



Release Document

XC4000E V1.0.0 Pre-Release

December 1995

Attention: This software is a *pre-release* version.

Some core functionality, product features, third party interfaces, and platform support may not be available or is limited (see the enclosed release note for supported attributes). This software is not fully integrated with the current XACT tools and it is not fully tested to the Xilinx production level quality assurance tests.

Read This Before Installation

Versions and Compatibility

The following master table lists the Xilinx XC4000E Pre-Release software programs with their current version numbers.

Software Versions

Program	Version	Program	Version
LCA2XNF	XC4000E-PRE-1.0.0	XDelay	5.2.0
MakeBits	5.2.0	XDM	5.2.0
MakePROM	5.2.0	XMake	5.2.0
PPR	XC4000E-PRE-1.0.0	XNFBA	5.2.0
XABEL	5.2.0	XNFMerge	5.2.0
XBLOX	XC4000E-PRE-1.0.0	XNFPrep	XC4000E-PRE-1.0.0
XChecker	5.2.0	XSimMake	XC4000E-PRE-1.0.0

Contents

Chapter 1 Introduction

Contents.....	1-1
Software.....	1-1
Documentation.....	1-1
Xilinx PC Protection Key	1-2
Maintenance and Support	1-2

Chapter 2 Installation

Disk Space	2-1
Contents.....	2-2
Installing on a Personal Computer	2-2
License.....	2-3
Installing Viewlogic.....	2-3
Sun Workstation.....	2-3
HP Workstation	2-4
IBM RS6000 Workstation.....	2-4
Installing XSI	2-5
Sun Workstation.....	2-5
HP Workstation	2-6
IBM RS6000 Workstation.....	2-6
Installing Mentor	2-7
Sun Workstation.....	2-7
HP Workstation	2-8
Installing Core Tools	2-8
Sun Workstation.....	2-8
HP Workstation	2-9
IBM RS6000 Workstation.....	2-9
Installing for Use with XACTstep 6.0.....	2-10
Files Required.....	2-10
Modifying Your Setup.....	2-11

XACT Environment Variable	2-11
Path	2-11
Using the XC4000E Software Within XACTstep 6.0	2-12

Chapter 3 Features in This Release

XC4000E Viewlogic.....	3-1
XC4000E Library.....	3-1
PC	3-1
Sun4, HP, and IBM RS6000	3-2
FXC4000E Library	3-2
PC	3-3
Sun4, HP, and IBM RS6000	3-3
Switching to XC4000E from XC4000	3-3
New Symbols	3-4
Location Constraints	3-4
Simulating XC4000E RAM	3-5
XC4000E OrCAD	3-5
XC4000E Library.....	3-5
New Symbols	3-6
XC4000E MemGen	3-6
SYNC_RAM and DP_RAM	3-6
Default and Data Commands.....	3-7
Symbols	3-8
Pin Names of RAM Components	3-8
XC4000E X-BLOX.....	3-9
SYNC_RAM	3-9
DP_RAM	3-9
Conditions for Implementation in an IOB	3-10
XC4000E XDE	3-10
Synchronous and Dual-Port RAMs	3-11
SetMemory Command Initializes All XC4000E RAMs	3-11
H Multiplexers	3-12
IOB Clock Enable.....	3-13
IOB O/EC Routing Model.....	3-13
XC4000 to XC4000E Conversion	3-13
CMOS Thresholds	3-14
Start-Up Sequence Options	3-14
XC4000E PPR	3-14
Guided Design	3-15
XC4000E XACT-Performance Support.....	3-15
Asynchronous RAMs	3-15
Single-Port Synchronous RAMs	3-16

Dual-Port Synchronous RAMs	3-16
XC4000E XSI	3-16
Configuring Synopsys to Synthesize and Simulate	
XC4000E Designs	3-17
New Components in XC4000E Synthesis and	
Simulation Libraries	3-18
Current Restrictions Imposed By Synopsys 3.3	3-20
Using Synchronous and/or Level-Sensitive RAM	3-21
XC4000E MN8 Mentor Interface	3-21
Script Changes	3-26
MEN2XNF8, FNCSIM8, TIMSIM8	3-26
PLD_DVE	3-26
GEN_SCH8	3-27
GEN_SYM8	3-27
Unsupported Features in This Release	3-27
XACT-Floorplanner™	3-27
OrCAD	3-27
XDE	3-27

Chapter 4 Device and Package Support

Device and Package Support	4-1
----------------------------------	-----

Chapter 5 Known Issues

Software	5-1
Design Entry	5-1
Inversion Bubble Is Missing for OrCAD I/O Primitives	5-1
Illegally Instantiating I/Os with XSI May Cause	
FPGA Compiler or Design Compiler to Crash	5-2
Implementation	5-2
PPR: Setting Ignore_rlocs to True May Cause	
Errors in LCA2XNF, MakeBits, XDelay	5-2
PPR Issues ERROR 1582	5-3
Timing Simulation	5-3
Input Latch Is Modeled Incorrectly for Timing Simulation ...	5-3
XNF2WIR May Issue Error on 3-State Output Flip-Flop	5-4
XNF2WIR Fails with Error if -L Option Is Used	5-5
OrCAD: Setup and Hold NOT Checked on RAM	
Address and WE Pins	5-5
XNF2INF Issues Error for Unconnected WE or	
ADDR pins on RAMs	5-6

Downloading and Configuration.....	5-6
MakeBits Should Configure TTL Inputs by Default.....	5-6

Chapter 6 Xilinx Customer Support Information

Registration, Authorization, and Customer Service	6-1
Technical Support	6-1
Training	6-2

Chapter 1

Introduction

Welcome to the XC4000E V1.0.0 Pre-Release from Xilinx!

Contents

The Development System (DS) product that you received contains software, documentation, and/or hardware. New DS Base, Standard, and Extended packages contain hardware, software, and documentation. Interface and Update products contain software and documentation only.

Software

Xilinx software for the PC and for Sun4, HP, and IBM RS6000 workstations is provided on CD-ROM. It consists of the following:

- Subset of FPGA Core Implementation Tools for XC4000E only (DS-502)
- Viewlogic Interface and Libraries for XC4000E only (DS-391)
- OrCAD Interface and Libraries for XC4000E only (DS-35)
- Synopsys Interface and Libraries for XC4000E only (DS-401)
- Mentor8 Interface and Libraries for XC4000E only (DS-344)

Documentation

The following documentation is available for the XC4000E V1.0.0 Pre-Release:

- Online *XC4000E Libraries Guide Supplement* with the XACTstep 5.2/6.0 release, which you can access by clicking on the **Xilinx Online Docs** → **LIBRARIES SUPPLEMENT GUIDE** command.

- Online XC4000E data sheet with the XACTstep 5.2/6.0 release, which you can access by clicking on the **Xilinx Apps Info** → **XC4000E DATA SHEET** command.
- The latest revision of the XC4000E specifications on Xilinx WEBLINX, which you can access at the following address:
<http://www.xilinx.com>

Xilinx PC Protection Key

Xilinx and Viewlogic software are protected by a hardware key for the parallel port of your PC. This key is required to operate the software properly.

Maintenance and Support

This product comes with free technical and product information telephone support (toll-free in the U.S. and Canada). You can also fax and e-mail your questions. See the “Xilinx Customer Support Information” section of this release note for offices and phone numbers.

This product comes with one year of maintenance; you will receive all software and documentation updates automatically during that time. You will receive a notice at the end of the year giving instructions on how to renew your maintenance contract.

Chapter 2

Installation

This section explains how to install the XACTstep XC4000E Pre-Release software on CD-ROM. This software package is intended to support only XC4000E designs.

Note: Keep this software in a separate area. Do not install it over your existing VL-STD, OR-STD, MN8-STD, or SY-STD software from the XACTstep 5.2 release.

Disk Space

To install the XC4000E Pre-Release, you need the amount of disk space shown in the following table. All workstation installations require that you extract the tar files. If tar files are extracted directly from the CD drive on workstations, disk space (direct) applies. If tar files must be first copied from the CD drive, disk space (indirect) applies.

Interface	Platform	Disk Space (Direct)	Disk Space (Indirect)
OrCAD	PC	14 MB	N/A
Viewlogic	PC	13 MB	N/A
Core Tools	PC	57 MB	N/A
Viewlogic	Sun	16 MB	31 MB
Viewlogic	HP	16 MB	31 MB
Viewlogic	IBM RS6000	16 MB	31 MB
Synopsys	Sun	19 MB	38 MB
Synopsys	HP	20 MB	39 MB
Synopsys	RS6000	20 MB	40 MB

Interface	Platform	Disk Space (Direct)	Disk Space (Indirect)
Mentor	Sun	32 MB	63 MB
Mentor	HP	35 MB	69 MB
Core Tools	Sun	68 MB	135 MB
Core Tools	HP	71 MB	142 MB
Core Tools	RS6000	67 MB	134 MB

Contents

The CD-ROM included with this product contains the directories shown in the following table.

Directory	Contents
or_pc	Core tools, data files, and OrCAD interface on PC
vl_pc	Core tools, data files, and Viewlogic interface on PC
4ke_pc	Core tools and data files on PC
vl_ws	Tar files of Viewlogic interface and libraries on Sun, HP, and IBM RS6000
xsi_4ke	Tar files of XSI interface and libraries on Sun, HP, and RS6000
mn8_hppa	Tar file for Mentor8 interface and libraries on HP
mn8_sun4	Tar file for Mentor8 interface and libraries on Sun
xact_4ke	Tar files of core tools and data files on Sun, HP, and RS6000

Installing on a Personal Computer

This section explains how to load your Xilinx software on a personal computer. In these instructions, drive c: represents the PC hard disk, and drive e: represents the CD-ROM drive.

1. Create a new directory, such as “xpre1” on the c: drive.

```
cd c:\xpre1
```

2. Execute one of the following sets of commands.

- For Viewlogic, type the following:

```
xcopy e:\vl_pc\*. * /s
```

```
xcopy e:\4ke_pc\*. * /s
```

- For OrCAD, type the following:

```
xcopy e:\or_pc\*. * /s
```

```
xcopy e:\4ke_pc\*. * /s
```

- For core tools only on the PC, type the following:

```
xcopy e:\4ke_pc\*. * /s
```

3. Modify the XACT environment variable:

```
set XACT=c:\xpre1;existing_xact_path
```

Make sure that c:\xpre1 in the path statement is placed before the directory where the other Xilinx software resides.

License

For workstation installations, you must append the path of the version of the XACT software that you have to the XACT variable that is defined for the XC4000E software in order to access the license. The XACTstep XC4000E Pre-Release does not include any license files.

Installing Viewlogic

This section describes how to install Viewlogic software on Sun, HP, and IBM RS6000 workstations.

Sun Workstation

This section describes how to install Viewlogic software on a Sun workstation. You must extract two tar files from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the two tar files from the CD as follows.

1. Use the cd command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as */correct_path/xpre1*.
2. Execute the following commands.

```
tar -xf cd_rom_path/vl_ws/vl_sn.tar
tar -xf cd_rom_path/xact_4ke/xact_sn.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=(/correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure that you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

HP Workstation

This section describes how to install Viewlogic software on a Hewlett-Packard workstation. You must extract two tar files from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the two tar files from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.
2. Execute the following commands:

```
tar -xf cd_rom_path/VL_WS/VL_HP.TAR\;1
tar -xf cd_rom_path/XACT_4KE/XACT_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`;

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=(/correct_path/xpre1/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

IBM RS6000 Workstation

This section describes how to install Viewlogic software on an IBM RS6000 workstation. You must extract two tar files from the CD and set the XACT variable and path. If your CD player is directly

mounted on your workstation, you can directly extract the two tar files from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/v1_ws/v1_rs.tar
tar -xf cd_rom_path/xact_4ke/xact_rs.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=( /correct_path/xpre1/bin/rs6000 $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing XSI

This section describes how to install XSI software on Sun, HP, and IBM RS6000 workstations.

Sun Workstation

This section describes how to install XSI software on a Sun workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xsi_4ke/xsi_sn.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=( /correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

HP Workstation

This section describes how to install XSI software on a Hewlett-Packard workstation. You need to extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/XSI_4KE/XSI_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path  
set path=(/correct_path/xpre1/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

IBM RS6000 Workstation

This section describes how to install XSI software on an IBM RS6000 workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xsi_4ke/xsi_rs.tar
```


3. Set the path to `/correct_path/xpre1`;

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=(/correct_path/xpre1/bin/rs6000 $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing Mentor

This section describes how to install Mentor software on Sun and HP workstations.

Sun Workstation

This section describes how to install Mentor8 software on a Sun workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/mn8_sun4/mn8_sn.tar
tar -xf cd_rom_path/xact_4ke/xact_sn.tar
```

3. Set the path to `/correct_path/xpre1`;

```
setenv LCA /correct_path/xpre1
setenv XACT /correct_path/xpre1:existing_xact_path
set path=($LCA/com/sparc $LCA/bin/sparc \
/correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file has occupied.

HP Workstation

This section describes how to install XSI software on an HP workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/MN8_HPPA/MN8_HP.TAR\;1
tar -xf cd_rom_path/XACT_4KE/XACT_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`:

```
setenv LCA /correct_path/xpre1
setenv XACT /correct_path/xpre1:existing_xact_path
set path=($LCA/com/hppa $LCA/bin/hppa \
/correct_path/xpre/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing Core Tools

This section describes how to install the core tools software on Sun, HP, and IBM RS6000 workstations.

Sun Workstation

This section describes how to install the core tools only on a Sun workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xact_4ke/xact_sn.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path  
set path=(/correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

HP Workstation

This section describes how to install the core tools only on a Hewlett-Packard workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/XACT_4KE/XACT_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path  
set path=(correct_path/xpre1/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file has occupied.

IBM RS6000 Workstation

This section describes how to install the core tools only on an IBM RS6000 workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xact_4ke/xact_rs.tar
```

3. Set the path to `/correct_path/xpre1`;

```
setenv XACT /correct_path/xpre1:existing_xact_path
```

```
set path=(correct_path/xpre1/bin/rs6000 $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing for Use with XACTstep 6.0

You can use the XC4000E software within the XACTstep 6.0 Windows environment on the PC.

Files Required

Install (copy) the Viewlogic XC4000E software and the XC4000E core tools and data files from the CD-ROM into a directory called `4kexact`. In the following instructions, drive `e:` represents the CD-ROM drive on your PC, and `c:` represents the PC hard disk. Then, to copy the appropriate files from the CD-ROM into the `\4kexact` directory, execute the following instructions:

```
mkdir c:\4kexact
cd c:\4kexact
xcopy e:\vl_pc\*. * /s
xcopy e:\4ke_pc\*. * /s
```

The XC4000E software is installed in a separate XACT tree so that you can maintain two XACT trees, one containing the XACTstep 6.0 software and data files, and the other the XC4000E software and data files.

In order to use the XC4000E device in conjunction with other Xilinx devices, a new `partlist.xct` file has been created. This `partlist.xct` file is available on the Xilinx Bulletin Board, as `4keptlst.zip`. This file contains the new `partlist.xct` file. Uncompress the zip file and copy

the partlist.xct to your C:\4KEXACT\DATA directory. You will overwrite the existing partlist.xct in C:\4KEXACT\DATA. Do not overwrite the partlist.xct file that exists in C:\XACT\DATA.

Assuming that drive b: represents the floppy drive on your PC, execute the following instructions:

1. Change drives to the b: drive.
2. Uncompress the ZIP file by typing the following:

```
pkunzip b:\4keptlst.zip
```
3. Copy the partlist.xct into your c:\4kexact\data directory. (You will overwrite the existing partlist.xct in c:\4kexact\data). Do not overwrite the partlist.xct file in c:\xact\data.

The partlist.xct file *must* reside in a writable directory; therefore, do not access it from the network.

Modifying Your Setup

You must modify the XACT environment variable and the path in your autoexec.bat file.

XACT Environment Variable

The XACT environment variable should point to at least three locations. If you have installed the XACTstep 6.0 software in c:\xact and the XC4000E software in c:\4kexact and the PRO Series software in c:\proser, verify that you have the following line in your autoexec.bat or setup file:

```
set xact=C:\4KEXACT;C:\XACT;C:\PROSER
```

The \4kexact directory should appear before the \xact directory.

Path

The path in your autoexec.bat (or setup) file should include all the directories containing Xilinx software.

```
PATH C:\4KEXACT;C:\XACT;C:\PROSER;rest_of_your_path
```

The directory containing the XC4000E software should appear before the XC4000 \xact directory.

Using the XC4000E Software Within XACTstep 6.0

This section lists the known issues in using the XC4000E software within the XACTstep 6.0 environment.

- You cannot select the XC4000E family from the Design Entry tools.

Select the XC4000 family instead.

- The Library List Editor always lists the XC4000 library instead of the XC4000E library.

Follow these steps to resolve this problem:

- a) In the Design Entry window, invoke the Library List Editor, select the XC4000 library from Current Libraries, and modify it to XC4000E.
- b) Now select **Add/Change** (the Replace command does not effect the desired action) to include the XC4000E libraries.
- c) Select the XC4000 library and select **Remove**.

This last step is important because the Library List Editor otherwise reverts to using the XC4000 library.

- An error message such as “Could not find xc4000.bos” is issued.

This message is issued if the XACTstep 6.0 software is unable to find all the files required in order to successfully process XC4000E designs. It is not enough to copy only the partlist.xct and device package files. Make sure that all the files have been copied over from the CD-ROM and that the path and the environment variable have been set correctly.

- The Library List Editor reverts to listing the XC4000 libraries instead of the XC4000E library after you exit Windows and re-enter the Design Manager.

Either re-invoke the Library List Editor and modify the library listing, or as a more permanent solution, hard-code the view-draw.ini file in the project directory so that the XC4000E library is always enabled instead of the XC4000 library.

- Simulation may fail because XNF2WIR fails with Error 210.

This error can be resolved by editing a file called runtsim.bat, which is a script file that runs timing simulation. This file is

found in c:\proser\standard\runtsim.bat. Edit line 4 of runtsim.bat to delete the -l option, with which XNF2WIR is invoked. Now XNF2WIR is executed without the -l option.

The -l option of XNF2WIR forces aliases to be attached to the primitives. For example, if XNF2WIR processes an IBUF, the -l option causes it to select the IBUF from the XC4000 library. Since the XC4000 library is not listed in the viewdraw.ini file, an error results. Removing the -l option allows XNF2WIR to select the IBUF from any library. Since only one device library is specified, XNF2WIR correctly chooses the XC4000E IBUF.

- When you invoke PROsim to perform a timing simulation, it should automatically invoke PROwave on completion of the simulation. However, it does not invoke PROwave when you use the XC4000E family.

When PROsim completes, you must manually invoke PROwave to view the waveform. Select the *design.wfm* file and read the waveform display.

- The Floorplanner erroneously allows some XC4000E placements but causes PPR to issue errors.

For example, the Floorplanner allows you to place a configuration composed of a RAM requiring a write clock and an additional clocked flip-flop in the same CLB. This configuration is allowed if the register is clocked with the write clock *only*. Two different clocks cannot be placed in the same CLB. The Floorplanner allows this configuration without reporting errors, but PPR crashes with errors.

The Floorplanner allows two dual-port RAMs to be placed in one CLB, but such placement causes an error because only one dual-port RAM can be placed in a CLB.

- The Floorplanner may issue errors on legal placements by PPR.

After PPR places and routes a design, reading back the routed LCA file may result in errors when you execute a Check Floorplan command. The error message indicates that some CLBs have H function generators in bad configurations. This message is not a valid error message; the Floorplanner writes a valid constraints (CST) file that PPR will accept. Make sure that new errors are not introduced when changing the design layout.

Features in This Release

The XC4000E Pre-Release contains all of the necessary features to support XC4000E designs, but it is not intended for use with other architectures. Please see the “Installation” section for instructions on installing this software. A full explanation of the XC4000E architecture is found in the *XC4000E Field Programmable Gate Array Family Data Sheet*.

The XC4000E design flow is the same as the XC4000 design flow.

XC4000E Viewlogic

This section describes the Viewlogic library and simulation support for the XC4000E architecture. See the *XC4000E Libraries Guide Supplement* for details.

XC4000E Library

The XC4000E library is a superset of the XC4000 library, with the addition of XC4000E-specific elements. To access the XC4000E library, follow these platform-specific instructions.

PC

To access the XC4000E library on a personal computer, perform the following steps.

1. Include this line in your viewdraw.ini file:

```
DIR [m] d:\correct_path\unified\xc4000e (xc4000e)
```

where d: is the drive on which the XC4000E Pre-Release software is installed.

For example, if the software is installed in d:\pre\xact, your viewdraw.ini file should contain the following lines:

```
DIR [m] d:\PRE\XACT\UNIFIED\XC4000E (xc4000e)
DIR [m] d:\PRE\XACT\UNIFIED\XBLOX (xblox)
DIR [m] d:\PRE\XACT\UNIFIED\XBUILTIN (xbuiltin)
DIR [m] d:\PRE\XACT\UNIFIED\BUILTIN (builtin)
```

2. Include the path to the \unified directory in your XACT environment variable:

```
set XACT=d:\PRE\XACT
```

Sun4, HP, and IBM RS6000

To access the XC4000E library on a Sun4, HP, or IBM RS6000 workstation, include this line in your viewdraw.ini file:

```
DIR [r] /correct_path/unified/xc4000e(xc4000e)
```

For example, if the software is installed in /customer/pre/xilinx, your viewdraw.ini file should contain the following lines:

```
DIR [r] /customer/pre/xilinx/unified/xc4000e \
(xc4000e)
DIR [r] /customer/pre/xilinx/unified/xblox (xblox)
DIR [r] /customer/pre/xilinx/unified/xbuiltin \
(xbuiltin)
DIR [r] /customer/pre/xilinx/unified/builtin \
(builtin)
```

FXC4000E Library

The FXC4000E library is the same as the XC4000E library with one difference: the simulation models in the FXC4000E library do not perform timing violation checks like setup, hold, and minimum pulse width. The “f” prefix stands for fast simulation because the removal of timing violation checks can speed up the simulation. To access the FXC4000E library, follow these platform-specific instructions.

Refer to the FXC4000E library using the XC4000E alias. You should not have both the XC4000E and the FXC4000E libraries in one viewdraw.ini file.

PC

To access the FXC4000E library on a personal computer, perform the following steps.

1. Include this line in your viewdraw.ini file:

```
DIR [m] d:\correct_path\unified\fxc4000e (xc4000e)
```

where d: is the drive on which the XC4000E Pre-Release software is installed.

For example, if the software is installed in d:\pre\xact, your viewdraw.ini file should contain the following lines:

```
dir [m] d:\pre\xact\unified\fxc4000e (xc4000e)
dir [m] d:\pre\xact\unified\xblox (xblox)
dir [m] d:\pre\xact\unified\xbuiltin (xbuiltin)
dir [m] d:\pre\xact\unified\builtin (builtin)
```

2. Include the path to the \unified directory in your XACT environment variable.

```
set XACT = d:\PRE\XACT
```

Sun4, HP, and IBM RS6000

To access the FXC4000E library on a Sun4, HP, or IBM RS6000 workstation, include this line in your viewdraw.ini file:

```
DIR [r] /correct_path/unified/fxc4000e(xc4000e)
```

For example, if the software is installed in /customer/pre/xilinx, your viewdraw.ini file should contain the following lines:

```
DIR [r] /customer/pre/xilinx/unified/fxc4000e \
(xc4000e)
DIR [r] /customer/pre/xilinx/unified/xblox (xblox)
DIR [r] /customer/pre/xilinx/unified/xbuiltin \
(xbuiltin)
DIR [r] /customer/pre/xilinx/unified/builtin \
(builtin)
```

Switching to XC4000E from XC4000

If you have an existing Viewlogic schematic design implemented with the XC4000 library and you wish to switch to the XC4000E library, you can do so by changing your viewdraw.ini file, as illustrated in the following examples.

Example 1: On a workstation, if the XC4000E software is installed in /customer/pre/xilinx and the XC4000 software is installed in /customer/old/xilinx, you can change the following line:

```
DIR [r] /customer/old/xilinx/unified/xc4000 (xc4000)
```

to:

```
DIR [r] /customer/pre/xilinx/unified/xc4000e \
(xc4000e)
DIR [r] /customer/pre/xilinx/unified/xc4000e (xc4000)
```

Example 2: On the PC, if the XC4000E software is installed in d:\pre\xact and the XC4000 software is installed in d:\old\xact, you can change the line:

```
DIR [m] d:\OLD\XACT\UNIFIED\XC4000 (xc4000)
```

to:

```
DIR [m] d:\PRE\XACT\UNIFIED\XC4000E (xc4000e)
DIR [m] d:\PRE\XACT\UNIFIED\XC4000E (xc4000)
```

New Symbols

Following are the new XC4000E symbols:

IFDX	IFDXI	IFDX4	IFDX8	IFDX16
IFDX_1	IFDXI_1			
ILDX	ILDXI	ILDX4	ILDX8	ILDX16
ILDX_1	ILDXI_1			
OFDX	OFDXI	OFDX4	OFDX8	OFDX16
OFDX_1	OFDXI_1			
OFDEX	OFDEXI	OFDEX4	OFDEX8	OFDEX16
OFDEX_1	OFDEXI_1			
OFDTX	OFDTXI	OFDTX4	OFDTX8	OFDTX16
OFDTX_1	OFDTXI_1			
RAM16X1D	RAM16X2D	RAM16X4D	RAM16X8D	
RAM16X1S	RAM16X2S	RAM16X4S	RAM16X8S	
RAM32X1S	RAM32X2S	RAM32X4S	RAM32X8S	

The X-BLOX XC4000E RAM symbols are DP_RAM and SYNC_RAM.

Location Constraints

In general, any parameter that is allowed on RAM16X1 or RAM32X1 for XC4000 is allowed on RAM16X1S, RAM32X1S, or RAM16X1D for

XC4000E. As an exception, when an RLOC attribute is used on a RAM16X1D symbol, no extension is allowed, because a RAM16X1D occupies an entire CLB.

Simulating XC4000E RAM

An INIT value placed on a RAM16X1D, RAM16X1S, or RAM32X1S primitive must be processed by XSimMake to be shown in simulation. The XSimMake functional flow creates a file named *sdesign.xmm*, which contains the appropriate ViewSim commands to pre-load the RAM elements. This command file should be executed in ViewSim at the start of both functional and timing simulation. XSimMake also adds commands to the XMM file for asynchronous RAMs, ROMs, and memories created by X-BLOX and MemGen.

XC4000E OrCAD

This section describes the OrCAD library and simulation support for the XC4000E architecture. See the *XC4000E Libraries Guide Supplement* for details.

XC4000E Library

The XC4000E library is a superset of the XC4000 library, with the addition of XC4000E-specific elements. To access the XC4000E library for both schematic and simulation, perform the following steps.

1. Set the XACT environment variable to point to the installed directory of the Xilinx DS35 OrCAD interface.
2. Create the design directory in either of the following two ways:
 - Run the Create Design command from the Design Management Tools in the OrCAD ESP.
 - Create a new directory, then copy into it the OrCAD SDT/VST configuration files (*sdt.cfg* and *vst.cfg*, respectively) from the OrCAD template directory.
3. In the newly created directory, run the Xdraft 4e command from the MS-DOS command line. This command configures both *sdt.cfg* and *vst.cfg* to use the XC4000E libraries for schematic and simulation.

New Symbols

Following are the new XC4000E symbols:

IFDX	IFDXI	IFDX4	IFDX8	IFDX16
IFDX_1	IFDXI_1			
ILDX	ILDXI	ILDX4	ILDX8	ILDX16
ILDX_1	ILDXI_1			
OFDX	OFDXI	OFDX4	OFDX8	OFDX16
OFDX_1	OFDXI_1			
OFDEX	OFDEXI	OFDEX4	OFDEX8	OFDEX16
OFDEX_1	OFDEXI_1			
OFDTX	OFDTXI	OFDTX4	OFDTX8	OFDTX16
OFDTX_1	OFDTXI_1			
RAM16X1D	RAM16X2D	RAM16X4D	RAM16X8D	
RAM16X1S	RAM16X2S	RAM16X4S	RAM16X8S	
RAM32X1S	RAM32X2S	RAM32X4S	RAM32X8S	

The X-BLOX XC4000E RAM symbols are DP_RAM and SYNC_RAM.

XC4000E MemGen

This section describes the new types that have been added to the MemGen program to support the synchronous and dual-port RAMs in the XC4000E architecture. It also provides the pin names of RAM components for instantiating cells in the HDL source code.

SYNC_RAM and DP_RAM

Two new TYPE values, SYNC_RAM and DP_RAM, have been added to the MemGen program. You can specify TYPE values interactively, through the command line, or in the MEM file.

- From the command line, enter the following:

```
type=[rom|ram|sync_ram|dp_ram]
```

Use SYNC_RAM for a single-port, edge-triggered RAM and DP_RAM for a dual-port, edge-triggered RAM.

Here is an example:

```
memgen ram64x1s type=sync_ram memory_depth=64  
word_width=1
```

- In interactive mode, you see the following:
What type of memory do you want to build?
Enter one of the following:
1 for RAMs
2 for ROMs
3 for SYNC_RAMs (XC4000E only)
4 for DP_RAM (XC4000E only)
- In the MEM file, type the following syntax:
`type [rom|ram|sync_ram|dp_ram]`
A sample MEM file illustrates this syntax:

```
TYPE SYNC_RAM ; The memory is a SYNC_RAM
DEPTH 64 ; The memory is 64-word deep
WIDTH 1 ; Each memory word is 1-bit wide
;
PART 4005EPG156
;
SYMBOL VIEWLOGIC PINS; Build a VIEWLOGIC symbol with
pin inputs
;
DEFAULT 0 ; Add a default value for unspecified
locations
DATA 0 ; Add your SYNC_RAM data here
;
```

Default and Data Commands

In the past, the Default and Data commands were only permitted on ROMs. Now they also apply to the XC4000E RAMs, SYNC_RAMs, and DP_RAMs. They are added to support the INIT attribute on all XC4000E RAMs.

Symbols

MemGen generates symbols for the new RAM types, SYNC_RAM and DP_RAM. To generate a symbol file for OrCAD, use the -o option. To generate a symbol file for Viewlogic, use the -v option.

Pin Names of RAM Components

This section describes how MemGen generates the pin names for RAM components. This information explains how pins are instantiated. The pin order is also important; that is, it should match the order that appears in the netlist.

1. MemGen generates pin names for XC4000/XC4000E RAMs from the following user inputs:

Address lines: An

Data lines: Dn

Output lines: On

Write-Enable line: WE

2. MemGen generates pin names for XC4000E SYNC_RAMs from the following user inputs:

Address lines: An

Data lines: Dn

Output lines: On

Write-Enable line: WE

Clock line: WCLK

3. MemGen generates pin names for XC4000E DP_RAMs from the following user inputs:

Address lines: An

Data lines: Dn

DP read address lines: DPRAn

SP Output lines: SPOn

DP output lines: DPOn

Write-Enable line: WE

Clock line: WCLK

XC4000E X-BLOX

This section describes the new TYPE values that have been added to the XBLOX program to support the synchronous and dual-port RAMs in the XC4000E.

Two new modules have been added to X-BLOX, SYNC_RAM and DP_RAM.

SYNC_RAM

The new SYNC_RAM module synthesizes a synchronous read-write static random-access memory.

The inputs, outputs, and attributes are identical to the current SRAM module with the following exceptions.

- A positive edge on WR_CLK outputs data selected by ADDR on D_OUT, and when WR_EN is active, it loads data on D_IN into the RAM. The WR_CLK pin must be connected.
- ADDR_ERROR is not supported.
- The INIT and MEMFILE attributes have been added. See the “Specifying the Initial Contents of an SRAM, SYNC_RAM, or DP_RAM” section later in this document.
- The value of DEPTH must be a multiple of 16.
- CLB utilization is the same as that of the SRAM in XC4000.

DP_RAM

The DP_RAM module synthesizes a synchronous dual-port read-write static random-access memory.

The inputs, outputs, and attributes are identical to the current SRAM module with the following exceptions.

- The following inputs have been added.
 - WR_CLK

A positive edge on WR_CLK outputs data selected by ADDR on SP_OUT, and when WR_EN is active, it loads data on D_IN into the RAM. The WR_CLK pin must be connected.

- DPRD_ADDR

The DPRD_ADDR (dual-port read address) port selects the word that appears on DP_OUT. DPRD_ADDR cannot be connected to a bus with ENCODING=ONE-HOT.

- The following outputs have been added or changed.

- SP_OUT

The SP_OUT port reflects the addressed word of the RAM specified by ADDR when WR_CLK goes active.

- DP_OUT

The DP_OUT port reflects the addressed word of the RAM specified by DPRD_ADDR.

- ADDR_ERROR is not supported.

- The INIT and the MEMFILE attributes have been added. See the “Specifying the Initial Contents of an SRAM, SYNC_RAM, or DP_RAM” section, following.

- The value of DEPTH must be a multiple of 16.

- CLB utilization is twice that of the SRAM in XC4000.

Conditions for Implementation in an IOB

Item 2 on page 4-53 of the *X-BLOX Reference/User Guide* states that the flip-flop does not use its clock-enable pin. This statement is not valid for XC4000E, because clock enable on IOB registers is supported.

XC4000E XDE

This section describes the new tag values that have been added to the XDE program to support the synchronous and dual-port RAMs, the H-MUXes, and the IOB clock enable. It also describes the new SetMemory command, which sets the initial values on RAMs; two options in MakeBits to set TTL/COMS thresholds; and two new options for the start-up sequence.

Synchronous and Dual-Port RAMs

This release offers a complete set of EDITBLK RAM tag values:
RAM:DP:K:NOT:F:G:FG.

- RAM:DP indicates a dual-port RAM.
- RAM:K indicates a synchronous RAM.
- RAM:NOT indicates that the K pin is inverted.

However, the new RAM:DP tag value and the existing RAM:FG tag value are mutually exclusive.

Allowed values for XC4000E RAM tag are shown in the following table.

RAM:F:G	Two 16x1 non-synchronous RAMs
RAM:F:G:K	Two 16x1 synchronous RAMs
RAM:F:G:K:NOT	Two 16x1 synchronous RAMs, inverted clock
RAM:F	One 16x1 non-synchronous RAM
RAM:G	One 16x1 non-synchronous RAM
RAM:FG	One 32x1 non-synchronous RAM
RAM:F:K	One 16x1 synchronous RAM
RAM:G:K	One 16x1 synchronous RAM
RAM:FG:K	One 32x1 synchronous RAM
RAM:F:K:NOT	One 16x1 synchronous RAM, inverted clock
RAM:G:K:NOT	One 16x1 synchronous RAM, inverted clock
RAM:FG:K:NOT	One 32x1 synchronous RAM, inverted clock
RAM:F:G:DP:K	One 16x1 synchronous, dual-port RAM
RAM:F:G:DP:K:NOT	One 16x1 synchronous, dual-port RAM, inverted clock

SetMemory Command Initializes All XC4000E RAMs

RAM initialization is implemented by the SetMemory command from the Config menu. The SetMemory command is allowed only

inside the block editor. The following table describes the values for the SetMemory options.

Setmemory F G FG value	Non-dual-port RAMs or ROMs. For F and G (16x1 case), a 16-bit value is expected. For FG (32x1), a 32-bit value is expected. G is in the left two bytes of the 32-bit number, and F is in the right two bytes.
Setmemory value	Dual-port RAMs. A 16-bit value is expected and loaded into F and G.

The *value* is a hexadecimal number with a leading “x” character.

In both cases, the MSB of the RAM initial value goes into the high RAM address, as in the following examples.

SETMEMORY F x8000 sets bit 15 of F RAM to 1

SETMEMORY G x0001 sets bit 0 of G RAM to 1.

H Multiplexers

This release offers a complete set of EDITBLK H function tag values: H:F:G:H1:DIN:SR.

- H:DIN indicates that the DIN pin sources the H function generator.
- H:SR indicates that the SR pin sources the H function generator.

However, the new H:DIN tag value and the existing H:F tag value are mutually exclusive. The new H:SR tag value and the existing H:G tag value are mutually exclusive.

Allowed values for XC4000E RAM tag are the following:

H:G	H:DIN:G	H:H1:F:G
H:SR	H:DIN:SR	H:H1:F:SR
H:F	H:H1	H:H1:DIN
H:F:G	H:H1:G	H:H1:DIN:G
H:F:SR	H:H1:SR	H:H1:DIN:SR
H:DIN	H:H1:F	

The H equation (Equate H) for XC4000E takes DIN and SR as arguments. Here are three examples:

```
Equate H = SR  
Equate H = DIN  
Equate H = (SR*DIN*H1)
```

IOB Clock Enable

The new EDITBLK IOB clock-enable tag values for XC4000E are INFF:EC and OUT:EC.

- INFF:EC indicates that the input register has a clock-enable pin.
- OUT:EC indicates that the output register has a clock-enable pin.

IOB O/EC Routing Model

The XC4000E parts now support a clock-enable function on each IOB that is common to both IOB flip-flops. This function is represented as the new EC pin on the IOB. The interconnect that connects to EC is shared between the EC pin and the existing O pin; this sharing results in some limitations in the use of the interconnect when both EC and O are used on the same IOB. In the Design Editor (XDE), the interconnect attached to the O pin of an IOB is shown with two forks. One fork wraps around the IOB parallel to the interconnect attached to the EC pin and matches it PIP for PIP. The second fork extends into the XC4000E device.

When the IOB clock enable is not in use, any of the PIPs on the O-pin interconnect can be enabled to drive a signal to the O pin.

When both the O pin and the EC pin are in use, the PIPs on the O interconnect fork that match those on the EC interconnect cannot be used to drive a signal to the O pin. They can only connect a net to the EC pin. The only PIPs available to drive a signal onto the O pin are those on the fork of interconnect that extends into the device. A DRC check ensures the correct use of the interconnect in this condition.

XC4000 to XC4000E Conversion

To convert an XC4000 LCA file into an XC4000E LCA file, select the Convert command from the Programs menu, then follow the prompts.

An existing XC4000A/D/H LCA file cannot be converted into an XC4000E LCA file.

CMOS Thresholds

The XC4000E Pre-Release allows you to specify the input thresholds for all IOBs on the device by using the MakeBits command from the Programs menu in XDE, as follows:

- INPUT CMOS sets all input thresholds to CMOS.
- INPUT TTL sets all input thresholds to TTL. It is the default.

The XC4000E allows you to specify the output thresholds for all IOBs on the device by using the MakeBits command from the Programs menu in XDE, as follows:

- OUTPUT CMOS sets all output thresholds to CMOS.
- OUTPUT TTL sets all output thresholds to TTL. It is the default.

Note: For additional information see the “MakeBits Should Configure TTL Inputs by Default” item in the “Known Issues” chapter of this release note.

Start-Up Sequence Options

These new options are available with the start-up sequences for the XC4000E.

- M0 Pin
Setting: Pullup, Pulldown
Default: Neither
- M2 Pin
Settings: Pullup, Pulldown
Default: Neither

Note: The Pullup and Pulldown options for the M0 and M2 pins act as toggles and are mutually exclusive. The default is inactive. Selecting one option enables it and disables the other. Selecting the same option a second time disables it.

XC4000E PPR

The XC4000E Pre-Release supports both guided design and XACT-Performance in PPR.

Guided Design

PPR supports the guiding feature for XC4000 or XC4000E designs guiding XC4000E designs in this release. See the *XACTstep Development System Reference Guide*, Volume 2, for information about guided design.

XC4000E XACT-Performance Support

XACT-Performance support for the XC4000E family is mostly unchanged from its behavior for the XC4000 family, but there are new varieties of RAM to consider.

All grouping mechanisms that apply to RAMs in the XC4000 family also apply to RAMs in the XC4000E family. The RAMS qualifier matches any type of RAM, and the predefined RAMS group includes RAMs of all types.

See the “XACT-Performance Utility” chapter in the *XACTstep Development Reference Guide*, Volume 1, for more information on grouping mechanisms.

For all RAM modes, path-tracing behavior is best considered according to where the RAM in question appears in the paths associated with a particular TIMESPEC:

- Type 1: RAM at the end of paths, for example, "from:others:to:rams"
- Type 2: RAM at the start of paths; for example, "from:rams:to:others"
- Type 3: RAM in the middle of paths, for example, an enclosing "from:ffs:to:ffs"

Asynchronous RAMs

For RAMs configured in asynchronous mode, path-tracing behavior is unchanged from the XC4000.

- Type 1 paths (to RAM) are always traced. PPR determines the setup time appropriate to the destination pin.
- Type 2 paths (from RAM) are always traced. PPR determines the worst-case time from a change on D or WE to data valid.

- Type 3 paths (through RAM) are always traced if they arrive at address pins, and they are not traced if they arrive at D or WE pins. Changes on address inputs propagate just as they do for ordinary look-up tables. However, propagation of changes on D or WE is assumed to be of interest only when the RAM is being read during a write operation. If you want PPR to control the delay on paths through the D or WE inputs, you must split the delay requirement into two segments: one ending at the RAM input pin, and the other beginning at the RAM output.

Single-Port Synchronous RAMs

Path-tracing for synchronous RAMs is essentially the same as for asynchronous RAMs.

- Type 1 paths (to RAM) are always traced. PPR determines setup times with respect to the WCLK pin, for all other inputs.
- Type 2 paths (from RAM) are always traced. PPR determines worst-case time after WCLK transition to data valid.
- Type 3 paths (through RAM) are traced through address pins only.

Dual-Port Synchronous RAMs

Tracing behavior for dual-port synchronous RAMs is as follows.

- Type 1 paths (to RAM) are traced to every input pin destination except DPRA0 to DPRA3. Read address inputs cannot impact paths that end at a RAM (write function).
- Type 2 paths (from RAM) are always traced.
- Type 3 paths (through RAM) are traced through address inputs only. In particular, there are A?-to-SP0 paths and DPRA?-to-DPO paths.

XC4000E XSI

The XC4000E Pre-Release of XSI is a superset of the existing XSI 5.2 product. The principal changes include the following:

- Addition of several new cells to the FPGA Compiler/Design Compiler synthesis libraries and the VSS FTGS simulation libraries

- Additional timing information to support new XC4000E speed grades
- XSI executables that now recognize new XC4000E part types
- Equal application of all other XC4000-specific optimizations performed by FPGA-Compiler to the XC4000E architecture, including:
 - Direct synthesis to XC4000 IOB and CLB structures
 - Automatic inference of X-BLOX modules for optimized arithmetic
 - Clock-buffer insertion
 - Synthesis to flip-flops with clock enables

Configuring Synopsys to Synthesize and Simulate XC4000E Designs

The .synopsys_dc.setup file for XC4000E synthesis should be identical to that for XC4000 designs with the exception of the target- and link-library settings and the reference to the X-BLOX Design Ware library. An example .synopsys_dc.setup file follows. The target- and link-library settings were created using the Synlibs 4005e-3 command.

```
search_path = { . \
<XC4000E_DS401_install_path>/synopsys/libraries/syn
\ <Synopsys_install_path>/libraries/syn}

define_design_lib xblox_4000e -path \
<XC4000E_DS401_install_path>/synopsys/libraries/dw/
lib/fpga/xc4000e

compile_fix_multiple_port_nets = true
xlnx_hier_blknm = 1
xnfout_library_version = "2.0.0"

bus_naming_style = "%s<%d>"
bus_dimension_separator_style = "><"
bus_inference_style = "%s<%d>"

link_library = {xprim_4005e-3.db xprim_4000e-3.db \
xgen_4000e.db xfpga_4000e-3.db xio_4000e-3.db}
target_library = {xprim_4005e-3.db xprim_4000e-3.db \
xgen_4000e.db xfpga_4000e-3.db xio_4000e-3.db}
```

```
symbol_library = {xc4000e.sdb}
synthetic_library = {xblox_4000e.sldb standard.sldb}
```

Notice that the reference to the XC4000E X-BLOX Design Ware library is differentiated from the XC4000 X-BLOX library with a new name, `xblox_4000e`.

The `.synopsys_vss.setup` file for XC4000E simulation should also be identical to that for XC4000 designs with the exception of the simulation library reference. The following is a sample `.synopsys_vss.setup` file.

```
timebase=ns
time_res_factor=0.1
no_hazard_mesg=true
WORK > DEFAULT
DEFAULT : ./WORK
xc4000e:<XC4000E_DS401_install_path>/synopsys/
libraries/sim/lib/xc4000e
```

New Components in XC4000E Synthesis and Simulation Libraries

Several new components have been added to the XC4000E synthesis and simulation libraries, reflecting the XC4000E architecture's new features. The new components, listed in the following tables, include synchronous single-port RAMs, synchronous dual-port RAMs, and I/O flip-flops/latches with clock/latch-enables. Synopsys does not synthesize cells marked with an asterisk, so these components must be instantiated.

Synchronous RAM			
Name	Outputs	Inputs	Notes
RAM16X1S	O	D, A3, A2, A1, A0, WE, WCLK	*
RAM32X1S	O	D, A4, A3, A2, A1, A0, WE, WCLK	*
RAM16X1D	SPO,DPO	D, A3, A2, A1, A0, DPRA3, DPRA2, DPRA1, DPRA0, WE, WCLK	*

IOB Input Flip-Flops with Clock Enable			
Name	Output	Inputs	Notes
IFDX	Q	D, C, CE	
IFDX_F	Q	D, C, CE	NODELAY*
IFDX_U	Q	D, C, CE	UNBONDED*
IFDXI	Q	D, C, CE	INIT=S
IFDXI_F	Q	D, C, CE	INIT=S, NODELAY*
IFDXI_U	Q	D, C, CE	INIT=S, NODELAY, UNBONDED*

IOB Output Flip-Flops with Clock Enable			
Name	Outputs	Inputs	Notes
OFDX	Q	D, C, CE	
OFDX_F	Q	D, C, CE	Fast slew rate
OFDX_FU	Q	D, C, CE	Fast slew rate, unbonded*
OFDX_S	Q	D, C, CE	Slow slew rate
OFDX_U	Q	D, C, CE	Unbonded*
OFDXI	Q	D, C, CE	INIT=S
OFDXI_F	Q	D, C, CE	INIT=S, fast slew rate
OFDXI_S	Q	D, C, CE	INIT=S, slow slew rate
OFDXI_U	Q	D, C, CE	INIT=S, unbonded*

IOB Tristatable Output Flip-Flops with Clock Enable			
Name	Outputs	Inputs	Notes
OFDTX	O	D, C, CE, T	
OFDTX_F	O	D, C, CE, T	Fast slew rate
OFDTX_S	O	D, C, CE, T	Slow slew rate
OFDTX_U	O	D, C, CE, T	Unbonded*

IOB Tristatable Output Flip-Flops with Clock Enable			
Name	Outputs	Inputs	Notes
OFDTXI	O	D, C, CE, T	INIT=S
OFDTXI_F	O	D, C, CE, T	INIT=S, fast slew rate
OFDTXI_S	O	D, C, CE, T	INIT=S, slow slew rate
OFDTXI_U	O	D, C, CE, T	INIT=S, unbonded*

IOB Input Latches with Gate Enable			
Name	Outputs	Inputs	Notes
ILDX_1	Q	D, G, GE	
ILDX_1F	Q	D, G, GE	NODELAY*
ILDX_1U	Q	D, G, GE	Unbonded*
ILDXI_1	Q	D, G, GE	INIT=S
ILDXI_1F	Q	D, G, GE	INIT=S, NODELAY
ILDXI_1U	Q	D, G, GE	INIT=S, unbonded*

Note: All IOB input gate-enabled latches have active-Low gate pins.

Current Restrictions Imposed By Synopsys 3.3

The Synopsys FPGA Compiler and Design Compiler products are currently unable to infer IOB registers with clock enables. The only way to access these features is to instantiate the appropriate cells in the HDL source code. This limitation will be addressed in a subsequent release of Synopsys' products; however, in the meantime, one of the optimizations performed by X-BLOX is to push clock-enabled CLB registers into IOBs where possible. Refer to the *XACTstep X-BLOX Reference/User Guide* for information about the rules governing this process.

Using Synchronous and/or Level-Sensitive RAM

The Synopsys FPGA Compiler and Design Compiler products are currently unable to infer RAM. (Although RAM can be described behaviorally, this methodology currently synthesizes to inefficient latch- or register-based implementations.) The only way to access these features is to instantiate the appropriate cells in the HDL source code. Alternatively, you can instantiate memory modules created by MemGen, which is explained in the “Pin Names of RAM Components” section earlier in this release note.

The XC4000E RAM modules allow you to specify their contents at power-on. When RAM modules are instantiated directly in the HDL source code, you can enter initialization values for the RAM using the following command:

```
set_attribute instance_name xnf_init init_value
-type string
```

For 16-location RAMs, specify a 4-digit hexadecimal value for *init_value*. For 32-location RAMs, specify an 8-digit hexadecimal value.

Although this mechanism permits RAM-initialization information to be carried into the FPGA implementation tools for incorporation into the configuration bitstream, it does not simulate behaviorally. For behavioral simulation, a RAM's contents remain unknown until they are defined by valid write access. However, back-annotated functional or timing simulation reflects this RAM initialization information.

XC4000E MN8 Mentor Interface

The XC4000E Pre-Release of Mentor Interface (MN8) is a superset of the existing MN8 5.2 product. The major changes include the following:

- Addition of several new cells to the Xilinx library for Mentor Interface
- Changes in the scripts to support the new cells and features
- Single-Port Synchronous RAM. The two new RAM symbols (primitives) are the following:
 - RAM16X1S

- RAM32X1S

There is a new WCLK pin, which occupies the same location as WE pin did in XC4000. The WE pin is now placed above the D data pin.

- Six new RAM macros:

- RAM16X2S
- RAM16X4S
- RAM16X8S
- RAM32X2S
- RAM32X4S
- RAM32X8S

The WE pin is now placed above the D pin.

- Dual-port RAM. RAM16X1D is the new primitive.

The WE pin is now placed above the D pin.

- Three new macros:

- RAM16X2D
- RAM16X4D
- RAM16X8D

The WE pin is now placed above the D pin.

- CE on I/O registers. The IOBs of the XC4000E parts, which have a pin, the CE or GE pin, that is not available on other XC4000 parts. The Mentor primitives that are affected are the following:

- INFF
- INLAT
- INREG
- OUTFF
- OUTFFT

- Eight new primitives:

- IFDX

- IFDXI
- ILDX_1
- ILDXI_1
- OFDX
- OFDXI
- OFDTX
- OFDTXI

The difference between these and the existing I/O register primitives is the addition of a CE pin between D and C pins for the flip-flops and the addition of a GE pin between the D and G pins for the latches.

- Twenty-seven new I/O macros:
 - IFDX4, IFDX8, IFDX16, IFDX_1, IFDXI_1
 - ILDX, ILDX4, ILDX8, ILDX16, ILDXI
 - OFDX4, OFDX8, OFDX16, OFDX_1, OFDXI_1
 - OFDEX, OFDEX4, OFDEX8, OFDEX16, OFDEXI, OFDEX_1, OFDEXI_1
 - OFDTX4, OFDTX8, OFDTX16, OFDTX_1, OFDTXI_1
- A new non-invertible pin required on the I/O register primitives. The pin name is CE for flip-flops and GE for latches. The XNF primitive names remain the same as they are currently.
- The INIT parameter on all RAM, RAMS, and RAMD primitives.
- Back-annotation. The -m option of the XNFBA program generates an MBA file for post-route timing back-annotation into Mentor. For RAM16X1S, RAM32X1S, and RAM16X1D, delays on all pins but the WCLK pin are routing delays and are put “on-parent.” The WCLK pin has double delay: routing and block. The delay on the WCLK pin is the block delay. The delay on the BUF driving the WCLK pin is the routing delay. The “on-parent” delay is written with the “RISE_{pin_name}” format. The double delay is written with the “RISE_{pin_name}” format for the routing delay and the “RISE_{pin_name}” format for the block delay.

The following example shows how a RAM16X1S symbol is back-annotated.

Following is an example of a RAM16X1S before routing:

```
SYM, FIFO/RAMS/RAMSL/$1I92, RAMS, SCHNM=RAM16X1S,
INIT=0, LIBVER=2.0.0

PIN, O, O, DO7

PIN, WE, I, FIFO/CE_WR

PIN, D, I, FIFO/DIQ7

PIN, WCLK, I, CLK

PIN, A0, I, FIFO/A0

PIN, A1, I, FIFO/A1

PIN, A2, I, FIFO/A2

PIN, A3, I, FIFO/A3

END
```

The next example shows a RAM16X1S after routing (PPR, XDelay, LCA2XNF):

```
SYM, DO7, RAMS, LIBVER=2.0.0, SCHNM=RAM16X1S,
INIT=0000

PIN, O, O, DO7, 2.0

PIN, WE, I, FIFO/CE_WR, 5.0

PIN, D, I, FIFO/DIQ7, 1.1

PIN, WCLK, I, YSIG121, 6.0

PULSE, WCLK, +, 8.4

PIN, A3, I, FIFO/A3, 3.1

PIN, A2, I, FIFO/A0, 5.3

PIN, A1, I, FIFO/A2, 2.7

PIN, A0, I, FIFO/A1, 5.8

SETUP, D, WCLK, +, 3.3, 1.1

SETUP, WE, WCLK, +, 1.3, 0.4

SETUP, A3, WCLK, +, 2.4, 0.7

SETUP, A2, WCLK, +, 2.4, 0.7

SETUP, A1, WCLK, +, 2.4, 0.7
```



```
        SETUP, A0, WCLK, +, 2.4, 0.7
END
SYM, XSYM220, BUF, LIBVER=2.0.0
        PIN, O, O, YSIG121
        PIN, I, I, CLK, 1.4
END
```

Here is an example of a RAM16X1S after back-annotation with XNFBA, which produces an MBA file:

```
/FIFO/RAMS/RAMSL/$1I92 N RISEO 2.0
/FIFO/RAMS/RAMSL/$1I92 N FALLO 2.0
/FIFO/RAMS/RAMSL/$1I92 N RISEA0 5.3
/FIFO/RAMS/RAMSL/$1I92 N FALLA0 5.3
/FIFO/RAMS/RAMSL/$1I92 N RISEA1 5.8
/FIFO/RAMS/RAMSL/$1I92 N FALLA1 5.8
/FIFO/RAMS/RAMSL/$1I92 N RISEA2 2.7
/FIFO/RAMS/RAMSL/$1I92 N FALLA2 2.7
/FIFO/RAMS/RAMSL/$1I92 N RISEA3 3.1
/FIFO/RAMS/RAMSL/$1I92 N FALLA3 3.1
/FIFO/RAMS/RAMSL/$1I92 N RISED 1.1
/FIFO/RAMS/RAMSL/$1I92 N FALLD 1.1
/FIFO/RAMS/RAMSL/$1I92 N RISEWE 5.0
/FIFO/RAMS/RAMSL/$1I92 N FALLWE 5.0
/FIFO/RAMS/RAMSL/$1I92 N RISEWCLK 1.4
/FIFO/RAMS/RAMSL/$1I92 N FALLWCLK 1.4
/FIFO/RAMS/RAMSL/$1I92 N RISE_WCLK 6.0
/FIFO/RAMS/RAMSL/$1I92 N FALL_WCLK 6.0
/FIFO/RAMS/RAMSL/$1I92 N SETUPA1 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA1 0.7
/FIFO/RAMS/RAMSL/$1I92 N SETUPA2 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA2 0.7
/FIFO/RAMS/RAMSL/$1I92 N SETUPA0 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA0 0.7
```

```
/FIFO/RAMS/RAMSL/$1I92 N SETUPA3 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA3 0.7
/FIFO/RAMS/RAMSL/$1I92 N SETUPWE 1.3
/FIFO/RAMS/RAMSL/$1I92 N HOLDWE 0.4
/FIFO/RAMS/RAMSL/$1I92 N SETUPD 3.3
/FIFO/RAMS/RAMSL/$1I92 N HOLDD 1.1
/FIFO/RAMS/RAMSL/$1I92 N PULSEWCLK 8.4
/FIFO/RAMS/RAMSL/$1I92 S PULSE_POLARITY_WCLK +
/FIFO/RAMS/RAMSL/$1I92 S INIT 0
```

Script Changes

The Mentor scripts have been changed to support the new cells in the XC4000E Pre-Release. In addition to these changes, XBLXGS and EDIF2XNF have been modified to support the new XC4000E library.

MEN2XNF8, FNCSIM8, TIMSIM8

From the Design Manager (pld_dmgr), you can invoke commands to execute specific scripts, as follows.

- PLD_Men2XNF8 executes the script to translate your design to an XNF file. You must run this command before you can use either PLD_FNCSIM8 or PLD_TIMSIM8.
- PLD_FNCSIM8 executes the script to prepare your design for functional simulation.
- PLD_TIMSIM8 executes the script to prepare your design for timing simulation.

Refer to the *Mentor Graphics Interface/Tutorial Guide* for more details these commands.

PLD_DVE

This command executes the script that invokes the Mentor Graphics Design Viewpoint Editor (DVE) configured for Xilinx designs. The following example shows the Pld_dve command as you would use it in Men2XNF8 to check for the valid technology:

```
pld_dve t2 xc4000e
```

GEN_SCH8

This program creates a new schematic composed of only schematic elements that can be used for functional simulation. An example follows:

```
gen_sch8 xnf_file
```

Xnf_file is any XC4000E file; that is, its part starts with “xc4000e.”

The output should be a valid design. To verify its validity, bring the design into the Design Architect and run the Check Sheet command.

GEN_SYM8

This program automatically creates a symbol based on information in the XSF file. An example follows:

```
gen_sym8 xnf_file
```

Xnf_file is any XC 4000E file. The output should be a valid symbol. To verify its validity, bring the design into the Design Architect and examine the symbol.

Unsupported Features in This Release

The XC4000E Pre-Release does not support the following features.

XACT-Floorplanner™

The XACT-Floorplanner is not available in this release.

OrCAD

The OrCAD library does not support the RAM, RAMS, or RAMD initialization simulation in this release.

XDE

The Convert command does not support the conversion from a smaller XC4000E part to a larger part.

Chapter 4

Device and Package Support

The following is a master table of Xilinx devices for this release. For more information on architectural families and specific device parameters, see *The Programmable Logic Data Book*.

Device	Packages	Speed Grades
XC4005E	PC84, PG156, PQ100, PQ160, PQ208, TQ144	-3
XC4006E	PC84, PG156, PQ160, PQ208, TQ144	-3
XC4008E	PC84, PG191, PQ160, PQ208	-3
XC4010E	BG225, PC84, PG191, PQ160, PQ208	-3
XC4013E	BG225, HQ208, HQ240, PG223, PQ160, PQ208, PQ240	-3
XC4020E ^a	HQ208, HQ240, PG223	-3
XC4025E	HQ240, HQ304, PG223, PG299	-3

a. XC4020E speed file is advanced.

The files shown in the table pertain to the XC4000E device family only. The advanced speed files are derived from simulation models and will change as characterized data becomes available.

Known Issues

This chapter describes the known issues in this release.

Software

The issues in this section are listed in the order that they occur in the design process.

Design Entry

Inversion Bubble Is Missing for OrCAD I/O Primitives

Platform: All

Architecture: XC4000E

Design Step: Design Entry

Reference Number: 27552

The OFDTXI_1, OFDXI_1, ILDXI_1, IFDXI_1, OFDTI_1, and OFDI_1 I/O primitives are missing the inversion bubbles on the clock pin.

The OFDEI symbol has a bubble on the clock pin but should not.

In all these cases, the netlist is correct. The problem is only in the drawing.

Illegally Instantiating I/Os with XSI May Cause FPGA Compiler or Design Compiler to Crash

Platform: All
Architecture: XC4000E
Design Step: Synopsys FPGA Synthesis
Reference Number: 27629

Certain illegal actions may cause the FPGA Compiler or the Design Compiler to crash rather than to give an error message when you instantiate I/Os in XC4000E devices. For example, you might instantiate one or more I/O cells, such as an IFD, OFD, or OFDTX, and connect them to a port *and* include that port in the port list of the `set_port_is_pad` command.

This situation is illegal because the `set_port_is_pad` command tells the FPGA Compiler or Design Compiler to synthesize I/O logic for the indicated ports. If you have already instantiated the I/O logic for any of these ports, further I/O logic should not be synthesized.

Ordinarily, this situation would cause an error message to be issued, but in the current Synopsys release, 3.3b, the FPGA Compiler or Design Compiler may crash.

Implementation

PPR: Setting Ignore_rlocs to True May Cause Errors in LCA2XNF, MakeBits, XDelay

Platform: All
Architecture: XC4000E
Design Step: Implementation
Reference Number: 27556

If you set the PPR `Ignore_rlocs` option to True on a design containing dual-port RAM symbols, PPR may implement the RAM incorrectly, causing errors to be reported in LCA2XNF, MakeBits, or XDelay.

To avoid this problem, rerun the XNFPrep program with the `Ignore_rlocs=True` option. Then rerun PPR using `Ignore_rlocs=False`, which is the default.

XNFPrep removes RLOC constraints from the design, avoiding the problem in PPR.

PPR Issues ERROR 1582

Platform: All
Architecture: XC4000E
Design Step: Implementation
Reference Number: 27387

PPR may issue the following Internal Error when it partially routes a net that includes a clock enable pin on an IO block:

```
*** PPR: ERROR 1582:
    Error in writing LCA data to memory:
    Pin already exists in memory
    Data: "PADnn.O"
This is an internal error; please contact Xilinx
support personnel.
```

PPR writes a duplicate Addnet or Addpin record for the PADnn.O pin to the .lca file. The workaround is to manually delete the duplicate record from the .lca file.

Timing Simulation

Input Latch Is Modeled Incorrectly for Timing Simulation

Platform: All
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: 27566

For XC4000E-3, the delay through the IOB input latch is modeled incorrectly in timing simulation. The delay from the pad to the latch output (while the latch is transparent) is modeled as 4.0 ns but should be 3.6 ns (Tp_{li}, the no-delay case).

XNF2WIR May Issue Error on 3-State Output Flip-Flop

Platform: All

Architecture: XC4000E

Design Step: Timing Simulation

Reference Number: 27640

If an XC4000E design contains an output flip-flop that does not have a clock enable, XNF2WIR may fail with the following errors:

```
XNF2WIR: ERROR 214: XNF pin name [T] on [$1I2] does  
not appear on VIEWLOGIC symbol [OFDX]
```

```
XNF2WIR: ERROR 214: XNF pin name [O] on [$1I2] does  
not appear on VIEWLOGIC symbol [OFDX]
```

The following situations cause this error:

- The output flip-flop symbol used in the schematic has a CE pin (for example, OFDTX), but the pin is not connected to any signal.
- The flip-flop symbol used in the schematic is not an output flip-flop, but X-BLOX converts it into one. The flip-flop has a CE pin, but the pin is not connected to any signal.
- An X-BLOX DATA_REG module has an unused CE pin and is implemented in output flip-flops by X-BLOX.

In all these cases, the workaround is to tie the CE pin to VCC.

Although the tied-off CE pin is removed during trimming, it allows XNF2WIR to back-annotate the design properly.

Alternatively, you can replace the flip-flop symbol with a version that does not have a CE pin; for example, OFDTX is replaced by OFDT. The CE pins are tied to VCC within these macros. (This option is not available for the DATA_REG case; the CE pin must be tied to VCC.)

XNF2WIR Fails with Error if -L Option Is Used

Platform: All
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: 27638

If you run XNF2WIR with the -L option on an XC4000E design, it fails with a message similar to the following:

```
XNF2WIR ERROR 210: Could not load symbol XSYM864 of  
type [xc4000:or2].Please ensure the correct libraries  
for the symbols in the XNF file are specified in the  
viewdraw.ini file.
```

This error occurs because XNF2WIR does not properly apply the “xc4000e” alias to each symbol.

The simplest workaround is to run XNF2WIR without the -L option. The -L option is necessary only if you are performing board-level simulation and therefore need to have non-XC4000E libraries in your viewdraw.ini file.

If you are using PROflow, you must modify the batch file that it uses to prepare for timing simulation. The runtsim.bat file is located in the \proser\standard directory. Modify line 4 of this file so that the -L option is not used.

OrCAD: Setup and Hold NOT Checked on RAM Address and WE Pins

Platform: PC
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: Not Available

The OrCAD interface does not support setup/hold violation checks on the address or the write enable pin (WE) pins of synchronous or dual-port RAM models. Violations on the address or WE pins will NOT cause the output to go unknown. In order to ensure proper circuit functionality, the address and the WE pins have to be stable prior to the rising edge of a clock. Please refer to XC4000E Data Sheet for setup time requirements.

The setup time on the data pin "D", however, does get checked. If the setup time is violated, the output correctly goes to an undefined state.

XNF2INF Issues Error for Unconnected WE or ADDR pins on RAMs

Platform: PC
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: 27569

If an XC4000E design contains synchronous or dual port RAMs with unconnected WE or ADDR pins, XNF2INF fails with the following error:

```
*** XNF2INF: ERROR 1303 The following problem(s) were  
found in the xnfba.xnf file; the file could not be  
processed successfully.
```

```
The SETUP record refers to an undefined pin 'WE' near  
line 224 of file xnfba.xnf.
```

```
XNF2INF: Fatal Error:
```

The workaround is to tie the unused RAM address pins to Ground and unused WE pin to VCC. Then recompile the design and create new files with the updated schematics.

Downloading and Configuration

MakeBits Should Configure TTL Inputs by Default

Platform: All
Architecture: XC4000E
Design Step: Downloading and Configuration
Reference Number: 27400

When invoked from the command line, MakeBits configures CMOS input levels for all IOBs by default. This problem does not occur if MakeBits is invoked from XDE. The workaround is to invoke MakeBits with the following -f configuration option:

```
makebits -f input:TTL
```

Chapter 6

Xilinx Customer Support Information

For registration, authorization codes, update information, warranty status, shipping, product issues, and technical support, call Monday through Friday, 8 a.m. to 5 p.m. Pacific time.

Registration, Authorization, and Customer Service

- United States and Canada.....1-800-624-4782
- Europe.....44-1-932-349401
- Japan.....81-33-297-9164
- Southeast Asia/All Other Countries.....852-2410-2739
- Facsimile Transmission.....1-408-559-0115
- International customers may also contact their local sales representative or distributor.

Technical Support

- Technical Support Hotline
 - United States and Canada.....1-800-255-7778
 - International.....1-408-879-5199
- Technical Support FAX (24 hours/7 days)1-408-879-4442
- Technical Support BBS (24 hours/7 days)1-408-559-9327
- Internet E-mail Address (24 hours/7 days).....hotline@xilinx.com

Training

- Xilinx Training Administrator1-408-879-5090
- International customers, contact your local sales representative or distributor.



The Programmable Logic CompanySM



0401450

Printed in U.S.A.



Release Document

XC4000E V1.0.0 Pre-Release

December 1995

Attention: This software is a *pre-release* version.

Some core functionality, product features, third party interfaces, and platform support may not be available or is limited (see the enclosed release note for supported attributes). This software is not fully integrated with the current XACT tools and it is not fully tested to the Xilinx production level quality assurance tests.

Read This Before Installation

Versions and Compatibility

The following master table lists the Xilinx XC4000E Pre-Release software programs with their current version numbers.

Software Versions

Program	Version	Program	Version
LCA2XNF	XC4000E-PRE-1.0.0	XDelay	5.2.0
MakeBits	5.2.0	XDM	5.2.0
MakePROM	5.2.0	XMake	5.2.0
PPR	XC4000E-PRE-1.0.0	XNFBA	5.2.0
XABEL	5.2.0	XNFMerge	5.2.0
XBLOX	XC4000E-PRE-1.0.0	XNFPrep	XC4000E-PRE-1.0.0
XChecker	5.2.0	XSimMake	XC4000E-PRE-1.0.0

Contents

Chapter 1 Introduction

Contents.....	1-1
Software.....	1-1
Documentation.....	1-1
Xilinx PC Protection Key	1-2
Maintenance and Support	1-2

Chapter 2 Installation

Disk Space	2-1
Contents.....	2-2
Installing on a Personal Computer	2-2
License.....	2-3
Installing Viewlogic.....	2-3
Sun Workstation.....	2-3
HP Workstation	2-4
IBM RS6000 Workstation.....	2-4
Installing XSI	2-5
Sun Workstation.....	2-5
HP Workstation	2-6
IBM RS6000 Workstation.....	2-6
Installing Mentor	2-7
Sun Workstation.....	2-7
HP Workstation	2-8
Installing Core Tools	2-8
Sun Workstation.....	2-8
HP Workstation	2-9
IBM RS6000 Workstation.....	2-9
Installing for Use with XACTstep 6.0.....	2-10
Files Required.....	2-10
Modifying Your Setup.....	2-11

XACT Environment Variable	2-11
Path	2-11
Using the XC4000E Software Within XACTstep 6.0	2-12

Chapter 3 Features in This Release

XC4000E Viewlogic.....	3-1
XC4000E Library.....	3-1
PC	3-1
Sun4, HP, and IBM RS6000	3-2
FXC4000E Library	3-2
PC	3-3
Sun4, HP, and IBM RS6000	3-3
Switching to XC4000E from XC4000	3-3
New Symbols	3-4
Location Constraints	3-4
Simulating XC4000E RAM	3-5
XC4000E OrCAD	3-5
XC4000E Library.....	3-5
New Symbols	3-6
XC4000E MemGen	3-6
SYNC_RAM and DP_RAM	3-6
Default and Data Commands.....	3-7
Symbols	3-8
Pin Names of RAM Components	3-8
XC4000E X-BLOX.....	3-9
SYNC_RAM	3-9
DP_RAM	3-9
Conditions for Implementation in an IOB	3-10
XC4000E XDE	3-10
Synchronous and Dual-Port RAMs	3-11
SetMemory Command Initializes All XC4000E RAMs	3-11
H Multiplexers	3-12
IOB Clock Enable.....	3-13
IOB O/EC Routing Model.....	3-13
XC4000 to XC4000E Conversion	3-13
CMOS Thresholds	3-14
Start-Up Sequence Options	3-14
XC4000E PPR	3-14
Guided Design	3-15
XC4000E XACT-Performance Support.....	3-15
Asynchronous RAMs	3-15
Single-Port Synchronous RAMs	3-16

Dual-Port Synchronous RAMs	3-16
XC4000E XSI	3-16
Configuring Synopsys to Synthesize and Simulate	
XC4000E Designs	3-17
New Components in XC4000E Synthesis and	
Simulation Libraries	3-18
Current Restrictions Imposed By Synopsys 3.3	3-20
Using Synchronous and/or Level-Sensitive RAM	3-21
XC4000E MN8 Mentor Interface	3-21
Script Changes	3-26
MEN2XNF8, FNCSIM8, TIMSIM8	3-26
PLD_DVE	3-26
GEN_SCH8	3-27
GEN_SYM8	3-27
Unsupported Features in This Release	3-27
XACT-Floorplanner™	3-27
OrCAD	3-27
XDE	3-27

Chapter 4 Device and Package Support

Device and Package Support	4-1
----------------------------------	-----

Chapter 5 Known Issues

Software	5-1
Design Entry	5-1
Inversion Bubble Is Missing for OrCAD I/O Primitives	5-1
Illegally Instantiating I/Os with XSI May Cause	
FPGA Compiler or Design Compiler to Crash	5-2
Implementation	5-2
PPR: Setting Ignore_rlocs to True May Cause	
Errors in LCA2XNF, MakeBits, XDelay	5-2
PPR Issues ERROR 1582	5-3
Timing Simulation	5-3
Input Latch Is Modeled Incorrectly for Timing Simulation ...	5-3
XNF2WIR May Issue Error on 3-State Output Flip-Flop	5-4
XNF2WIR Fails with Error if -L Option Is Used	5-5
OrCAD: Setup and Hold NOT Checked on RAM	
Address and WE Pins	5-5
XNF2INF Issues Error for Unconnected WE or	
ADDR pins on RAMs	5-6

Downloading and Configuration.....	5-6
MakeBits Should Configure TTL Inputs by Default.....	5-6

Chapter 6 Xilinx Customer Support Information

Registration, Authorization, and Customer Service	6-1
Technical Support	6-1
Training	6-2

Chapter 1

Introduction

Welcome to the XC4000E V1.0.0 Pre-Release from Xilinx!

Contents

The Development System (DS) product that you received contains software, documentation, and/or hardware. New DS Base, Standard, and Extended packages contain hardware, software, and documentation. Interface and Update products contain software and documentation only.

Software

Xilinx software for the PC and for Sun4, HP, and IBM RS6000 workstations is provided on CD-ROM. It consists of the following:

- Subset of FPGA Core Implementation Tools for XC4000E only (DS-502)
- Viewlogic Interface and Libraries for XC4000E only (DS-391)
- OrCAD Interface and Libraries for XC4000E only (DS-35)
- Synopsys Interface and Libraries for XC4000E only (DS-401)
- Mentor8 Interface and Libraries for XC4000E only (DS-344)

Documentation

The following documentation is available for the XC4000E V1.0.0 Pre-Release:

- Online *XC4000E Libraries Guide Supplement* with the XACTstep 5.2/6.0 release, which you can access by clicking on the **Xilinx Online Docs** → **LIBRARIES SUPPLEMENT GUIDE** command.

- Online XC4000E data sheet with the XACTstep 5.2/6.0 release, which you can access by clicking on the **Xilinx Apps Info** → **XC4000E DATA SHEET** command.
- The latest revision of the XC4000E specifications on Xilinx WEBLINX, which you can access at the following address:
<http://www.xilinx.com>

Xilinx PC Protection Key

Xilinx and Viewlogic software are protected by a hardware key for the parallel port of your PC. This key is required to operate the software properly.

Maintenance and Support

This product comes with free technical and product information telephone support (toll-free in the U.S. and Canada). You can also fax and e-mail your questions. See the “Xilinx Customer Support Information” section of this release note for offices and phone numbers.

This product comes with one year of maintenance; you will receive all software and documentation updates automatically during that time. You will receive a notice at the end of the year giving instructions on how to renew your maintenance contract.

Chapter 2

Installation

This section explains how to install the XACTstep XC4000E Pre-Release software on CD-ROM. This software package is intended to support only XC4000E designs.

Note: Keep this software in a separate area. Do not install it over your existing VL-STD, OR-STD, MN8-STD, or SY-STD software from the XACTstep 5.2 release.

Disk Space

To install the XC4000E Pre-Release, you need the amount of disk space shown in the following table. All workstation installations require that you extract the tar files. If tar files are extracted directly from the CD drive on workstations, disk space (direct) applies. If tar files must be first copied from the CD drive, disk space (indirect) applies.

Interface	Platform	Disk Space (Direct)	Disk Space (Indirect)
OrCAD	PC	14 MB	N/A
Viewlogic	PC	13 MB	N/A
Core Tools	PC	57 MB	N/A
Viewlogic	Sun	16 MB	31 MB
Viewlogic	HP	16 MB	31 MB
Viewlogic	IBM RS6000	16 MB	31 MB
Synopsys	Sun	19 MB	38 MB
Synopsys	HP	20 MB	39 MB
Synopsys	RS6000	20 MB	40 MB

Interface	Platform	Disk Space (Direct)	Disk Space (Indirect)
Mentor	Sun	32 MB	63 MB
Mentor	HP	35 MB	69 MB
Core Tools	Sun	68 MB	135 MB
Core Tools	HP	71 MB	142 MB
Core Tools	RS6000	67 MB	134 MB

Contents

The CD-ROM included with this product contains the directories shown in the following table.

Directory	Contents
or_pc	Core tools, data files, and OrCAD interface on PC
vl_pc	Core tools, data files, and Viewlogic interface on PC
4ke_pc	Core tools and data files on PC
vl_ws	Tar files of Viewlogic interface and libraries on Sun, HP, and IBM RS6000
xsi_4ke	Tar files of XSI interface and libraries on Sun, HP, and RS6000
mn8_hppa	Tar file for Mentor8 interface and libraries on HP
mn8_sun4	Tar file for Mentor8 interface and libraries on Sun
xact_4ke	Tar files of core tools and data files on Sun, HP, and RS6000

Installing on a Personal Computer

This section explains how to load your Xilinx software on a personal computer. In these instructions, drive c: represents the PC hard disk, and drive e: represents the CD-ROM drive.

1. Create a new directory, such as "xpre1" on the c: drive.

```
cd c:\xpre1
```

2. Execute one of the following sets of commands.

- For Viewlogic, type the following:

```
xcopy e:\vl_pc\*. * /s
```

```
xcopy e:\4ke_pc\*. * /s
```

- For OrCAD, type the following:

```
xcopy e:\or_pc\*. * /s
```

```
xcopy e:\4ke_pc\*. * /s
```

- For core tools only on the PC, type the following:

```
xcopy e:\4ke_pc\*. * /s
```

3. Modify the XACT environment variable:

```
set XACT=c:\xpre1;existing_xact_path
```

Make sure that c:\xpre1 in the path statement is placed before the directory where the other Xilinx software resides.

License

For workstation installations, you must append the path of the version of the XACT software that you have to the XACT variable that is defined for the XC4000E software in order to access the license. The XACTstep XC4000E Pre-Release does not include any license files.

Installing Viewlogic

This section describes how to install Viewlogic software on Sun, HP, and IBM RS6000 workstations.

Sun Workstation

This section describes how to install Viewlogic software on a Sun workstation. You must extract two tar files from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the two tar files from the CD as follows.

1. Use the cd command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as */correct_path/xpre1*.
2. Execute the following commands.

```
tar -xf cd_rom_path/vl_ws/vl_sn.tar
tar -xf cd_rom_path/xact_4ke/xact_sn.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=(/correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure that you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

HP Workstation

This section describes how to install Viewlogic software on a Hewlett-Packard workstation. You must extract two tar files from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the two tar files from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.
2. Execute the following commands:

```
tar -xf cd_rom_path/VL_WS/VL_HP.TAR\;1
tar -xf cd_rom_path/XACT_4KE/XACT_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`;

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=(/correct_path/xpre1/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

IBM RS6000 Workstation

This section describes how to install Viewlogic software on an IBM RS6000 workstation. You must extract two tar files from the CD and set the XACT variable and path. If your CD player is directly

mounted on your workstation, you can directly extract the two tar files from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/v1_ws/v1_rs.tar
tar -xf cd_rom_path/xact_4ke/xact_rs.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=( /correct_path/xpre1/bin/rs6000 $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing XSI

This section describes how to install XSI software on Sun, HP, and IBM RS6000 workstations.

Sun Workstation

This section describes how to install XSI software on a Sun workstation. You must extract one tar file from the CD and set the `XACT` variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xsi_4ke/xsi_sn.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=( /correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

HP Workstation

This section describes how to install XSI software on a Hewlett-Packard workstation. You need to extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/XSI_4KE/XSI_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path  
set path=(/correct_path/xpre1/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

IBM RS6000 Workstation

This section describes how to install XSI software on an IBM RS6000 workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xsi_4ke/xsi_rs.tar
```


3. Set the path to `/correct_path/xpre1`;

```
setenv XACT /correct_path/xpre1:existing_xact_path
set path=(/correct_path/xpre1/bin/rs6000 $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing Mentor

This section describes how to install Mentor software on Sun and HP workstations.

Sun Workstation

This section describes how to install Mentor8 software on a Sun workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/mn8_sun4/mn8_sn.tar
tar -xf cd_rom_path/xact_4ke/xact_sn.tar
```

3. Set the path to `/correct_path/xpre1`;

```
setenv LCA /correct_path/xpre1
setenv XACT /correct_path/xpre1:existing_xact_path
set path=($LCA/com/sparc $LCA/bin/sparc \
/correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file has occupied.

HP Workstation

This section describes how to install XSI software on an HP workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/MN8_HPPA/MN8_HP.TAR\;1
tar -xf cd_rom_path/XACT_4KE/XACT_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`:

```
setenv LCA /correct_path/xpre1
setenv XACT /correct_path/xpre1:existing_xact_path
set path=($LCA/com/hppa $LCA/bin/hppa \
/correct_path/xpre/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing Core Tools

This section describes how to install the core tools software on Sun, HP, and IBM RS6000 workstations.

Sun Workstation

This section describes how to install the core tools only on a Sun workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xact_4ke/xact_sn.tar
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path  
set path=(/correct_path/xpre1/bin/sparc $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

HP Workstation

This section describes how to install the core tools only on a Hewlett-Packard workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/XACT_4KE/XACT_HP.TAR\;1
```

3. Set the path to `/correct_path/xpre1`:

```
setenv XACT /correct_path/xpre1:existing_xact_path  
set path=(correct_path/xpre1/bin/hppa $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file has occupied.

IBM RS6000 Workstation

This section describes how to install the core tools only on an IBM RS6000 workstation. You must extract one tar file from the CD and set the XACT variable and path. If your CD player is directly mounted on your workstation, you can directly extract the tar file from the CD as follows.

1. Use the `cd` command in UNIX to move to the directory in which you want the XC4000E Pre-Release software to be installed, such as `/correct_path/xpre1`.

2. Execute the following commands:

```
tar -xf cd_rom_path/xact_4ke/xact_rs.tar
```

3. Set the path to `/correct_path/xpre1`;

```
setenv XACT /correct_path/xpre1:existing_xact_path
```

```
set path=(correct_path/xpre1/bin/rs6000 $path)
```

If you need to copy the tar files from the CD first, make sure you have enough disk space for twice the size of the tar file. Also, remove the tar file after extracting it to recover the disk space that the tar file occupied.

Installing for Use with XACTstep 6.0

You can use the XC4000E software within the XACTstep 6.0 Windows environment on the PC.

Files Required

Install (copy) the Viewlogic XC4000E software and the XC4000E core tools and data files from the CD-ROM into a directory called `4kexact`. In the following instructions, drive `e:` represents the CD-ROM drive on your PC, and `c:` represents the PC hard disk. Then, to copy the appropriate files from the CD-ROM into the `\4kexact` directory, execute the following instructions:

```
mkdir c:\4kexact
cd c:\4kexact
xcopy e:\vl_pc\*. * /s
xcopy e:\4ke_pc\*. * /s
```

The XC4000E software is installed in a separate XACT tree so that you can maintain two XACT trees, one containing the XACTstep 6.0 software and data files, and the other the XC4000E software and data files.

In order to use the XC4000E device in conjunction with other Xilinx devices, a new `partlist.xct` file has been created. This `partlist.xct` file is available on the Xilinx Bulletin Board, as `4keptlst.zip`. This file contains the new `partlist.xct` file. Uncompress the zip file and copy

the partlist.xct to your C:\4KEXACT\DATA directory. You will overwrite the existing partlist.xct in C:\4KEXACT\DATA. Do not overwrite the partlist.xct file that exists in C:\XACT\DATA.

Assuming that drive b: represents the floppy drive on your PC, execute the following instructions:

1. Change drives to the b: drive.
2. Uncompress the ZIP file by typing the following:

```
pkunzip b:\4keptlst.zip
```
3. Copy the partlist.xct into your c:\4kexact\data directory. (You will overwrite the existing partlist.xct in c:\4kexact\data). Do not overwrite the partlist.xct file in c:\xact\data.

The partlist.xct file *must* reside in a writable directory; therefore, do not access it from the network.

Modifying Your Setup

You must modify the XACT environment variable and the path in your autoexec.bat file.

XACT Environment Variable

The XACT environment variable should point to at least three locations. If you have installed the XACTstep 6.0 software in c:\xact and the XC4000E software in c:\4kexact and the PRO Series software in c:\proser, verify that you have the following line in your autoexec.bat or setup file:

```
set xact=C:\4KEXACT;C:\XACT;C:\PROSER
```

The \4kexact directory should appear before the \xact directory.

Path

The path in your autoexec.bat (or setup) file should include all the directories containing Xilinx software.

```
PATH C:\4KEXACT;C:\XACT;C:\PROSER;rest_of_your_path
```

The directory containing the XC4000E software should appear before the XC4000 \xact directory.

Using the XC4000E Software Within XACTstep 6.0

This section lists the known issues in using the XC4000E software within the XACTstep 6.0 environment.

- You cannot select the XC4000E family from the Design Entry tools.

Select the XC4000 family instead.

- The Library List Editor always lists the XC4000 library instead of the XC4000E library.

Follow these steps to resolve this problem:

- a) In the Design Entry window, invoke the Library List Editor, select the XC4000 library from Current Libraries, and modify it to XC4000E.
- b) Now select **Add/Change** (the Replace command does not effect the desired action) to include the XC4000E libraries.
- c) Select the XC4000 library and select **Remove**.

This last step is important because the Library List Editor otherwise reverts to using the XC4000 library.

- An error message such as “Could not find xc4000.bos” is issued.

This message is issued if the XACTstep 6.0 software is unable to find all the files required in order to successfully process XC4000E designs. It is not enough to copy only the partlist.xct and device package files. Make sure that all the files have been copied over from the CD-ROM and that the path and the environment variable have been set correctly.

- The Library List Editor reverts to listing the XC4000 libraries instead of the XC4000E library after you exit Windows and re-enter the Design Manager.

Either re-invoke the Library List Editor and modify the library listing, or as a more permanent solution, hard-code the view-draw.ini file in the project directory so that the XC4000E library is always enabled instead of the XC4000 library.

- Simulation may fail because XNF2WIR fails with Error 210.

This error can be resolved by editing a file called runtsim.bat, which is a script file that runs timing simulation. This file is

found in c:\proser\standard\runtsim.bat. Edit line 4 of runtsim.bat to delete the -l option, with which XNF2WIR is invoked. Now XNF2WIR is executed without the -l option.

The -l option of XNF2WIR forces aliases to be attached to the primitives. For example, if XNF2WIR processes an IBUF, the -l option causes it to select the IBUF from the XC4000 library. Since the XC4000 library is not listed in the viewdraw.ini file, an error results. Removing the -l option allows XNF2WIR to select the IBUF from any library. Since only one device library is specified, XNF2WIR correctly chooses the XC4000E IBUF.

- When you invoke PROsim to perform a timing simulation, it should automatically invoke PROwave on completion of the simulation. However, it does not invoke PROwave when you use the XC4000E family.

When PROsim completes, you must manually invoke PROwave to view the waveform. Select the *design.wfm* file and read the waveform display.

- The Floorplanner erroneously allows some XC4000E placements but causes PPR to issue errors.

For example, the Floorplanner allows you to place a configuration composed of a RAM requiring a write clock and an additional clocked flip-flop in the same CLB. This configuration is allowed if the register is clocked with the write clock *only*. Two different clocks cannot be placed in the same CLB. The Floorplanner allows this configuration without reporting errors, but PPR crashes with errors.

The Floorplanner allows two dual-port RAMs to be placed in one CLB, but such placement causes an error because only one dual-port RAM can be placed in a CLB.

- The Floorplanner may issue errors on legal placements by PPR.

After PPR places and routes a design, reading back the routed LCA file may result in errors when you execute a Check Floorplan command. The error message indicates that some CLBs have H function generators in bad configurations. This message is not a valid error message; the Floorplanner writes a valid constraints (CST) file that PPR will accept. Make sure that new errors are not introduced when changing the design layout.

Features in This Release

The XC4000E Pre-Release contains all of the necessary features to support XC4000E designs, but it is not intended for use with other architectures. Please see the “Installation” section for instructions on installing this software. A full explanation of the XC4000E architecture is found in the *XC4000E Field Programmable Gate Array Family Data Sheet*.

The XC4000E design flow is the same as the XC4000 design flow.

XC4000E Viewlogic

This section describes the Viewlogic library and simulation support for the XC4000E architecture. See the *XC4000E Libraries Guide Supplement* for details.

XC4000E Library

The XC4000E library is a superset of the XC4000 library, with the addition of XC4000E-specific elements. To access the XC4000E library, follow these platform-specific instructions.

PC

To access the XC4000E library on a personal computer, perform the following steps.

1. Include this line in your viewdraw.ini file:

```
DIR [m] d:\correct_path\unified\xc4000e (xc4000e)
```

where d: is the drive on which the XC4000E Pre-Release software is installed.

For example, if the software is installed in d:\pre\xact, your viewdraw.ini file should contain the following lines:

```
DIR [m] d:\PRE\XACT\UNIFIED\XC4000E (xc4000e)
DIR [m] d:\PRE\XACT\UNIFIED\XBLOX (xblox)
DIR [m] d:\PRE\XACT\UNIFIED\XBUILTIN (xbuiltin)
DIR [m] d:\PRE\XACT\UNIFIED\BUILTIN (builtin)
```

2. Include the path to the \unified directory in your XACT environment variable:

```
set XACT=d:\PRE\XACT
```

Sun4, HP, and IBM RS6000

To access the XC4000E library on a Sun4, HP, or IBM RS6000 workstation, include this line in your viewdraw.ini file:

```
DIR [r] /correct_path/unified/xc4000e(xc4000e)
```

For example, if the software is installed in /customer/pre/xilinx, your viewdraw.ini file should contain the following lines:

```
DIR [r] /customer/pre/xilinx/unified/xc4000e \
(xc4000e)
DIR [r] /customer/pre/xilinx/unified/xblox (xblox)
DIR [r] /customer/pre/xilinx/unified/xbuiltin \
(xbuiltin)
DIR [r] /customer/pre/xilinx/unified/builtin \
(builtin)
```

FXC4000E Library

The FXC4000E library is the same as the XC4000E library with one difference: the simulation models in the FXC4000E library do not perform timing violation checks like setup, hold, and minimum pulse width. The “f” prefix stands for fast simulation because the removal of timing violation checks can speed up the simulation. To access the FXC4000E library, follow these platform-specific instructions.

Refer to the FXC4000E library using the XC4000E alias. You should not have both the XC4000E and the FXC4000E libraries in one viewdraw.ini file.

PC

To access the FXC4000E library on a personal computer, perform the following steps.

1. Include this line in your viewdraw.ini file:

```
DIR [m] d:\correct_path\unified\fxc4000e (xc4000e)
```

where d: is the drive on which the XC4000E Pre-Release software is installed.

For example, if the software is installed in d:\pre\xact, your viewdraw.ini file should contain the following lines:

```
dir [m] d:\pre\xact\unified\fxc4000e (xc4000e)
dir [m] d:\pre\xact\unified\xblox (xblox)
dir [m] d:\pre\xact\unified\xbuiltin (xbuiltin)
dir [m] d:\pre\xact\unified\builtin (builtin)
```

2. Include the path to the \unified directory in your XACT environment variable.

```
set XACT = d:\PRE\XACT
```

Sun4, HP, and IBM RS6000

To access the FXC4000E library on a Sun4, HP, or IBM RS6000 workstation, include this line in your viewdraw.ini file:

```
DIR [r] /correct_path/unified/fxc4000e(xc4000e)
```

For example, if the software is installed in /customer/pre/xilinx, your viewdraw.ini file should contain the following lines:

```
DIR [r] /customer/pre/xilinx/unified/fxc4000e \
(xc4000e)
DIR [r] /customer/pre/xilinx/unified/xblox (xblox)
DIR [r] /customer/pre/xilinx/unified/xbuiltin \
(xbuiltin)
DIR [r] /customer/pre/xilinx/unified/builtin \
(builtin)
```

Switching to XC4000E from XC4000

If you have an existing Viewlogic schematic design implemented with the XC4000 library and you wish to switch to the XC4000E library, you can do so by changing your viewdraw.ini file, as illustrated in the following examples.

Example 1: On a workstation, if the XC4000E software is installed in /customer/pre/xilinx and the XC4000 software is installed in /customer/old/xilinx, you can change the following line:

```
DIR [r] /customer/old/xilinx/unified/xc4000 (xc4000)
```

to:

```
DIR [r] /customer/pre/xilinx/unified/xc4000e \
(xc4000e)
DIR [r] /customer/pre/xilinx/unified/xc4000e (xc4000)
```

Example 2: On the PC, if the XC4000E software is installed in d:\pre\xact and the XC4000 software is installed in d:\old\xact, you can change the line:

```
DIR [m] d:\OLD\XACT\UNIFIED\XC4000 (xc4000)
```

to:

```
DIR [m] d:\PRE\XACT\UNIFIED\XC4000E (xc4000e)
DIR [m] d:\PRE\XACT\UNIFIED\XC4000E (xc4000)
```

New Symbols

Following are the new XC4000E symbols:

IFDX	IFDXI	IFDX4	IFDX8	IFDX16
IFDX_1	IFDXI_1			
ILDX	ILDXI	ILDX4	ILDX8	ILDX16
ILDX_1	ILDXI_1			
OFDX	OFDXI	OFDX4	OFDX8	OFDX16
OFDX_1	OFDXI_1			
OFDEX	OFDEXI	OFDEX4	OFDEX8	OFDEX16
OFDEX_1	OFDEXI_1			
OFDTX	OFDTXI	OFDTX4	OFDTX8	OFDTX16
OFDTX_1	OFDTXI_1			
RAM16X1D	RAM16X2D	RAM16X4D	RAM16X8D	
RAM16X1S	RAM16X2S	RAM16X4S	RAM16X8S	
RAM32X1S	RAM32X2S	RAM32X4S	RAM32X8S	

The X-BLOX XC4000E RAM symbols are DP_RAM and SYNC_RAM.

Location Constraints

In general, any parameter that is allowed on RAM16X1 or RAM32X1 for XC4000 is allowed on RAM16X1S, RAM32X1S, or RAM16X1D for

XC4000E. As an exception, when an RLOC attribute is used on a RAM16X1D symbol, no extension is allowed, because a RAM16X1D occupies an entire CLB.

Simulating XC4000E RAM

An INIT value placed on a RAM16X1D, RAM16X1S, or RAM32X1S primitive must be processed by XSimMake to be shown in simulation. The XSimMake functional flow creates a file named *sdesign.xmm*, which contains the appropriate ViewSim commands to pre-load the RAM elements. This command file should be executed in ViewSim at the start of both functional and timing simulation. XSimMake also adds commands to the XMM file for asynchronous RAMs, ROMs, and memories created by X-BLOX and MemGen.

XC4000E OrCAD

This section describes the OrCAD library and simulation support for the XC4000E architecture. See the *XC4000E Libraries Guide Supplement* for details.

XC4000E Library

The XC4000E library is a superset of the XC4000 library, with the addition of XC4000E-specific elements. To access the XC4000E library for both schematic and simulation, perform the following steps.

1. Set the XACT environment variable to point to the installed directory of the Xilinx DS35 OrCAD interface.
2. Create the design directory in either of the following two ways:
 - Run the Create Design command from the Design Management Tools in the OrCAD ESP.
 - Create a new directory, then copy into it the OrCAD SDT/VST configuration files (*sdt.cfg* and *vst.cfg*, respectively) from the OrCAD template directory.
3. In the newly created directory, run the Xdraft 4e command from the MS-DOS command line. This command configures both *sdt.cfg* and *vst.cfg* to use the XC4000E libraries for schematic and simulation.

New Symbols

Following are the new XC4000E symbols:

IFDX	IFDXI	IFDX4	IFDX8	IFDX16
IFDX_1	IFDXI_1			
ILDX	ILDXI	ILDX4	ILDX8	ILDX16
ILDX_1	ILDXI_1			
OFDX	OFDXI	OFDX4	OFDX8	OFDX16
OFDX_1	OFDXI_1			
OFDEX	OFDEXI	OFDEX4	OFDEX8	OFDEX16
OFDEX_1	OFDEXI_1			
OFDTX	OFDTXI	OFDTX4	OFDTX8	OFDTX16
OFDTX_1	OFDTXI_1			
RAM16X1D	RAM16X2D	RAM16X4D	RAM16X8D	
RAM16X1S	RAM16X2S	RAM16X4S	RAM16X8S	
RAM32X1S	RAM32X2S	RAM32X4S	RAM32X8S	

The X-BLOX XC4000E RAM symbols are DP_RAM and SYNC_RAM.

XC4000E MemGen

This section describes the new types that have been added to the MemGen program to support the synchronous and dual-port RAMs in the XC4000E architecture. It also provides the pin names of RAM components for instantiating cells in the HDL source code.

SYNC_RAM and DP_RAM

Two new TYPE values, SYNC_RAM and DP_RAM, have been added to the MemGen program. You can specify TYPE values interactively, through the command line, or in the MEM file.

- From the command line, enter the following:

```
type=[rom|ram|sync_ram|dp_ram]
```

Use SYNC_RAM for a single-port, edge-triggered RAM and DP_RAM for a dual-port, edge-triggered RAM.

Here is an example:

```
memgen ram64x1s type=sync_ram memory_depth=64  
word_width=1
```

- In interactive mode, you see the following:
What type of memory do you want to build?
Enter one of the following:
1 for RAMs
2 for ROMs
3 for SYNC_RAMs (XC4000E only)
4 for DP_RAM (XC4000E only)
- In the MEM file, type the following syntax:
type [rom|ram|sync_ram|dp_ram]
A sample MEM file illustrates this syntax:

```
TYPE SYNC_RAM ; The memory is a SYNC_RAM
DEPTH 64 ; The memory is 64-word deep
WIDTH 1 ; Each memory word is 1-bit wide
;
PART 4005EPG156
;
SYMBOL VIEWLOGIC PINS; Build a VIEWLOGIC symbol with
pin inputs
;
DEFAULT 0 ; Add a default value for unspecified
locations
DATA 0 ; Add your SYNC_RAM data here
;
```

Default and Data Commands

In the past, the Default and Data commands were only permitted on ROMs. Now they also apply to the XC4000E RAMs, SYNC_RAMs, and DP_RAMs. They are added to support the INIT attribute on all XC4000E RAMs.

Symbols

MemGen generates symbols for the new RAM types, SYNC_RAM and DP_RAM. To generate a symbol file for OrCAD, use the -o option. To generate a symbol file for Viewlogic, use the -v option.

Pin Names of RAM Components

This section describes how MemGen generates the pin names for RAM components. This information explains how pins are instantiated. The pin order is also important; that is, it should match the order that appears in the netlist.

1. MemGen generates pin names for XC4000/XC4000E RAMs from the following user inputs:

Address lines: An

Data lines: Dn

Output lines: On

Write-Enable line: WE

2. MemGen generates pin names for XC4000E SYNC_RAMs from the following user inputs:

Address lines: An

Data lines: Dn

Output lines: On

Write-Enable line: WE

Clock line: WCLK

3. MemGen generates pin names for XC4000E DP_RAMs from the following user inputs:

Address lines: An

Data lines: Dn

DP read address lines: DPRAn

SP Output lines: SPOn

DP output lines: DPOn

Write-Enable line: WE

Clock line: WCLK

XC4000E X-BLOX

This section describes the new TYPE values that have been added to the XBLOX program to support the synchronous and dual-port RAMs in the XC4000E.

Two new modules have been added to X-BLOX, SYNC_RAM and DP_RAM.

SYNC_RAM

The new SYNC_RAM module synthesizes a synchronous read-write static random-access memory.

The inputs, outputs, and attributes are identical to the current SRAM module with the following exceptions.

- A positive edge on WR_CLK outputs data selected by ADDR on D_OUT, and when WR_EN is active, it loads data on D_IN into the RAM. The WR_CLK pin must be connected.
- ADDR_ERROR is not supported.
- The INIT and MEMFILE attributes have been added. See the “Specifying the Initial Contents of an SRAM, SYNC_RAM, or DP_RAM” section later in this document.
- The value of DEPTH must be a multiple of 16.
- CLB utilization is the same as that of the SRAM in XC4000.

DP_RAM

The DP_RAM module synthesizes a synchronous dual-port read-write static random-access memory.

The inputs, outputs, and attributes are identical to the current SRAM module with the following exceptions.

- The following inputs have been added.
 - WR_CLK

A positive edge on WR_CLK outputs data selected by ADDR on SP_OUT, and when WR_EN is active, it loads data on D_IN into the RAM. The WR_CLK pin must be connected.

- **DPRD_ADDR**

The DPRD_ADDR (dual-port read address) port selects the word that appears on DP_OUT. DPRD_ADDR cannot be connected to a bus with ENCODING=ONE-HOT.

- The following outputs have been added or changed.

- **SP_OUT**

The SP_OUT port reflects the addressed word of the RAM specified by ADDR when WR_CLK goes active.

- **DP_OUT**

The DP_OUT port reflects the addressed word of the RAM specified by DPRD_ADDR.

- ADDR_ERROR is not supported.
- The INIT and the MEMFILE attributes have been added. See the “Specifying the Initial Contents of an SRAM, SYNC_RAM, or DP_RAM” section, following.
- The value of DEPTH must be a multiple of 16.
- CLB utilization is twice that of the SRAM in XC4000.

Conditions for Implementation in an IOB

Item 2 on page 4-53 of the *X-BLOX Reference/User Guide* states that the flip-flop does not use its clock-enable pin. This statement is not valid for XC4000E, because clock enable on IOB registers is supported.

XC4000E XDE

This section describes the new tag values that have been added to the XDE program to support the synchronous and dual-port RAMs, the H-MUXes, and the IOB clock enable. It also describes the new SetMemory command, which sets the initial values on RAMs; two options in MakeBits to set TTL/COMS thresholds; and two new options for the start-up sequence.

Synchronous and Dual-Port RAMs

This release offers a complete set of EDITBLK RAM tag values:
RAM:DP:K:NOT:F:G:FG.

- RAM:DP indicates a dual-port RAM.
- RAM:K indicates a synchronous RAM.
- RAM:NOT indicates that the K pin is inverted.

However, the new RAM:DP tag value and the existing RAM:FG tag value are mutually exclusive.

Allowed values for XC4000E RAM tag are shown in the following table.

RAM:F:G	Two 16x1 non-synchronous RAMs
RAM:F:G:K	Two 16x1 synchronous RAMs
RAM:F:G:K:NOT	Two 16x1 synchronous RAMs, inverted clock
RAM:F	One 16x1 non-synchronous RAM
RAM:G	One 16x1 non-synchronous RAM
RAM:FG	One 32x1 non-synchronous RAM
RAM:F:K	One 16x1 synchronous RAM
RAM:G:K	One 16x1 synchronous RAM
RAM:FG:K	One 32x1 synchronous RAM
RAM:F:K:NOT	One 16x1 synchronous RAM, inverted clock
RAM:G:K:NOT	One 16x1 synchronous RAM, inverted clock
RAM:FG:K:NOT	One 32x1 synchronous RAM, inverted clock
RAM:F:G:DP:K	One 16x1 synchronous, dual-port RAM
RAM:F:G:DP:K:NOT	One 16x1 synchronous, dual-port RAM, inverted clock

SetMemory Command Initializes All XC4000E RAMs

RAM initialization is implemented by the SetMemory command from the Config menu. The SetMemory command is allowed only

inside the block editor. The following table describes the values for the SetMemory options.

Setmemory F G FG value	Non-dual-port RAMs or ROMs. For F and G (16x1 case), a 16-bit value is expected. For FG (32x1), a 32-bit value is expected. G is in the left two bytes of the 32-bit number, and F is in the right two bytes.
Setmemory value	Dual-port RAMs. A 16-bit value is expected and loaded into F and G.

The *value* is a hexadecimal number with a leading “x” character.

In both cases, the MSB of the RAM initial value goes into the high RAM address, as in the following examples.

SETMEMORY F x8000 sets bit 15 of F RAM to 1

SETMEMORY G x0001 sets bit 0 of G RAM to 1.

H Multiplexers

This release offers a complete set of EDITBLK H function tag values: H:F:G:H1:DIN:SR.

- H:DIN indicates that the DIN pin sources the H function generator.
- H:SR indicates that the SR pin sources the H function generator.

However, the new H:DIN tag value and the existing H:F tag value are mutually exclusive. The new H:SR tag value and the existing H:G tag value are mutually exclusive.

Allowed values for XC4000E RAM tag are the following:

H:G	H:DIN:G	H:H1:F:G
H:SR	H:DIN:SR	H:H1:F:SR
H:F	H:H1	H:H1:DIN
H:F:G	H:H1:G	H:H1:DIN:G
H:F:SR	H:H1:SR	H:H1:DIN:SR
H:DIN	H:H1:F	

The H equation (Equate H) for XC4000E takes DIN and SR as arguments. Here are three examples:

```
Equate H = SR  
Equate H = DIN  
Equate H = (SR*DIN*H1)
```

IOB Clock Enable

The new EDITBLK IOB clock-enable tag values for XC4000E are INFF:EC and OUT:EC.

- INFF:EC indicates that the input register has a clock-enable pin.
- OUT:EC indicates that the output register has a clock-enable pin.

IOB O/EC Routing Model

The XC4000E parts now support a clock-enable function on each IOB that is common to both IOB flip-flops. This function is represented as the new EC pin on the IOB. The interconnect that connects to EC is shared between the EC pin and the existing O pin; this sharing results in some limitations in the use of the interconnect when both EC and O are used on the same IOB. In the Design Editor (XDE), the interconnect attached to the O pin of an IOB is shown with two forks. One fork wraps around the IOB parallel to the interconnect attached to the EC pin and matches it PIP for PIP. The second fork extends into the XC4000E device.

When the IOB clock enable is not in use, any of the PIPs on the O-pin interconnect can be enabled to drive a signal to the O pin.

When both the O pin and the EC pin are in use, the PIPs on the O interconnect fork that match those on the EC interconnect cannot be used to drive a signal to the O pin. They can only connect a net to the EC pin. The only PIPs available to drive a signal onto the O pin are those on the fork of interconnect that extends into the device. A DRC check ensures the correct use of the interconnect in this condition.

XC4000 to XC4000E Conversion

To convert an XC4000 LCA file into an XC4000E LCA file, select the Convert command from the Programs menu, then follow the prompts.

An existing XC4000A/D/H LCA file cannot be converted into an XC4000E LCA file.

CMOS Thresholds

The XC4000E Pre-Release allows you to specify the input thresholds for all IOBs on the device by using the MakeBits command from the Programs menu in XDE, as follows:

- INPUT CMOS sets all input thresholds to CMOS.
- INPUT TTL sets all input thresholds to TTL. It is the default.

The XC4000E allows you to specify the output thresholds for all IOBs on the device by using the MakeBits command from the Programs menu in XDE, as follows:

- OUTPUT CMOS sets all output thresholds to CMOS.
- OUTPUT TTL sets all output thresholds to TTL. It is the default.

Note: For additional information see the “MakeBits Should Configure TTL Inputs by Default” item in the “Known Issues” chapter of this release note.

Start-Up Sequence Options

These new options are available with the start-up sequences for the XC4000E.

- M0 Pin
Setting: Pullup, Pulldown
Default: Neither
- M2 Pin
Settings: Pullup, Pulldown
Default: Neither

Note: The Pullup and Pulldown options for the M0 and M2 pins act as toggles and are mutually exclusive. The default is inactive. Selecting one option enables it and disables the other. Selecting the same option a second time disables it.

XC4000E PPR

The XC4000E Pre-Release supports both guided design and XACT-Performance in PPR.

Guided Design

PPR supports the guiding feature for XC4000 or XC4000E designs guiding XC4000E designs in this release. See the *XACTstep Development System Reference Guide*, Volume 2, for information about guided design.

XC4000E XACT-Performance Support

XACT-Performance support for the XC4000E family is mostly unchanged from its behavior for the XC4000 family, but there are new varieties of RAM to consider.

All grouping mechanisms that apply to RAMs in the XC4000 family also apply to RAMs in the XC4000E family. The RAMS qualifier matches any type of RAM, and the predefined RAMS group includes RAMs of all types.

See the “XACT-Performance Utility” chapter in the *XACTstep Development Reference Guide*, Volume 1, for more information on grouping mechanisms.

For all RAM modes, path-tracing behavior is best considered according to where the RAM in question appears in the paths associated with a particular TIMESPEC:

- Type 1: RAM at the end of paths, for example, "from:others:to:rams"
- Type 2: RAM at the start of paths; for example, "from:rams:to:others"
- Type 3: RAM in the middle of paths, for example, an enclosing "from:ffs:to:ffs"

Asynchronous RAMs

For RAMs configured in asynchronous mode, path-tracing behavior is unchanged from the XC4000.

- Type 1 paths (to RAM) are always traced. PPR determines the setup time appropriate to the destination pin.
- Type 2 paths (from RAM) are always traced. PPR determines the worst-case time from a change on D or WE to data valid.

- Type 3 paths (through RAM) are always traced if they arrive at address pins, and they are not traced if they arrive at D or WE pins. Changes on address inputs propagate just as they do for ordinary look-up tables. However, propagation of changes on D or WE is assumed to be of interest only when the RAM is being read during a write operation. If you want PPR to control the delay on paths through the D or WE inputs, you must split the delay requirement into two segments: one ending at the RAM input pin, and the other beginning at the RAM output.

Single-Port Synchronous RAMs

Path-tracing for synchronous RAMs is essentially the same as for asynchronous RAMs.

- Type 1 paths (to RAM) are always traced. PPR determines setup times with respect to the WCLK pin, for all other inputs.
- Type 2 paths (from RAM) are always traced. PPR determines worst-case time after WCLK transition to data valid.
- Type 3 paths (through RAM) are traced through address pins only.

Dual-Port Synchronous RAMs

Tracing behavior for dual-port synchronous RAMs is as follows.

- Type 1 paths (to RAM) are traced to every input pin destination except DPRA0 to DPRA3. Read address inputs cannot impact paths that end at a RAM (write function).
- Type 2 paths (from RAM) are always traced.
- Type 3 paths (through RAM) are traced through address inputs only. In particular, there are A?-to-SP0 paths and DPRA?-to-DPO paths.

XC4000E XSI

The XC4000E Pre-Release of XSI is a superset of the existing XSI 5.2 product. The principal changes include the following:

- Addition of several new cells to the FPGA Compiler/Design Compiler synthesis libraries and the VSS FTGS simulation libraries

- Additional timing information to support new XC4000E speed grades
- XSI executables that now recognize new XC4000E part types
- Equal application of all other XC4000-specific optimizations performed by FPGA-Compiler to the XC4000E architecture, including:
 - Direct synthesis to XC4000 IOB and CLB structures
 - Automatic inference of X-BLOX modules for optimized arithmetic
 - Clock-buffer insertion
 - Synthesis to flip-flops with clock enables

Configuring Synopsys to Synthesize and Simulate XC4000E Designs

The .synopsys_dc.setup file for XC4000E synthesis should be identical to that for XC4000 designs with the exception of the target- and link-library settings and the reference to the X-BLOX Design Ware library. An example .synopsys_dc.setup file follows. The target- and link-library settings were created using the Synlibs 4005e-3 command.

```
search_path = { . \
<XC4000E_DS401_install_path>/synopsys/libraries/syn
\ <Synopsys_install_path>/libraries/syn}

define_design_lib xblox_4000e -path \
<XC4000E_DS401_install_path>/synopsys/libraries/dw/
lib/fpga/xc4000e

compile_fix_multiple_port_nets = true
xlnx_hier_blknm = 1
xnfout_library_version = "2.0.0"

bus_naming_style = "%s<%d>"
bus_dimension_separator_style = "><"
bus_inference_style = "%s<%d>"

link_library = {xprim_4005e-3.db xprim_4000e-3.db \
xgen_4000e.db xfpga_4000e-3.db xio_4000e-3.db}
target_library = {xprim_4005e-3.db xprim_4000e-3.db \
xgen_4000e.db xfpga_4000e-3.db xio_4000e-3.db}
```

```
symbol_library = {xc4000e.sdb}
synthetic_library = {xblox_4000e.sldb standard.sldb}
```

Notice that the reference to the XC4000E X-BLOX Design Ware library is differentiated from the XC4000 X-BLOX library with a new name, `xblox_4000e`.

The `.synopsys_vss.setup` file for XC4000E simulation should also be identical to that for XC4000 designs with the exception of the simulation library reference. The following is a sample `.synopsys_vss.setup` file.

```
timebase=ns
time_res_factor=0.1
no_hazard_mesg=true
WORK > DEFAULT
DEFAULT : ./WORK
xc4000e:<XC4000E_DS401_install_path>/synopsys/
libraries/sim/lib/xc4000e
```

New Components in XC4000E Synthesis and Simulation Libraries

Several new components have been added to the XC4000E synthesis and simulation libraries, reflecting the XC4000E architecture's new features. The new components, listed in the following tables, include synchronous single-port RAMs, synchronous dual-port RAMs, and I/O flip-flops/latches with clock/latch-enables. Synopsys does not synthesize cells marked with an asterisk, so these components must be instantiated.

Synchronous RAM			
Name	Outputs	Inputs	Notes
RAM16X1S	O	D, A3, A2, A1, A0, WE, WCLK	*
RAM32X1S	O	D, A4, A3, A2, A1, A0, WE, WCLK	*
RAM16X1D	SPO,DPO	D, A3, A2, A1, A0, DPRA3, DPRA2, DPRA1, DPRA0, WE, WCLK	*

IOB Input Flip-Flops with Clock Enable			
Name	Output	Inputs	Notes
IFDX	Q	D, C, CE	
IFDX_F	Q	D, C, CE	NODELAY*
IFDX_U	Q	D, C, CE	UNBONDED*
IFDXI	Q	D, C, CE	INIT=S
IFDXI_F	Q	D, C, CE	INIT=S, NODELAY*
IFDXI_U	Q	D, C, CE	INIT=S, NODELAY, UNBONDED*

IOB Output Flip-Flops with Clock Enable			
Name	Outputs	Inputs	Notes
OFDX	Q	D, C, CE	
OFDX_F	Q	D, C, CE	Fast slew rate
OFDX_FU	Q	D, C, CE	Fast slew rate, unbonded*
OFDX_S	Q	D, C, CE	Slow slew rate
OFDX_U	Q	D, C, CE	Unbonded*
OFDXI	Q	D, C, CE	INIT=S
OFDXI_F	Q	D, C, CE	INIT=S, fast slew rate
OFDXI_S	Q	D, C, CE	INIT=S, slow slew rate
OFDXI_U	Q	D, C, CE	INIT=S, unbonded*

IOB Tristatable Output Flip-Flops with Clock Enable			
Name	Outputs	Inputs	Notes
OFDTX	O	D, C, CE, T	
OFDTX_F	O	D, C, CE, T	Fast slew rate
OFDTX_S	O	D, C, CE, T	Slow slew rate
OFDTX_U	O	D, C, CE, T	Unbonded*

IOB Tristatable Output Flip-Flops with Clock Enable			
Name	Outputs	Inputs	Notes
OFDTXI	O	D, C, CE, T	INIT=S
OFDTXI_F	O	D, C, CE, T	INIT=S, fast slew rate
OFDTXI_S	O	D, C, CE, T	INIT=S, slow slew rate
OFDTXI_U	O	D, C, CE, T	INIT=S, unbonded*

IOB Input Latches with Gate Enable			
Name	Outputs	Inputs	Notes
ILDX_1	Q	D, G, GE	
ILDX_1F	Q	D, G, GE	NODELAY*
ILDX_1U	Q	D, G, GE	Unbonded*
ILDXI_1	Q	D, G, GE	INIT=S
ILDXI_1F	Q	D, G, GE	INIT=S, NODELAY
ILDXI_1U	Q	D, G, GE	INIT=S, unbonded*

Note: All IOB input gate-enabled latches have active-Low gate pins.

Current Restrictions Imposed By Synopsys 3.3

The Synopsys FPGA Compiler and Design Compiler products are currently unable to infer IOB registers with clock enables. The only way to access these features is to instantiate the appropriate cells in the HDL source code. This limitation will be addressed in a subsequent release of Synopsys' products; however, in the meantime, one of the optimizations performed by X-BLOX is to push clock-enabled CLB registers into IOBs where possible. Refer to the *XACTstep X-BLOX Reference/User Guide* for information about the rules governing this process.

Using Synchronous and/or Level-Sensitive RAM

The Synopsys FPGA Compiler and Design Compiler products are currently unable to infer RAM. (Although RAM can be described behaviorally, this methodology currently synthesizes to inefficient latch- or register-based implementations.) The only way to access these features is to instantiate the appropriate cells in the HDL source code. Alternatively, you can instantiate memory modules created by MemGen, which is explained in the “Pin Names of RAM Components” section earlier in this release note.

The XC4000E RAM modules allow you to specify their contents at power-on. When RAM modules are instantiated directly in the HDL source code, you can enter initialization values for the RAM using the following command:

```
set_attribute instance_name xnf_init init_value
-type string
```

For 16-location RAMs, specify a 4-digit hexadecimal value for *init_value*. For 32-location RAMs, specify an 8-digit hexadecimal value.

Although this mechanism permits RAM-initialization information to be carried into the FPGA implementation tools for incorporation into the configuration bitstream, it does not simulate behaviorally. For behavioral simulation, a RAM's contents remain unknown until they are defined by valid write access. However, back-annotated functional or timing simulation reflects this RAM initialization information.

XC4000E MN8 Mentor Interface

The XC4000E Pre-Release of Mentor Interface (MN8) is a superset of the existing MN8 5.2 product. The major changes include the following:

- Addition of several new cells to the Xilinx library for Mentor Interface
- Changes in the scripts to support the new cells and features
- Single-Port Synchronous RAM. The two new RAM symbols (primitives) are the following:
 - RAM16X1S

- RAM32X1S

There is a new WCLK pin, which occupies the same location as WE pin did in XC4000. The WE pin is now placed above the D data pin.

- Six new RAM macros:

- RAM16X2S
- RAM16X4S
- RAM16X8S
- RAM32X2S
- RAM32X4S
- RAM32X8S

The WE pin is now placed above the D pin.

- Dual-port RAM. RAM16X1D is the new primitive.

The WE pin is now placed above the D pin.

- Three new macros:

- RAM16X2D
- RAM16X4D
- RAM16X8D

The WE pin is now placed above the D pin.

- CE on I/O registers. The IOBs of the XC4000E parts, which have a pin, the CE or GE pin, that is not available on other XC4000 parts. The Mentor primitives that are affected are the following:

- INFF
- INLAT
- INREG
- OUTFF
- OUTFFT

- Eight new primitives:

- IFDX

- IFDXI
- ILDX_1
- ILDXI_1
- OFDX
- OFDXI
- OFDTX
- OFDTXI

The difference between these and the existing I/O register primitives is the addition of a CE pin between D and C pins for the flip-flops and the addition of a GE pin between the D and G pins for the latches.

- Twenty-seven new I/O macros:
 - IFDX4, IFDX8, IFDX16, IFDX_1, IFDXI_1
 - ILDX, ILDX4, ILDX8, ILDX16, ILDXI
 - OFDX4, OFDX8, OFDX16, OFDX_1, OFDXI_1
 - OFDEX, OFDEX4, OFDEX8, OFDEX16, OFDEXI, OFDEX_1, OFDEXI_1
 - OFDTX4, OFDTX8, OFDTX16, OFDTX_1, OFDTXI_1
- A new non-invertible pin required on the I/O register primitives. The pin name is CE for flip-flops and GE for latches. The XNF primitive names remain the same as they are currently.
- The INIT parameter on all RAM, RAMS, and RAMD primitives.
- Back-annotation. The -m option of the XNFBA program generates an MBA file for post-route timing back-annotation into Mentor. For RAM16X1S, RAM32X1S, and RAM16X1D, delays on all pins but the WCLK pin are routing delays and are put “on-parent.” The WCLK pin has double delay: routing and block. The delay on the WCLK pin is the block delay. The delay on the BUF driving the WCLK pin is the routing delay. The “on-parent” delay is written with the “RISE_{pin_name}” format. The double delay is written with the “RISE_{pin_name}” format for the routing delay and the “RISE_{pin_name}” format for the block delay.

The following example shows how a RAM16X1S symbol is back-annotated.

Following is an example of a RAM16X1S before routing:

```
SYM, FIFO/RAMS/RAMSL/$1I92, RAMS, SCHNM=RAM16X1S,
INIT=0, LIBVER=2.0.0

PIN, O, O, DO7

PIN, WE, I, FIFO/CE_WR

PIN, D, I, FIFO/DIQ7

PIN, WCLK, I, CLK

PIN, A0, I, FIFO/A0

PIN, A1, I, FIFO/A1

PIN, A2, I, FIFO/A2

PIN, A3, I, FIFO/A3

END
```

The next example shows a RAM16X1S after routing (PPR, XDelay, LCA2XNF):

```
SYM, DO7, RAMS, LIBVER=2.0.0, SCHNM=RAM16X1S,
INIT=0000

PIN, O, O, DO7, 2.0

PIN, WE, I, FIFO/CE_WR, 5.0

PIN, D, I, FIFO/DIQ7, 1.1

PIN, WCLK, I, YSIG121, 6.0

PULSE, WCLK, +, 8.4

PIN, A3, I, FIFO/A3, 3.1

PIN, A2, I, FIFO/A0, 5.3

PIN, A1, I, FIFO/A2, 2.7

PIN, A0, I, FIFO/A1, 5.8

SETUP, D, WCLK, +, 3.3, 1.1

SETUP, WE, WCLK, +, 1.3, 0.4

SETUP, A3, WCLK, +, 2.4, 0.7

SETUP, A2, WCLK, +, 2.4, 0.7

SETUP, A1, WCLK, +, 2.4, 0.7
```



```
        SETUP, A0, WCLK, +, 2.4, 0.7
END
SYM, XSYM220, BUF, LIBVER=2.0.0
        PIN, O, O, YSIG121
        PIN, I, I, CLK, 1.4
END
```

Here is an example of a RAM16X1S after back-annotation with XNFBA, which produces an MBA file:

```
/FIFO/RAMS/RAMSL/$1I92 N RISEO 2.0
/FIFO/RAMS/RAMSL/$1I92 N FALLO 2.0
/FIFO/RAMS/RAMSL/$1I92 N RISEA0 5.3
/FIFO/RAMS/RAMSL/$1I92 N FALLA0 5.3
/FIFO/RAMS/RAMSL/$1I92 N RISEA1 5.8
/FIFO/RAMS/RAMSL/$1I92 N FALLA1 5.8
/FIFO/RAMS/RAMSL/$1I92 N RISEA2 2.7
/FIFO/RAMS/RAMSL/$1I92 N FALLA2 2.7
/FIFO/RAMS/RAMSL/$1I92 N RISEA3 3.1
/FIFO/RAMS/RAMSL/$1I92 N FALLA3 3.1
/FIFO/RAMS/RAMSL/$1I92 N RISED 1.1
/FIFO/RAMS/RAMSL/$1I92 N FALLD 1.1
/FIFO/RAMS/RAMSL/$1I92 N RISEWE 5.0
/FIFO/RAMS/RAMSL/$1I92 N FALLWE 5.0
/FIFO/RAMS/RAMSL/$1I92 N RISEWCLK 1.4
/FIFO/RAMS/RAMSL/$1I92 N FALLWCLK 1.4
/FIFO/RAMS/RAMSL/$1I92 N RISE_WCLK 6.0
/FIFO/RAMS/RAMSL/$1I92 N FALL_WCLK 6.0
/FIFO/RAMS/RAMSL/$1I92 N SETUPA1 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA1 0.7
/FIFO/RAMS/RAMSL/$1I92 N SETUPA2 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA2 0.7
/FIFO/RAMS/RAMSL/$1I92 N SETUPA0 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA0 0.7
```

```
/FIFO/RAMS/RAMSL/$1I92 N SETUPA3 2.4
/FIFO/RAMS/RAMSL/$1I92 N HOLDA3 0.7
/FIFO/RAMS/RAMSL/$1I92 N SETUPWE 1.3
/FIFO/RAMS/RAMSL/$1I92 N HOLDWE 0.4
/FIFO/RAMS/RAMSL/$1I92 N SETUPD 3.3
/FIFO/RAMS/RAMSL/$1I92 N HOLDD 1.1
/FIFO/RAMS/RAMSL/$1I92 N PULSEWCLK 8.4
/FIFO/RAMS/RAMSL/$1I92 S PULSE_POLARITY_WCLK +
/FIFO/RAMS/RAMSL/$1I92 S INIT 0
```

Script Changes

The Mentor scripts have been changed to support the new cells in the XC4000E Pre-Release. In addition to these changes, XBLXGS and EDIF2XNF have been modified to support the new XC4000E library.

MEN2XNF8, FNCSIM8, TIMSIM8

From the Design Manager (pld_dmgr), you can invoke commands to execute specific scripts, as follows.

- PLD_Men2XNF8 executes the script to translate your design to an XNF file. You must run this command before you can use either PLD_FNCSIM8 or PLD_TIMSIM8.
- PLD_FNCSIM8 executes the script to prepare your design for functional simulation.
- PLD_TIMSIM8 executes the script to prepare your design for timing simulation.

Refer to the *Mentor Graphics Interface/Tutorial Guide* for more details these commands.

PLD_DVE

This command executes the script that invokes the Mentor Graphics Design Viewpoint Editor (DVE) configured for Xilinx designs. The following example shows the Pld_dve command as you would use it in Men2XNF8 to check for the valid technology:

```
pld_dve t2 xc4000e
```

GEN_SCH8

This program creates a new schematic composed of only schematic elements that can be used for functional simulation. An example follows:

```
gen_sch8 xnf_file
```

Xnf_file is any XC4000E file; that is, its part starts with “xc4000e.”

The output should be a valid design. To verify its validity, bring the design into the Design Architect and run the Check Sheet command.

GEN_SYM8

This program automatically creates a symbol based on information in the XSF file. An example follows:

```
gen_sym8 xnf_file
```

Xnf_file is any XC 4000E file. The output should be a valid symbol. To verify its validity, bring the design into the Design Architect and examine the symbol.

Unsupported Features in This Release

The XC4000E Pre-Release does not support the following features.

XACT-Floorplanner™

The XACT-Floorplanner is not available in this release.

OrCAD

The OrCAD library does not support the RAM, RAMS, or RAMD initialization simulation in this release.

XDE

The Convert command does not support the conversion from a smaller XC4000E part to a larger part.

Chapter 4

Device and Package Support

The following is a master table of Xilinx devices for this release. For more information on architectural families and specific device parameters, see *The Programmable Logic Data Book*.

Device	Packages	Speed Grades
XC4005E	PC84, PG156, PQ100, PQ160, PQ208, TQ144	-3
XC4006E	PC84, PG156, PQ160, PQ208, TQ144	-3
XC4008E	PC84, PG191, PQ160, PQ208	-3
XC4010E	BG225, PC84, PG191, PQ160, PQ208	-3
XC4013E	BG225, HQ208, HQ240, PG223, PQ160, PQ208, PQ240	-3
XC4020E ^a	HQ208, HQ240, PG223	-3
XC4025E	HQ240, HQ304, PG223, PG299	-3

a. XC4020E speed file is advanced.

The files shown in the table pertain to the XC4000E device family only. The advanced speed files are derived from simulation models and will change as characterized data becomes available.

Known Issues

This chapter describes the known issues in this release.

Software

The issues in this section are listed in the order that they occur in the design process.

Design Entry

Inversion Bubble Is Missing for OrCAD I/O Primitives

Platform: All

Architecture: XC4000E

Design Step: Design Entry

Reference Number: 27552

The OFDTXI_1, OFDXI_1, ILDXI_1, IFDXI_1, OFDTI_1, and OFDI_1 I/O primitives are missing the inversion bubbles on the clock pin.

The OFDEI symbol has a bubble on the clock pin but should not.

In all these cases, the netlist is correct. The problem is only in the drawing.

Illegally Instantiating I/Os with XSI May Cause FPGA Compiler or Design Compiler to Crash

Platform: All
Architecture: XC4000E
Design Step: Synopsys FPGA Synthesis
Reference Number: 27629

Certain illegal actions may cause the FPGA Compiler or the Design Compiler to crash rather than to give an error message when you instantiate I/Os in XC4000E devices. For example, you might instantiate one or more I/O cells, such as an IFD, OFD, or OFDTX, and connect them to a port *and* include that port in the port list of the `set_port_is_pad` command.

This situation is illegal because the `set_port_is_pad` command tells the FPGA Compiler or Design Compiler to synthesize I/O logic for the indicated ports. If you have already instantiated the I/O logic for any of these ports, further I/O logic should not be synthesized.

Ordinarily, this situation would cause an error message to be issued, but in the current Synopsys release, 3.3b, the FPGA Compiler or Design Compiler may crash.

Implementation

PPR: Setting Ignore_rlocs to True May Cause Errors in LCA2XNF, MakeBits, XDelay

Platform: All
Architecture: XC4000E
Design Step: Implementation
Reference Number: 27556

If you set the PPR `Ignore_rlocs` option to True on a design containing dual-port RAM symbols, PPR may implement the RAM incorrectly, causing errors to be reported in LCA2XNF, MakeBits, or XDelay.

To avoid this problem, rerun the XNFPrep program with the `Ignore_rlocs=True` option. Then rerun PPR using `Ignore_rlocs=False`, which is the default.

XNFPrep removes RLOC constraints from the design, avoiding the problem in PPR.

PPR Issues ERROR 1582

Platform: All
Architecture: XC4000E
Design Step: Implementation
Reference Number: 27387

PPR may issue the following Internal Error when it partially routes a net that includes a clock enable pin on an IO block:

```
*** PPR: ERROR 1582:
      Error in writing LCA data to memory:
      Pin already exists in memory
      Data: "PADnn.O"
This is an internal error; please contact Xilinx
support personnel.
```

PPR writes a duplicate Addnet or Addpin record for the PADnn.O pin to the .lca file. The workaround is to manually delete the duplicate record from the .lca file.

Timing Simulation

Input Latch Is Modeled Incorrectly for Timing Simulation

Platform: All
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: 27566

For XC4000E-3, the delay through the IOB input latch is modeled incorrectly in timing simulation. The delay from the pad to the latch output (while the latch is transparent) is modeled as 4.0 ns but should be 3.6 ns (Tp_{li}, the no-delay case).

XNF2WIR May Issue Error on 3-State Output Flip-Flop

Platform: All

Architecture: XC4000E

Design Step: Timing Simulation

Reference Number: 27640

If an XC4000E design contains an output flip-flop that does not have a clock enable, XNF2WIR may fail with the following errors:

```
XNF2WIR: ERROR 214: XNF pin name [T] on [$1I2] does  
not appear on VIEWLOGIC symbol [OFDX]
```

```
XNF2WIR: ERROR 214: XNF pin name [O] on [$1I2] does  
not appear on VIEWLOGIC symbol [OFDX]
```

The following situations cause this error:

- The output flip-flop symbol used in the schematic has a CE pin (for example, OFDTX), but the pin is not connected to any signal.
- The flip-flop symbol used in the schematic is not an output flip-flop, but X-BLOX converts it into one. The flip-flop has a CE pin, but the pin is not connected to any signal.
- An X-BLOX DATA_REG module has an unused CE pin and is implemented in output flip-flops by X-BLOX.

In all these cases, the workaround is to tie the CE pin to VCC.

Although the tied-off CE pin is removed during trimming, it allows XNF2WIR to back-annotate the design properly.

Alternatively, you can replace the flip-flop symbol with a version that does not have a CE pin; for example, OFDTX is replaced by OFDT. The CE pins are tied to VCC within these macros. (This option is not available for the DATA_REG case; the CE pin must be tied to VCC.)

XNF2WIR Fails with Error if -L Option Is Used

Platform: All
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: 27638

If you run XNF2WIR with the -L option on an XC4000E design, it fails with a message similar to the following:

```
XNF2WIR ERROR 210: Could not load symbol XSYM864 of
type [xc4000:or2].Please ensure the correct libraries
for the symbols in the XNF file are specified in the
viewdraw.ini file.
```

This error occurs because XNF2WIR does not properly apply the “xc4000e” alias to each symbol.

The simplest workaround is to run XNF2WIR without the -L option. The -L option is necessary only if you are performing board-level simulation and therefore need to have non-XC4000E libraries in your viewdraw.ini file.

If you are using PROflow, you must modify the batch file that it uses to prepare for timing simulation. The runtsim.bat file is located in the \proser\standard directory. Modify line 4 of this file so that the -L option is not used.

OrCAD: Setup and Hold NOT Checked on RAM Address and WE Pins

Platform: PC
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: Not Available

The OrCAD interface does not support setup/hold violation checks on the address or the write enable pin (WE) pins of synchronous or dual-port RAM models. Violations on the address or WE pins will NOT cause the output to go unknown. In order to ensure proper circuit functionality, the address and the WE pins have to be stable prior to the rising edge of a clock. Please refer to XC4000E Data Sheet for setup time requirements.

The setup time on the data pin "D", however, does get checked. If the setup time is violated, the output correctly goes to an undefined state.

XNF2INF Issues Error for Unconnected WE or ADDR pins on RAMs

Platform: PC
Architecture: XC4000E
Design Step: Timing Simulation
Reference Number: 27569

If an XC4000E design contains synchronous or dual port RAMs with unconnected WE or ADDR pins, XNF2INF fails with the following error:

```
*** XNF2INF: ERROR 1303 The following problem(s) were  
found in the xnfba.xnf file; the file could not be  
processed successfully.
```

```
The SETUP record refers to an undefined pin 'WE' near  
line 224 of file xnfba.xnf.
```

```
XNF2INF: Fatal Error:
```

The workaround is to tie the unused RAM address pins to Ground and unused WE pin to VCC. Then recompile the design and create new files with the updated schematics.

Downloading and Configuration

MakeBits Should Configure TTL Inputs by Default

Platform: All
Architecture: XC4000E
Design Step: Downloading and Configuration
Reference Number: 27400

When invoked from the command line, MakeBits configures CMOS input levels for all IOBs by default. This problem does not occur if MakeBits is invoked from XDE. The workaround is to invoke MakeBits with the following -f configuration option:

```
makebits -f input:TTL
```

Chapter 6

Xilinx Customer Support Information

For registration, authorization codes, update information, warranty status, shipping, product issues, and technical support, call Monday through Friday, 8 a.m. to 5 p.m. Pacific time.

Registration, Authorization, and Customer Service

- United States and Canada.....1-800-624-4782
- Europe.....44-1-932-349401
- Japan.....81-33-297-9164
- Southeast Asia/All Other Countries.....852-2410-2739
- Facsimile Transmission.....1-408-559-0115
- International customers may also contact their local sales representative or distributor.

Technical Support

- Technical Support Hotline
 - United States and Canada.....1-800-255-7778
 - International.....1-408-879-5199
- Technical Support FAX (24 hours/7 days)1-408-879-4442
- Technical Support BBS (24 hours/7 days)1-408-559-9327
- Internet E-mail Address (24 hours/7 days).....hotline@xilinx.com

Training

- Xilinx Training Administrator1-408-879-5090
- International customers, contact your local sales representative or distributor.



The Programmable Logic CompanySM



0401450

Printed in U.S.A.