROM option. In this case, only a single PROM is needed. Click **OK** to accept the PROM Properties.

4. Select **File** → **Save** to save the PROM file.

5. Specify your working directory as the area where the PROM Description File will be saved.

   The PROM File Formatter saves both the PROM file (watch.mcs) and a PROM Description File (watch.pdr). The PDR file can be reopened if any changes are required. Verify that the files exist in your directory.

6. Select **`File`** → **`Exit`** to close the PROM File Formatter.

   This completes the tutorial.