



Release Document

XACT^{step} Version 5.2/6.0
Synopsys

October 1995

Read This Before Installation

Installing Online Documentation

Starting with the 5.2/6.0 release, online documentation is now available on the Sun and HP workstations.

Installing Online Documentation on a Sun Workstation

To use online documentation, you must install the Acrobat reader and the online documents on your workstation.

Installing the Reader and Documents

Version 1.0 of the Acrobat reader is included on the XACTstep Sun 5.2 CD-ROM disk. To install the Acrobat reader, follow the instructions on page 4-4 of the *Getting Started & Installation Guide*.

Because the online documents are in tar format, you must use the XACTstep 5.2 installation program to install the online documents on your workstation. Refer to the Sun4 instructions on page 3-2 of the *Getting Started & Installation Guide*.

Opening Documents with Acrobat Reader — Sun Workstation Installations

To access the AcroRead program from the command line, follow these instructions:

1. Include the path to the /AcroRead_1.0/bin directory in the \$path variable of your configuration file,
2. At the command line, type the following to invoke the Acrobat reader:

```
acroread
```

The Open file dialog box of the Acrobat reader is displayed.

3. Specify the following path in the Filter box of the Open file dialog box to view Xilinx Online Documents:

```
/xact_dir/online/online/*.pdf
```

To view Xilinx Application Information, specify the following path:

```
/xact_dir/online/onlinedb/*.pdf
```

4. Select the document you want from the displayed list of .pdf files.

Installing Online Documentation on a HP Workstation

To use online documentation, you must install the Acrobat reader and, optionally, the online documents on your workstation.

Installing the Reader and Documents

Version 2.1 of the Acrobat reader is available on HP workstations on a separate enclosed Acrobat CD-ROM disk also provided by Xilinx. Use the instructions outlined in this section to install the Acrobat software on an HP workstation.

Installation of the Acrobat reader requires the HP-UX 9.05 operating system, the HP-VUE window manager, and 12 MB of disk space. You do not have to install the online documents to your hard disk, but if you choose to do so, you will need 52 MB of disk space.

1. Insert the CD-ROM disk into the CD-ROM drive.
2. Mount the CD-ROM drive. You need system administrator privileges to complete this step.
3. Invoke the Acrobat Installation program as follows:

```
/cdrom_dir/acrobat/unix/install
```

By default, after you have installed the desired products to your HP workstation, the installation program copies the Acrobat reader to the /usr/AcroRead directory. Xilinx recommends that you install the reader to /xact_dir/doc/AcroRead. You must include the AcroRead/bin directory in your path.

For more information, print the “Introducing Adobe Acrobat Reader 2.1” file located in `/cdrom_dir/acrobat/unix/instguid.txt`.

Note: If you want to install the document files on your workstation, copy the `/cdrom_dir/onlindb` and `/cdrom_dir/online` directory trees from the XACTstep Version 5.2 CD-ROM to your disk. For example:

```
cp -Rp /cdrom_dir/onlindb /xact_dir/doc/onlindb ↵
cp -Rp /cdrom_dir/online /xact_dir/doc/online ↵
chmod -R u+w xact_dir/doc↵
```

Opening Documents with Acrobat Reader — HP Workstation Installations

To view documents on an HP workstation, follow the instructions outlined in this section. For additional information refer to the “Viewing Documents with Acrobat Reader” section on page 4-7 of the *Getting Started & Installation Guide*.

You can either start the reader first and then decide what type of documents you want to view, or you can open the type of documents you want to view at the same time you load the reader.

To start the reader without specifying any documents, follow these instructions:

1. Ensure that the Acrobat Reader `AcroRead/bin` directory is in your path.
2. To start the reader, type the following:
acroread
3. Specify one of the following paths corresponding to the type of documents you wish to view:

To view Xilinx Online Documents, open the file:

`/cdrom_dir/online/linkpage.pdf`

or

`/xact_dir/online/linkpage.pdf`

To view Xilinx Application Information, open the file:

`/cdrom_dir/onlindb/dblink.pdf`

or

```
/xact_dir/onlindb/dblink.pdf
```

To specify the type of documents you wish to view at the time you invoke the reader, include the path you want after the `acroread` command as follows:

To view Xilinx Online Documents, use the command:

```
acroread /cdrom_dir/online/linkpage.pdf &
```

or

```
acroread /xact_dir/online/linkpage.pdf &
```

To view Xilinx Application Information, use the command:

```
acroread /cdrom_dir/onlindb/dblink.pdf &
```

or

```
acroread /xact_dir/onlindb/dblink.pdf &
```

Versions and Compatibility

The following master table indicates Xilinx core software with the current version numbers.

Software Versions

Program	Windows Version	DOS Version	Workstation Version
APR	5.2	5.2	5.2
APRLOOP	5.2	5.2	5.2
CstCvt	5.2	5.2	5.2
Design Manager	6.0	N/A	N/A
Floorplanner	6.0	N/A	5.2
Flow Engine	6.0	N/A	N/A
Hardware Debugger	6.0	N/A	N/A
HM2RPM	5.2	5.2	5.2
LCA2XNF	5.2	5.2	5.2
MakeBits	5.2	5.2	5.2
MakePROM	5.2	5.2	5.2
MAP2LCA	5.2	5.2	5.2
MemGen	5.2	5.2	5.2
PPR	5.2	5.2	5.2
PROM File Formatter	6.0	N/A	N/A
Report Browser	6.0	N/A	N/A
SymGen	5.2	5.2	5.2
Timing Analyzer	6.0	N/A	N/A

Program	Windows Version	DOS Version	Workstation Version
XACT	5.2	5.2	5.2
XBLOX	5.2	5.2	5.2
XChecker	5.2	5.2	5.2
XCK88	N/A	5.2	N/A
XDE	5.2	5.2	5.2
XDelay	5.2	5.2	5.2
XDM	N/A	N/A	5.2
xdm	5.2	5.2	5.2
XEMake	N/A	N/A	5.2
XEMake6	6.0	N/A	N/A
XKey	5.2	5.2	N/A
XMake	5.2	5.2	5.2
XNFBA	5.2	5.2	5.2
XNFCvt	5.2	5.2	5.2
XNFMAP	5.2	5.2	5.2
XNFMerge	5.2	5.2	5.2
XNFPrep	5.2	5.2	5.2
XPP	5.2	5.2	5.2
XPrint	5.2	5.2	5.2
XSimMake	5.2	5.2	5.2

Contents

Chapter 1 Introduction

Contents.....	1-1
Hardware	1-2
Software.....	1-2
Documentation.....	1-2
Online Documentation	1-2
Maintenance and Support.....	1-3

Chapter 2 System Requirements

Hardware.....	2-1
Disk Space Requirements	2-1
SPARC.....	2-1
HP-PA	2-2
Configuring Your Workstation	2-2
Synopsys Startup File and Library Setup.....	2-3
Setting the .synopsys_dc.setup File	2-3
Editing the .synopsys_vss.setup File	2-4
Analyzing the DesignWare and Simulation Libraries	2-4

Chapter 3 Features in This Release

Lookup Table Optimization	3-1
XC5200 Support.....	3-1
XC5200 X-BLOX Support	3-2
XC3000 LUT Optimization Support.....	3-3
XC3000 and XC5200 FPGA Compiler Design Flow	3-4
XC3100A -2 Speedgrade Support	3-5
Targeting XC5200 with Synopsys Version 3.2.....	3-5
Targeting XC4025 with XSI Version 5.2.....	3-5
Targeting XC4000E with XSI Version 5.2	3-6
FPGA Compiler Does Not Write Block Names as Default	3-6

Synlibs Output Enhanced.....	3-6
Support for Default Slew Rates.....	3-6
Support for Inferred Pull-up/Pull-down Control	3-7
Support for Reading XNF Netlists with Carry Logic Primitives.....	3-7
Improved X-BLOX Inference and Optimization	3-7
Enhancements/Modifications/Bug Fixes to Programs.....	3-8
XC5200 Primitives.....	3-8
AND/OR Gates	3-8
Inverter	3-9
Buffer	3-10
3-State Buffer	3-10
Decoders.....	3-10
Decoders Implementing Carry Logic.....	3-10
Resistor to VCC for Inputs, Open-Drain and 3-State Outputs .	3-11
Resistor to Ground for Inputs	3-11
Flip-Flops and Latches.....	3-11
D Flip-Flops.....	3-11
1-bit Transparent-High Latches	3-12
Clocks	3-12
Clock Buffers.....	3-12
I/O Primitives	3-12
Input Buffers.....	3-13
Output Buffers	3-13
3-State Output Buffers	3-13
Bidirectional Buffers	3-14
Special Functions.....	3-14
Boundary-Scan Logic Controller	3-14
Mode Pins	3-15
LCA Bitstream Readback Controller	3-15
Startup and Configuration Controller	3-15
Internal Clock Generator/Divider.....	3-15
Mapping Primitives.....	3-16
Flag Cells	3-17
Power/Ground.....	3-17
Carry Logic.....	3-17

Chapter 4 **Device and Package Support**

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

Chapter 5 Known Issues

Software.....	5-1
Installation.....	5-1
X-BLOX DesignWare Library Must Be Analyzed.....	5-1
Path Reference To DesignWare Libraries Has Changed....	5-2
Design Coding.....	5-2
Turn Off Density Optimization for Better EPLD Timing	5-2
Do Not Use OSC5, OSC52, and CK_DIV Cells in the XC5200 Synthesis Libraries	5-3
Design Optimization.....	5-4
XC4000A Slew Rate Selection Is Incorrect in Synopsys V3.3a	5-4
I/O Insertion	5-5
Unbonded I/Os Must Be Instantiated.....	5-5
Compile.....	5-5
Using the Enhanced LUT Mapping Algorithm	5-5
FPGA Compiler Infers ILD_1F Latches	5-6
FPGA Compiler Issues Errors if Your Design Contains Timing Loops.....	5-6
Compiler Issues Warning for Asynchronous Preset and Clear Flip-Flops	5-7
Compiler Replaces a BUFGP_F with a BUFGS_F.....	5-7
Synopsys May Infer BUFGS_F Instead Of BUFG_F.....	5-7
Write XNF File	5-8
Design Analyzer Requires Replace FPGA Command for EPLDs	5-8
Cannot Save XC5200 XNF Files from Design Analyzer Menus.....	5-8
XNF File Contains Incorrect Clock-to-Pad TimeSpecs.....	5-9
Implementation	5-9
PPR Issues Error if X-BLOX Merges Flip-Flops into IOB	5-9
PPR Issues an Error Indicating That Too Many Instances Have the Same BLKNM	5-10
XNFPrep Error When Design Exceeds Maximum Number of Clock Buffers	5-11
Timing Simulation	5-11
RAM32X1 Simulation Model Inaccuracy	5-11
Documentation.....	5-12
Synopsys (XSI) for FPGAs Interface/Tutorial Guide.....	5-12
Xilinx-supplied Hard Macros are Automatically Incorporated into Your Designs	5-12

Promdata.xnf File Must Be Copied to	
Promdata.sxnf File	5-12
Path Reference to VSS Simulation Models is Incorrect.....	5-13

Chapter 6 Xilinx Customer Support Information

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

Introduction

Welcome to the Synopsys Interface from Xilinx!

Xilinx software products have prefixes to designate the type of products you receive.

- DS = Development System (new system)
- DX = Development System Upgrade (upgrade to current system)
- SC = Support Contract (update for current system)
- SR = Support Reinstatement (update for non-current system)

The labels on the box indicate the product you have received.

This release note supports the following products.

- Xilinx Synopsys Interface (DS-401)
- Xilinx Synopsys Interface Standard Development System Package (DS-SY-STD)

Contents

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

Hardware¹

The hardware consists of the following items.

- XChecker Download and Readback Cable Set

Software

Xilinx software for all platforms is provided on CD-ROM. It consists of the following.

- Installation Program
- FPGA Core Implementation Tools (DS-502)
- Synopsys Interface and Libraries (DS-401)
- XEPLD Translator Core Tools (DS-550)
- X-BLOX (DS-380)

Documentation

The following documentation is available in print for Xilinx Synopsys Interface products.

- *Getting Started & Installation Guide*
- *Additional Products & Services Packet*
- *Synopsys (XSI) for FPGAs Interface/Tutorial Guide*¹
- *Synopsys (XSI) for EPLDs Interface/Tutorial Guide*¹

Online Documentation

The following online documentation is included with your Xilinx Synopsys Interface products.

- *Libraries Guide*
- *Libraries Supplement Guide*
- *X-BLOX Reference/User Guide*
- *HDL Synthesis for FPGAs Design Guide*
- *Floorplanner Reference/User Guide*

1. Included in DS and SR packages only.

- *Development System User Guide*
- *Development System Reference Guide, Vols 1-3*
- *Hardware & Peripherals User Guide*
- *XEPLD Reference Guide*
- *XEPLD Schematic Design Guide*

Note: Xilinx Core FPGA and EPLD documentation for Sun and PC platforms is available online via CD-ROM. Some documentation for product updates and for other workstation platforms is included on the basis of product configuration.

Selected Xilinx manuals are available in printed form. See the “Documentation Order Form” in the *Additional Products & Services Packet*.

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” chapter 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.

System Requirements

Before installing the DS-401 software, verify that your system meets the requirements listed below.

Hardware

Disk Space Requirements

The following amount of disk space is required for each option. The numbers listed below are an approximation and may not exactly match the numbers that appear on your screen during the installation process.

- XC3000 (XC3000/A/L and XC3100/A): 15 MB
- XC4000 with X-BLOX (XC4000/A/D/H): 28 MB
- XC5200: 14 MB
- XC3000, XC4000, and XC5200: 68 MB
- XC7000: 10 MB

SPARC

- Sun-4 (SPARC) workstation running SUN OS 4.1.x
- 50 MB of Swap Space

Note: 50 MB is the swap space requirement for the Xilinx tools; for Synopsys tool swap space requirement information, refer to the *Synopsys System Installation and Configuration Guide*.

- RAM: 32 MB
- Network: TCP/IP

- Display Interface: X-Windows (Xapollo, HPVue, OpenWindows)
- Disk space for full install: 94 MB
- Tape drive or CD-ROM drive

HP-PA

- HP-PA workstation running HP-UX Version 9.01 or higher
- 50 MB of Swap Space

Note: 50 MB is the swap space requirement for the Xilinx tools; for Synopsys tool swap space requirement information, refer to the *Synopsys System Installation and Configuration Guide*.

- 38 MB of Disk Space
- 4 mm cartridge tape drive, local or network-connected
- Disk space for full install: 94 MB
- Tape drive or CD-ROM drive

Configuring Your Workstation

To run the XSI software, you need to make changes to your environment as described in this section.

1. Modify your .cshrc or .login file to include the full path to the XSI executables as follows.

```
set path=(DS-401_v5.2_Dir/bin/platform $path)
```

Substitute platform for one of the following: sparc, hppa, or rs6000.

2. Add the full path to the XSI installation directory to the XACT environment variable setting.

The following table lists the environment variables you need to set depending on your system configuration/setup.

System Configuration	Environment Variable
DS-401 and XACT 5.2 are installed on same platform but different directories	<code>setenv XACT {DS401_v5.2_Dir} : {XACT_5.2_Dir}</code>
DS-401 and XACT 5.2 are installed on different platforms	On DS-401 platform, enter the following. <code>setenv XACT {DS-401_v5.2_Dir}</code>
	On the XACT 5.2 platform, enter the following. <code>setenv XACT {XACT_5.2_Dir}</code>

3. Set a new environment variable, DS401, to the full path of the XSI installation directory, as follows.

```
setenv DS401 {DS-401_v5.2_Dir}
```

Synopsys Startup File and Library Setup

Setting the .synopsys_dc.setup File

Refer to the “Getting Started” chapter of the *Synopsys (XSI) for FPGAs Interface/Tutorial Guide* for instructions on how to set up the Synopsys start-up file for FPGA designs.

For XC5200 designs, use the XC4000 .synopsys_dc.setup file, however, you must modify the link, target, symbol, and synthetic library statements. Xilinx recommends that you move all four library statements to the end of your .synopsys_dc.setup file. Since Synlibs now outputs a complete set of library definition statements, you can append the Synlibs output to the end of your .synopsys_dc.setup file.

For EPLD designs, your .synopsys_dc.setup file must contain the lines listed below. (This supercedes the “Creating Synopsys Setup Files” section in the *Synopsys (XSI) for EPLDs Interface/Tutorial Guide*.)

```
search_path = { . \
    DS401-XACT-Directory/synopsys/libraries/syn \
    SYNOPSYS-Directory/libraries/syn }
define_design_lib xc7000 -path\
    DS401-XACT-Directory/synopsys/libraries/dw/lib/epld
link_library = {xc7000.db xc7000.sldb}
target_library = {xc7000.db}
symbol_library = {xc7000.sdb}
```

```
synthetic_library = {xc7000.sldb}  
bus_naming_style = "%s<%d>"  
bus_dimension_separator_style = "><"  
bus_inference_style = "%s<%d>"  
edifout_netlist_only = true  
edifout_power_and_ground_representation = cell  
edifout_write_properties_list = {LOC}  
compile_fix_multiple_port_nets = true  
xnfout_library_version = "2.0.0"
```

Editing the .synopsys_vss.setup File

To access the VSS simulation models for the XC5200 family, add the following line to your .synopsys_vss.setup file:

```
XC5200 : $DS401/synopsys/libraries/sim/lib/xc5200
```

Analyzing the DesignWare and Simulation Libraries

DS-401 provides DesignWare libraries that support X-BLOX functions in XC4000 and XC5200 designs and high-level macro functions in XC7000 designs. DS-401 also provides simulation libraries supporting VSS. You need to analyze the DS-401 DesignWare VHDL files after you install DS-401 and before you synthesize your first Xilinx design. If you use VSS, you also need to analyze the VHDL simulation models after you install DS-401 and before you simulate your first Xilinx design. You must repeat these steps each time you install an update to your Synopsys software.

To analyze the DesignWare files, change your current directory to each of the DesignWare library source directories for FPGA and/or EPLD (whichever is applicable) and run the install_dw.dc script, as follows.

Note: In the following command, the device_family variable can be XC4000, XC5200, or XC7000.

```
cd $DS401/synopsys/libraries/dw/src/fpga/device_family  
dc_shell -f install_dw.dc  
cd $DS401/synopsys/libraries/dw/src/epld  
dc_shell -f install_dw.dc
```

The previous commands analyze the encrypted DesignWare VHDL files and place the output files into the \$DS401/synopsys/libraries/dw/lib/fpga/device_family and \$DS401/synopsys/libraries/dw/lib/epld directories.

You should analyze the DesignWare library files if you use either the VHDL or Verilog HDL languages for synthesis.

Make sure the \$SYNOPSYS environment variable is set to where the Synopsys software is installed and you have write privileges for the \$SDS401/synopsys/libraries/dw/lib/* directories.

To analyze the VSS model files, change your current directory to the simulation library source directory for each Xilinx family you are using and run the install_family.dc script as follows.

```
cd $SDS401/synopsys/libraries/sim/src/xc3000
dc_shell -f install_xc3000.dc
cd $SDS401/synopsys/libraries/sim/src/xc4000
dc_shell -f install_xc4000.dc
cd $SDS401/synopsys/libraries/sim/src/xc5200
dc_shell -f install_xc5200.dc
cd $SDS401/synopsys/libraries/dw/src/epld
dc_shell -f install_xc7000.dc
```

Note: The simulation files for the XC7000 family are located in the \$SDS401/synopsys/libraries/dw/src/epld directory.

The previous commands analyze the encrypted models and place the output files into the \$SDS401/synopsys/libraries/sim/lib/xc3000, \$SDS401/synopsys/libraries/sim/lib/xc4000, \$SDS401/synopsys/libraries/sim/lib/xc5200, and \$SDS401/synopsys/libraries/dw/lib/epld directories.

Features in This Release

This section of the release note provides information on the new features in the current software release.

Lookup Table Optimization

In the 3.3a version of the Synopsys tools, the FPGA Compiler libraries use new lookup table (LUT) optimization. For the XC3000, XC3100, XC3000A, XC3100A, and XC5200 families, these new libraries allow the FPGA Compiler to synthesize your design to a collection of lookup tables (function generators), registers, and I/O pads. Using these libraries, the compiler works directly with the building blocks of the architecture and your design is implemented in the targeted device exactly as it is synthesized.

XC5200 Support

Synopsys Design Compiler and FPGA Compiler synthesis libraries; DesignWare libraries; and VHDL System Simulator (VSS) simulation libraries are included for the new XC5200 family. Support for LUT mapping using the FPGA compiler is also provided. A complete listing of the cells in the XC5200 synthesis and simulation libraries is included in these notes.

The various translation programs and utilities have been updated for the XC5200 family. To access the new FPGA Compiler LUT cell libraries, use the default `-fc` option in Synlibs. For example, when using the FPGA Compiler to target the XC5210-5, type:

```
synlibs 5210-5
```

The Synlibs utility responds as follows:

Note: The “\” is a continuation character added for readability.

```
link_library = {xprim_5210-5.db xprim_5200-5.db \  
xgen_5200.db xfpga_5200-5.db xio_5200-5.db}  
target_library = {xprim_5210-5.db xprim_5200-5.db \  
xgen_5200.db xfpga_5200-5.db xio_5200-5.db}  
symbol_library = {xc5200.sdb}  
synthetic_library = {xblox_5200.sldb standard.sldb}
```

XC5200 X-BLOX Support

DesignWare libraries are included for the new XC5200 family. These libraries allow the FPGA Compiler or the Design Compiler to automatically infer X-BLOX modules when synthesizing XC5200 designs. Although using X-BLOX modules usually results in fast and efficient designs, the FPGA and Design compilers will select the most appropriate implementation for your design. In some cases, your design may not be implemented with X-BLOX modules.

The XC5200 DesignWare library contains the same modules as those currently available for the XC4000 family. See “Appendix B” in the *Synopsys (XSI) for FPGAs Interface/Tutorial Guide* for a listing of these modules.

To use the XC5200 DesignWare library, you must first analyze the library for the FPGA Compiler or Design Compiler version that you are currently using. See the “Analyzing the DesignWare and Simulation Libraries” section in this release note for more information. To make these libraries available to the synthesizer, add the following lines to your .synopsys_dc.setup file.

Note: The “\” is a continuation character added for readability.

```
synthetic_library = {xblox_5200.sldb standard.sldb}  
define_design_lib xblox_5200 -path \  
DS401_installation_path/synopsys/libraries/dw/lib/ \  
fpga/xc5200
```


Examples of XC5200 designs that contain X-BLOX modules are located as follows:

*DS401_installation_path/examples/synopsys/FPGA/
xc5200/vhd/calc*

*DS401_installation_path/examples/synopsys/FPGA/
xc5200/vhd/scan*

XC3000 LUT Optimization Support

New FPGA Compiler libraries have been created for the XC3000, XC3100, XC3000A, and XC3100A families with a 4-input LUT cell (xc3000_lut4) and a 5-input LUT cell (xc3000_lut5). The 3.3a version of the FPGA Compiler uses a single LUT optimization cell at a time. If more than one LUT cell is present in the target library, the compiler uses the cell with the largest number of inputs. The FPGA Compiler uses the 4-input LUT cell unless you enable the 5-input cell with the following commands:

Note: Substitute the appropriate library name for your target architecture.

```
remove_attribute xfpga_3100a-3/xc3000_lut5
dont_use

remove_attribute xfpga_3100a-3/xc3000_lut5
dont_touch

set_dont_use xfpga_3100a-3/xc3000_lut4

set_dont_touch xfpga_3100a-3/xc3000_lut4
```

Preliminary testing indicates that the default 4-input LUT cell gives the best results, however, the 5-input cell may improve critical path block level delays at the expense of using a few more function generators.

To access the new FPGA Compiler LUT cell libraries, use the default -fc option in Synlibs. For example, when using the FPGA Compiler to target the XC3142A-3, type:

```
synlibs 3142a-3
```

The Synlibs utility responds as follows:

Note: The “\” is a continuation character added for readability.

```
link_library = {xprim_3142a-3.db xprim_3100a-3.db \  
xgen_3000.db xfpga_3100a-3.db}  
target_library = {xprim_3142a-3.db xprim_3100a-3.db \  
xgen_3000.db xfpga_3100a-3.db}  
symbol_library = {xc3000.sdb}  
synthetic_library = {standard.sldb}
```

XC3000 and XC5200 FPGA Compiler Design Flow

The FPGA Compiler design flow for XC3000 and XC5200 device families is similar to the XC4000 flow. However, although the FPGA Compiler libraries are used for these families, the `Replace_fpga` command should not be used. If you try to use this command with a XC3000 or XC5200 design, the compiler issues a warning message indicating that the `Replace_fpga` command is not used for these families. If you want to convert the LUT cells to gates for schematic viewing or simulation, you can use the `Replace_fpga` command with the “-force” option. Refer to the Synopsys online command help for more information.

The default `.synopsys_dc.setup` files and default compile scripts for XC3000 and XC5200 FPGA Compiler designs are now included in the `$DS401/examples/synopsys` directory.

A suggested default compile script for XC3000 and XC5200 FPGA Compiler designs is shown here:

```
/* =====  
*/  
/* Script for Synopsys to Xilinx FPGA Compiler v3.3*/  
/* LUT Optimization Design Flow for XC3000/5200*/  
/* Xilinx June 1995 */  
/* =====  
*/  
  
TOP = <design_name>  
PART = <part_type>  
  
analyze -format vhd1 TOP + “.vhd”  
elaborate TOP  
current_design TOP
```

```
remove_constraint -all
create_clock "CLOCK" -period 50
set_port_is_pad ""
insert_pads
compile
report_fpga > TOP + ".fpga"
report_timing -nets > TOP + ".timing"
report_cell > TOP + ".cell"
write -format db -hierarchy -output TOP + ".db"
set_attribute TOP "part" -type string PART
write -format xnf -hierarchy -output TOP + ".sxnf"
exit
```

XC3100A -2 Speedgrade Support

The XC3100A synthesis and simulation libraries now contain delay and wireload information for the -2 speedgrade. Use the Synlibs command to generate the link and target library settings necessary to access these libraries, as shown in the following example.

```
synlibs 3190a-2
```

Targeting XC5200 with Synopsys Version 3.2

If you are using Synopsys Version 3.2 tools and you want to target XC5200 devices, you cannot use the libraries in this release because the database files were compiled using Synopsys Version 3.3. Database files in the gate-level Design Compiler libraries are available upon request from the Xilinx technical hotline by calling 1-800-255-7778. Since the LUT mapping in the FPGA Compiler is only available in Version 3.3, these libraries are not available for Version 3.2.

Targeting XC4025 with XSI Version 5.2

This release does not contain device-specific libraries for the XC4025. For XC4025 designs, Xilinx recommends that you compile with XC4013 libraries with the speed grade you want for the XC4025 and then change the part type in the XMake command line. Alternatively, you can set the "part" attribute in your dc_shell script to the appropriate XC4025 device (with package and speed grade).

Targeting XC4000E with XSI Version 5.2

This release does not contain device-specific libraries for the XC4000E family. However, the XSI 5.2 scripts and executables can process XC4000E designs. Contact your local FAE or the Xilinx Technical Support Hotline for more information on XC4000E synthesis and simulation libraries.

FPGA Compiler Does Not Write Block Names as Default

The FPGA Compiler does not write block names on CLB elements as the default in this release. This modification greatly improves the place and route results. In previous releases, the following line was added at the end of compile scripts just before the SXNF files were written:

```
set_attribute design "xnfout_use_blknames" -type  
boolean FALSE
```

You no longer need to use this command, however, there are no adverse effects if it is used (the attribute is set to FALSE as the default).

Synlibs Output Enhanced

Synlibs now outputs symbol library and synthetic library strings in addition to the link and target libraries. By default the appropriate X-BLOX DesignWare library is included in the synthetic library string. If you do not want to use this default, delete the X-BLOX DesignWare information from the synthetic library string.

Support for Default Slew Rates

A default slow slew rate (high slew control) is specified for the I/O libraries for all outputs unless otherwise specified.

Refer to the *Synopsys (XSI) for FPGAs Interface/Tutorial Guide* for more information on setting slew rates.

Support for Inferred Pull-up/Pull-down Control

The I/O pad cells now have attributes that allow you to infer pull-up or pull-down resistors using the `set_pad_type` command in the Synopsys tools. For example, to attach a pull-up resistor to an input port named `INPUT1`, type:

```
set_pad_type -pullup INPUT1
```

For the XC3000 families, you can only infer pull-ups. You can infer a pull-up on the pad connections of the following I/O cells: `IBUF`, `IFD`, and `ILD`.

For the XC4000 families, you can infer both pull-ups and pull-downs. You can infer a pull-up or pull-down on the pad connections of the following I/O cells: `IBUF`, `IFDI_F`, `IFDI`, `IFD_F`, `IFD`, `ILD_1`, `ILD_1F`, `ILDI_1`, `ILDI_1F`, `IOBUF`, `IOBUF_N_F`, `IOBUF_N_S`, `OBUFT`, `OBUFT_F`, `OBUFT_S`, `OFDT`, `OFDT_F`, `OFDT_S`, `OFDTI`, `OFDTI_F`, `OFDTI_S`, `BUFG_F`, `BUFGP_F`, and `BUFGS_F`.

Support for Reading XNF Netlists with Carry Logic Primitives

To improve post-implementation analysis, the `CY4` cell and 42 `CY4_mode` cells have been added to the `xgen_4000.db` library and the `CY_MUX` and `CY_INIT` cells have been added to the `xgen_XC5200.db` library. These library additions allow you to use the Read command's `-xnf` option to allow the FPGA Compiler to read post-route XNF files. After your design is read, you can perform timing analysis or synthesize your design again.

Post-route XNF files (generated with `LCA2XNF`) must be processed by `XNF2VSS` before they are read by the FPGA Compiler to avoid unresolved reference or illegal XNF record error messages.

Improved X-BLOX Inference and Optimization

The `Don't Touch` and `Don't Use` attributes have been removed from the X-BLOX cells in the common primitive libraries. This allows inferred X-BLOX components with unconnected outputs to be removed during optimization.

Enhancements/Modifications/Bug Fixes to Programs

- The Syn2XNF, XSIFIX, Synlibs, SpeedCheck, and XNF2VSS programs now support XC4025 and XC4000E devices. Note that XSI 5.2 does not include libraries for these devices. Contact the Xilinx Technical Support Hotline for information on their availability.
- Synlibs has been enhanced to provide link, target, synthetic, and symbol library information. The default library information is for the FPGA compiler. Use the -dc switch to obtain Design Compiler library information.

XC5200 Primitives

This section lists the XC5200 primitives, which include basic gates, flip-flops, latches, clock buffers, special input and output pads, I/O primitives, and special functions.

AND/OR Gates

Name	Outputs	Inputs
AND2	O	I1, I0
AND3	O	I2, I1, I0
AND4	O	I3, I2, I1, I0
AND5	O	I4, I3, I2, I1, I0
AND12	O	I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0
AND16	O	I15, I14, I13, I12, I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0
NAND2	O	I1, I0
NAND3	O	I2, I1, I0
NAND4	O	I3, I2, I1, I0
NAND5	O	I4, I3, I2, I1, I0
NAND12	O	I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0

Name	Outputs	Inputs
NAND16	O	I15, I14, I13, I12, I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0
OR2	O	I1, I0
OR3	O	I2, I1, I0
OR4	O	I3, I2, I1, I0
OR5	O	I4, I3, I2, I1, I0
OR12	O	I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0
OR16	O	I15, I14, I13, I12, I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0
NOR2	O	I1, I0
NOR3	O	I2, I1, I0
NOR4	O	I3, I2, I1, I0
NOR5	O	I4, I3, I2, I1, I0
NOR12	O	I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0
NOR16	O	I15, I14, I13, I12, I11, I10, I9, I8, I7, I6, I5, I4, I3, I2, I1, I0
XOR2	O	I1, I0
XOR3	O	I2, I1, I0
XOR4	O	I3, I2, I1, I0
XOR5	O	I4, I3, I2, I1, I0
XNOR2	O	I1, I0
XNOR3	O	I2, I1, I0
XNOR4	O	I3, I2, I1, I0
XNOR5	O	I4, I3, I2, I1, I0

Inverter

Name	Outputs	Inputs	Notes
INV	O	I	No delay

Buffer

Name	Outputs	Inputs	Notes
BUF	O	I	No delay

3-State Buffer

Name	Outputs	Inputs	Notes
BUFT	O	I, T	Synopsys tools synthesize an internal 3-state condition using BUFTs. A high-impedance state is floating unless a pull-up resistor is instantiated.

Decoders

Name	Outputs	Inputs
DECODE4 ^a	O	A<3:0>
DECODE8 ^a	O	A<7:0>
DECODE16 ^a	O	A<15:0>
DECODE32 ^a	O	A<31:0>
DECODE64 ^a	O	A<63:0>

a.Indicates that you must instantiate this primitive.

Decoders Implementing Carry Logic

Name	Outputs	Inputs
DEC_CC4	O	C_IN, A<3:0>
DEC_CC8	O	C_IN, A<7:0>

Name	Outputs	Inputs
DEC_CC16	O	C_IN, A<15:0>

Resistor to VCC for Inputs, Open-Drain and 3-State Outputs

Name	Outputs	Inputs	Notes
PULLUP	O		No delay; used for IOBs or BUFTs

Resistor to Ground for Inputs

Name	Outputs	Inputs	Notes
PULL-DOWN	O		No delay; used for IOB or BUFTs

Flip-Flops and Latches

This section lists flip-flops and latches, which include D flip-flops and 1-bit transparent-High latches.

D Flip-Flops

Name	Outputs	Inputs	Notes
FDC	Q	D, C, CLR	With Clear Direct; initial startup value is 0
FDCE	Q	D, C, CE, CLR	Clock Enable with Clear Direct; initial startup value is 0
FDPI	Q	D, C, PRE	With Preset Direct; initial startup value is 1
FDPEI	Q	D, C, CE, PRE	Clock Enable with Preset Direct; initial startup value is 1

1-bit Transparent-High Latches

Name	Outputs	Inputs	Notes
LD	Q	D, G	
LD_1	Q	D, G	Active Low Enable
LDC	Q	D, G, CLR	
LDC_1	Q	D, G, CLR	Active Low Enable
LDCE	Q	D, G, GE, CLR	
LDCE_1	Q	D, G, GE, CLR	Active Low Enable

Clocks

This section lists the clock buffer primitives.

Clock Buffers

Names	Outputs	Inputs	Notes
BUFG ^a	O	I	No pad delay included
BUFG_F	O	I	Fast implementation of clock; using dedicated pad

a.Indicates that you must instantiate this primitive.

I/O Primitives

This section lists the I/O primitives, which include input buffers, input buffers with D flip-flop, input buffers with inverted latch, output buffers, 3-state output buffers, 3-state output buffers with D flip-flop, output buffers with D flip-flop, and bidirectional buffers.

I/O buffers with flip-flops or latches are not available for the XC4000H libraries.

Input Buffers

Name	Outputs	Inputs	Notes
IBUF	O	I	
IBUF_U ^a	O	I	Unbonded pad
IBUF_F	O	I	Reduced input delay. Use when connecting to register or latch D inputs. Equivalent to nodelay attribute for XC4000.

a. Indicates that you must instantiate this primitive.

Output Buffers

Name	Outputs	Inputs	Notes
OBUF	O	I	
OBUF_F	O	I	Fast slew rate
OBUF_S	O	I	Slow slew rate
OBUF_U ^a	O	I	Unbonded pad

a. Indicates that you must instantiate this primitive.

3-State Output Buffers

Name	Outputs	Inputs	Notes
OBUFT	O	I, T	
OBUFT_F	O	I, T	Fast slew rate
OBUFT_S	O	I, T	Slow slew rate
OBUFT_U ^a	O	I, T	Unbonded pad

a. Indicates that you must instantiate this primitive.

Bidirectional Buffers

Name	Outputs	Inputs/ Outputs	Inputs	Notes
IOBUF	O	IO	I, T	Slow slew rate
IOBUF_N_F	O	IO	I, T	Fast output slew rate
IOBUF_N_S	O	IO	I, T	Slow output slew rate

Special Functions

This section lists special functions, which include the boundary scan, readback, startup, mapping, flag cells, power, and ground primitives.

Boundary-Scan Logic Controller

Name	Outputs	Inputs	Notes
BSCAN	TDO, DRCK, IDLE, SEL1, SEL2, RESET, UPDATE, SHIFT	TDI, TMS, TCK, TDO1, TDO2,	No delay
TDO		O	Output pad for BSCAN
TCK	I		Input pad for BSCAN
TDI	I		Input pad for BSCAN
TMS	I		Input pad for BSCAN

Do not connect an IBUF to the TCK, TDI, or TMS input pads. Similarly, do not connect an OBUF to the TDO output.

Mode Pins

Name	Outputs	Inputs	Notes
MD1		O	Output pad for BSCAN
MD0	I		Input pad for BSCAN
MD2	I		Input pad for BSCAN

You must connect MD0 and MD2 to an IBUF symbol. Similarly, you must connect an MD1 pad to an OBUF symbol.

LCA Bitstream Readback Controller

Name	Outputs	Inputs	Notes
READ-BACK	DATA, RIP	CLK, TRIG	No delay
RDCLK		I	No delay
RDBK	DATA, RIP	TRIG	No delay

Startup and Configuration Controller

Name	Outputs	Inputs
STARTUP	Q2, Q3, Q1Q4, DONEIN	GR, GTS, CLK

Internal Clock Generator/Divider

Name	Outputs	Inputs	Notes
OSC5	OSC1, OSC2		No delay
OSC52	OSC1, OSC2	C	No delay
CK_DIV	OSC1, OSC2	C	No delay

Mapping Primitives

Name	Outputs	Inputs	Notes
FMAP_PUC		I4, I3, I2, I1, O	Pins unlocked from signals; function generator closed to additional logic.
F5MAP_PUC		I5, I4, I3, I2, I1, O	Pins unlocked from signals; function generator closed to additional logic.
FMAP_PLC		I4, I3, I2, I1, O	Pins locked to external signals; function generator closed to additional logic.
FMAP_PUO		I4, I3, I2, I1, O	Pins unlocked from signals; function generator open to additional logic.
FMAP_PLO		I4, I3, I2, I1, O	Pins locked to external signals; function generator open to additional logic.
F5_MUX	O	DI, I2, I1	Used to connect two FMAPs to form a 5-input function.

Flag Cells

Cell	Inputs	Description
C_FLAG	I	Signal is on a critical path.
N_FLAG	I	Signal timing is not critical.
S_FLAG	I	Save signal; treat it as an external connection.
X_FLAG	I	Signal is an explicit LCA net.

Power/Ground

Name	Outputs
VCC	VCC
GND	GROUND

Carry Logic

Cell	Outputs	Inputs	Description
CY_MUX	CO	DI, CI, S	Carry chain multiplexer.
CY_INIT	COUT	INIT	Initialization element for carry chain.

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
XC2018 ^a	PC44, PC68, PC84, PG84, TQ100, VQ64	-33, -50, -70, -100, -130
XC2064 ^a	PC44, PC68, PD48, PG68	-33, -50, -70, -100, -130
XC2018L ^a	PC84, VQ64, VQ100	-10
XC2064L ^a	PC68, VQ64	-10
XC3020 ^a	CB100, CQ100, PC68, PC84, PG84, PQ100	-50, -70, -100, -125
XC3030 ^a	PC44, PC68, PC84, PG84, PQ100, TQ100	-50, -70, -100, -125
XC3042 ^a	CB100, CQ100, PC84, PG84, PG132, PP132, PQ100, TQ100	-50, -70, -100, -125
XC3064 ^{a b}	PC84, PG132, PP132, PQ160	-50, -70, -100, -125
XC3090 ^{a b}	CB164, CQ164, PC84, PG175, PP175, PQ160, PQ208	-50, -70, -100, -125
XC3020A	CB100, PC68, PC84, PG84, PQ100	-6, -7
XC3030A	PC44, PC68, PC84, PG84, PQ100, VQ64, VQ100	-6, -7
XC3042A	CB100, PC84, PG84, PG132, PP132, PQ100, TQ144, VQ100	-6, -7
XC3064A ^b	PC84, PG132, PP132, PQ160, TQ144	-6, -7
XC3090A ^b	CB164, PC84, PG175, PP175, PQ160, PQ208, TQ176	-6, -7
XC3020L	PC84	-8
XC3030L	PC84, VQ64, VQ100	-8
XC3042L	PC84, TQ144, VQ100	-8
XC3064L ^b	PC84, TQ144	-8

Device	Packages	Speed Grades
XC3090L ^b	PC84, TQ176	-8
XC3120 ^a	CB100, PC68, PC84, PG84, PQ100	-3, -4, -5
XC3130 ^a	PC44, PC68, PC84, PG84, PQ100, TQ100	-3, -4, -5
XC3142 ^a	CB100, PC84, PG84, PG132, PP132, PQ100, TQ100, TQ144	-3, -4, -5
XC3164 ^{a b}	PC84, PG132, PP132, PQ160	-3, -4, -5
XC3190 ^{a b}	CB164, PC84, PG175, PP175, PQ160, PQ208	-3, -4, -5
XC3195 ^{a b}	CB164, PC84, PG175, PG223, PP175, PQ160, PQ208	-3, -4, -5
XC3120A	CB100, PC68, PC84, PG84, PQ100	-1, -2, -3, -4, -5
XC3130A	PC44, PC68, PC84, PG84, PQ100, VQ64, VQ100	-1, -2, -3, -4, -5
XC3142A	CB100, PC84, PG84, PG132, PP132, PQ100, TQ144, VQ100	-1, -2, -3, -4, -5
XC3164A ^b	PC84, PG132, PP132, PQ160, TQ144	-1, -2, -3, -4, -5
XC3190A ^b	CB164, PC84, PG175, PP175, PQ160, PQ208, TQ176	-1, -2, -3, -4, -5
XC3195A ^b	CB164, PC84, PG175, PG223, PP175, PQ160, PQ208	-1, -2, -3, -4, -5
XC4003	PC84, PG120, PQ100	-4, -5, -6
XC4005 ^b	CB164, PC84, PG156, PQ100, PQ160, PQ208	-3, -4, -5, -6, -6B, -10
XC4006 ^b	PC84, PG156, PQ160, PQ208	-3, -4, -5, -6
XC4008 ^b	MQ208, PC84, PG191, PQ160, PQ208	-3, -4, -5, -6
XC4010 ^b	BG225, CB196, MQ208, PC84, PG191, PQ160, PQ208	-3, -4, -5, -6, -10
XC4013 ^b	BG225, CB228, MQ208, MQ240, PG223, PQ160, PQ208, PQ240	-3, -4, -5, -6, -10
XC4002A	PC84, PG120, PQ100, VQ100	-5, -6
XC4003A	CB100, PC84, PG120, PQ100, VQ100	-4, -5, -6, -10
XC4004A ^b	PC84, PG120, PQ160, TQ144	-5, -6
XC4005A ^b	PC84, PG156, PQ160, PQ208, TQ144	-4, -5, -6
XC4010D ^b	BG225, PC84, PQ160, PQ208	-5, -6
XC4013D ^b	BG225, PQ160, PQ208, PQ240	-5, -6
XC4003H	PG191, PQ208	-5, -6
XC4005H ^b	MQ240, PG223, PQ240	-5, -6
XC5202	PC84, PG156, PQ100, TQ144, VQ100	-5, -6

Device	Packages	Speed Grades
XC5204	PC84, PG156, PQ100, PQ160, TQ144, VQ100	-5, -6
XC5206 ^b	PC84, PG191, PQ100, PQ160, PQ208, TQ144, VQ100	-5, -6
XC5210 ^b	BG225, PC84, PG223, PQ160, PQ208, PQ240, TQ144	-5, -6
XC5215 ^b	HQ304, PG299, PQ208, PQ240	-5, -6
XC7236A ^a	PC44	-16, -20, -25
XC7318 ^a	PC44, PQ44	-5, -7
XC7336 ^a	PC44, PQ44	-5, -7, -10, -12, -15
XC7354 ^a	PC44, PC68	-7, -10, -12, -15
XC7372 ^a	PC68, PC84, PQ100	-7, -10, -12, -15
XC7336Q ^a	PC44, PQ44, VQ44	-10, -12, -15
XC73108 ^a	BG225, PC84, PG144, PQ100, PQ160	-7, -10, -12, -15, -20
XC73144 ^a	BG225, PQ160	-7, -10, -12, -15

a. Not supported in X-BLOX.

b. Not supported in Base packages.

Known Issues

This chapter describes the known issues and workarounds for the current software release.

Software

Installation

X-BLOX DesignWare Library Must Be Analyzed

Platform: Workstations

Architecture: XC4000/A/D/H, XC5200, XC7000

Design Step: Installation

Reference Number: Not Available

If you are not using the Synopsys compiler Version 3.3a, you must analyze the X-BLOX DesignWare files and the simulation libraries.

You need to analyze the X-BLOX DesignWare VHDL files before you can target the X-BLOX DesignWare library. To analyze the DesignWare files, refer to the README files located in the following directories:

Note: In the following command, the `device_family` variable can be XC4000, XC5200, or XC7000.

DS-401-Directory/synopsys/libraries/dw/src/fpga/*device_family*

DS-401-Directory/synopsys/libraries/dw/src/epld

You need to analyze the simulation libraries before you can target them. To analyze the simulation libraries, refer to the “Analyzing the DesignWare and Simulation Libraries” section in this release note for more information.

Path Reference To DesignWare Libraries Has Changed

Platform: Workstations
Architecture: XC4000, XC5200
Design Step: Installation
Reference Number: Not Available

The DesignWare libraries for the XC4000 and XC5200 families are separated into subdirectories within the DesignWare directory. You should make sure that any DesignWare pointers use the new paths. The documentation currently points you to the following locations:

DS401_installation_path/synopsys/libraries/dw/lib/fpga
DS401_installation_path/synopsys/libraries/dw/src/fpga

The new subdirectories are as follows:

DS401_installation_path/synopsys/libraries/dw/lib/fpga/xc4000
DS401_installation_path/synopsys/libraries/dw/lib/fpga/xc5200
DS401_installation_path/synopsys/libraries/dw/src/fpga/xc4000
DS401_installation_path/synopsys/libraries/dw/src/fpga/xc5200

Design Coding

Turn Off Density Optimization for Better EPLD Timing

Platform: Workstations
Architecture: XC7000
Design Step: Design Coding
Reference Number: Not Available

By default, the fitnet density optimization property (DENSITY_OPT) is enabled, which causes some logic paths to retain longer propagation delays than is possible in an attempt to pack more logic into the device. For many synthesis designs, density optimization produces little density benefits. Therefore, to achieve maximum performance, you should disable density optimization by specifying the NO_DENS attribute cell in your design. If the design fails to map into the desired device because it requires one or two more function blocks than available, try removing NO_DENS to see if density optimization can help fit your design.

Do Not Use OSC5, OSC52, and CK_DIV Cells in the XC5200 Synthesis Libraries

Platform: Workstations

Architecture: XC5200

Design Step: Design Coding

Reference Number: 26287

Xilinx does not recommend using the OSC5, OSC52, and CK_DIV cells in the XC5200 synthesis libraries. Due to a limitation in most synthesis tools, you cannot attach parameters or attributes to synthesized or instantiated cells. The OSC5, OSC52, and CK_DIV cells require the DIVIDE1_BY or the DIVIDE2_BY attribute to specify the appropriate clock division ratios. Currently, the FPGA Compiler and the Design Compiler write netlists that include these cells without the required attributes. This causes XNFPREP to generate an error. However, you can use the OSC5 or CK_DIV module by following these steps:

1. Go to the *DS401_install_directory*/data/synopsys/xprim_5200 directory.
2. Copy the CK_DIV.XNF or OSC5.XNF file to your working directory.
3. Add the DIVIDE1_BY or DIVIDE2_BY attributes to the XNF file, as shown below. You must use the DIVIDE1_BY attribute with the OSC1 output and the DIVIDE2_BY attribute with the OSC2 output. When your design is implemented, these XNF files are merged in and the DIVIDE1_BY and DIVIDE2_BY attributes are preserved.

The following is the original OSC5. XNF file:

```
LCANET, 6
SYM, $1I1, OSC52, SCHNM=OSC5, LIBVER=2.0.0
    PIN, OSC1, O, OSC1
    PIN, OSC2, O, OSC2
END
EOF
```

Add attributes to this file as shown. Refer to the *Libraries Supplement Guide* for the appropriate values of X and Y.

```
LCANET, 6
SYM, $1I1, OSC52, SCHNM=OSC5, LIBVER=2.0.0, DIVIDE1_BY=X, DIVIDE2_BY=Y
    PIN, OSC1, O, OSC1
    PIN, OSC2, O, OSC2
END
EOF
```

The following is the original CK_DIV.XNF file:

```
LCANET, 6
SYM, $1I1, OSC52, SCHNM=CK_DIV, LIBVER=2.0.0
    PIN, C, I, C
    PIN, OSC1, O, OSC1
    PIN, OSC2, O, OSC2
END
EOF
```

Add attributes to this file as shown. Refer to the *Libraries Supplement Guide* for the appropriate values of X and Y.

```
LCANET, 6
SYM, $1I1, OSC52, SCHNM=CK_DIV, LIBVER=2.0.0, DIVIDE1_BY=X, DIVIDE2_BY=Y
    PIN, C, I, C
    PIN, OSC1, O, OSC1
    PIN, OSC2, O, OSC2
END
EOF
```

Design Optimization

XC4000A Slew Rate Selection Is Incorrect in Synopsys V3.3a

Platform: Workstations
Architecture: XC4000A
Design Step: Design Optimization
Reference Number: 19166

In Synopsys V3.3a, the slew rate selection for XC4000A devices is incorrect. A medium slew rate maps to an output that is slow.

Refer to the *Synopsys (XSI) for FPGAs Interface/Tutorial Guide* for detailed information on slew rates.

I/O Insertion

Unbonded I/Os Must Be Instantiated

Platform: Workstations

Architecture: XC3000/A/L, XC3100/A, and XC4000/A/D/H

Design Step: I/O Insertion

Reference Number: Not Available

You must instantiate anything connected to an unbonded I/O, for example, OFD_U, or the Synopsys tools will replace it with another component, its bonded counterpart. Therefore, you must instantiate both the unbonded I/O and the primitive to which it is connected.

Compile

Using the Enhanced LUT Mapping Algorithm

Platform: Workstations

Architecture: XC3000, XC5200

Design Step: Compile

Reference Number: Not Available

Synopsys FPGA Compiler V3.3b contains an enhancement to the LUT-based optimization algorithms built into V3.3a. Investigation reveals that the enhanced algorithms are generally more effective for designs targeting Xilinx XC3000- and XC5200-family of devices. To activate the enhancement requires the following line to be added to the .synopsys_dc.setup file:

```
fpga_improved_delay_mapping=1
```

These enhancements are most effective when used with an area constraint of 0 (zero). Such a constraint can be issued from design_analyzer graphical user interface or at the dc_shell command line as follows:

```
set_max_area 0
```

FPGA Compiler Infers ILD_1F Latches

Platform: Workstations
Architecture: XC4000
Design Step: Compile
Reference Number: 27227

FPGA Compiler and Design Compiler infer the ILD_1F input latch by default. While these latches have a shorter setup-time than the preferred ILD_1 input latch, its shorter setup-time is achieved at the expense of a small hold-time. Since an external hold-time is not tolerable in many systems, this default selection may introduce problems during system integration.

The compilers default selection of the ILD_1F input latch cell can be overridden with the following command:

```
set_pad_type -exact ILD_1 {<port_name>}
```

Where *port_name* refers to a port or a list of ports which drive the data- inputs of input latches.

FPGA Compiler Issues Errors if Your Design Contains Timing Loops

Platform: Workstations
Architecture: XC4000/A/D/H
Design Step: Compile
Reference Number: 14433

The following error occurs if you have timing loops and timing constraints specified in your design. The FPGA Compiler breaks the timing loops and places Don't Touch attributes on the cells in the timing loop, which causes the Replace FPGA command to not replace all the CLBs.

```
Design contains CLBs, perform a replace_fpga first.
```

Remove the Don't Touch attributes on the cells in the timing loop before running the Replace FPGA command as follows.

```
remove_attribute find(cell, "") dont_touch  
replace_fpga
```

Compiler Issues Warning for Asynchronous Preset and Clear Flip-Flops

Platform: Workstations

Architecture: XC3000/A/L, XC3100/A, XC4000/A/D/H, XC5200

Design Step: Compile

Reference Number: Not Available

Synopsys issues the following warning message if you described an asynchronous preset and clear flip-flop in an XC4000, XC3000, or XC5200 design.

```
Warning: Target library contains no replacement for  
register 'register' (**FFGEN**). (TRANS-4)
```

This warning occurs because there are no primitives available to match these descriptions in the library, and the device architecture does not support these flip-flops.

Compiler Replaces a BUFGP_F with a BUFGS_F

Platform: Workstations

Architecture: XC4000

Design Step: Compile

Reference Number: 21544

The Synopsys compiler replaces BUFGP_F primitives with BUFGS_F primitives.

If you instantiate a BUFGP_F primitive, you must attach a Don't Touch attribute to it.

Synopsys May Infer BUFGS_F Instead Of BUFG_F

Platform: Workstations

Architecture: XC4000/A/D/H

Design Step: Compile

Reference Number: 25411

For input ports that drive clocks, Synopsys may infer a BUFGS_F cell instead of a BUFG_F cell. The BUFG_F cell is preferred because the implementation software automatically assigns it to one of the four primary global buffers (BUFGP) or to one of the four secondary global buffers (BUFGS). Although Xilinx recommends that your designs contain no more than four global buffers, eight are available with XC4000 devices. However, since Synopsys infers BUFGS_F cells,

of which there are only four, if Synopsys infers more than four global buffers, XNFPREP issues an error.

Note: Use the following commands before issuing the `Insert_pads` command.

Use the following command to prevent Synopsys from inferring a `BUFGS_F` instead of a `BUFG_F`:

```
set_pad_type -exact BUFG_F <port_name>
```

Use the following command to prevent Synopsys from inferring a clock buffer for a particular port:

```
set_pad_type -no_clock <port_name>
```

Write XNF File

Design Analyzer Requires Replace FPGA Command for EPLDs

Platform: Workstations

Architecture: XC7000

Design Step: Write XNF File

Reference Number: Not Available

If you use the FPGA Compiler and process an EPLD design using the Design Analyzer interface, you must execute the `Replace FPGA` command before you can write the XNF file, even though the command has no effect on EPLD designs.

Cannot Save XC5200 XNF Files from Design Analyzer Menus

Platform: Workstations

Architecture: XC5200

Design Step: Write XNF File

Reference Number: 25656

If are using the FPGA Compiler, and you attempt to save a design in the Design Analyzer using the `File/Save As` menu option or the `Save As` button from the FPGA Compiler tool box, Synopsys reports that the `Replace_fpga` command must be run first. If you try to use the `Replace_fpga` command with a XC5200 design, the compiler issues a warning message indicating that the `Replace_fpga` command is not

used for this device family. Therefore, to save your XC5200 designs, you must enter the following at the command line:

```
write -f xnf -h -o output_xnf_filename.sxnf
```

XNF File Contains Incorrect Clock-to-Pad TimeSpecs

Platform: Workstations

Architecture: XC3000A/L, XC3100A, XC4000/A/D/H and XC5200

Design Step: Write XNF File

Reference Number: 24744

The FPGA Compiler V3.3a may generate an XNF file that contains incorrect clock-to-pad timing specifications. If your design contains clock-to-setup and clock-to-pad specifications, the clock-to-setup specification takes precedence over the clock-to-pad specification.

If your design has clock-to-setup and clock-to-pad specifications that have different values, perform the following steps:

1. Before an XNF file is generated, turn off the Synopsys XNF writer's timing specification generator with the following command:

```
xnfout_constraints_per_endpoint = 0
```
2. Add your PPR timing specifications to a constraints file. Alternatively, you can enter default timing specifications at the PPR command line. See the PPR chapter in the *Development System Reference Guide* for more information on timing specifications.

Implementation

PPR Issues Error if X-BLOX Merges Flip-Flops into IOB

Platform: Workstations

Architecture: XC3000A/L, XC3100A, and XC4000/A/D/H

Design Step: Implementation

Reference Number: 17444

PPR issues this error message if your design contains timing specifications.

```
ERROR 7019: Qualifier "pad_name" on P2S spec doesn't  
match any input pad.
```

If a flip-flop is synthesized into a CLB flip-flop (DFF) by Synopsys, a subsequent run of X-BLOX on the design may merge the flip-flop into an IOB and thus cause PPR to issue an error as follows.

```
***PPR: ERROR 7019:  
  
Qualifier "pad_name" on P2S spec doesn't match any  
input pad.
```

You can perform one of the following.

- Remove the timing specifications from your design and include them in a constraints file.
- Remove "xlnx_hier_blknm=1" from the .synopsys_dc.setup file, and use the Uniquify command.
- To prevent X-BLOX from merging the flip-flop into an IOB, run X-BLOX with the "mergeio=false" option as shown in the following example.

```
xblox design mergeio=false
```

PPR Issues an Error Indicating That Too Many Instances Have the Same BLKNM

Platform: Workstations

Architecture: XC3000A/L, XC3100A, XC4000/A/D/H, and XC5200

Design Step: Implementation

Reference Number: Not Available

This error occurs if you have more than one occurrence of the same subdesign in your design, and the synthesis tool attached the same block name (BLKNM) parameter to all instances of the same subdesign. PPR issues this message because the logic that is being mapped to a CLB by this BLKNM attribute exceeds the amount allowed in one CLB.

To correct this problem, perform one of the following.

- Execute the following command before writing out the XNF file.

```
xlnx_hier_blknm=1
```
- Issue a Uniquify command before compiling the top level of the design.
- Perform a compile -ungroup_all.

XNFPrep Error When Design Exceeds Maximum Number of Clock Buffers

Platform: Workstations

Architecture: XC3000/A/L, XC3100/A, XC4000/A/D/H, XC5200

Design Step: Implementation

Reference Number: Not Available

This XNFPrep error occurs when you use the `insert_pads` command in Synopsys to automatically infer I/Os.

```
Error 3673: The design uses 'number' BUFGS symbols.  
The maximum allowed is: 4.
```

There are only four primary global clock buffers (BUFGP) and four secondary global clock buffers (BUFGS) in the XC4000 architecture. There is one global clock buffer (GCLK) and one alternate clock buffer (ACLK) in the XC3000 family. There are four global clock buffers (BUFG_F) in the XC5200 architecture. However, Synopsys may place more than four BUFPGs or BUFGs in an XC4000 design; more than one GCLK or ACLK in an XC3000 design; or more than four BUFG_Fs in an XC5200 design. Consequently, XNFPrep issues an error. To correct the problem, include the following lines in your script file before you compile the design, or type them in at the command line.

```
set_pad_type -no_clock ""  
set_pad_type -clock "clock_port"
```

Refer to the *Synopsys (XSI) for FPGAs Interface/Tutorial Guide* for more information.

Timing Simulation

RAM32X1 Simulation Model Inaccuracy

Platform: Workstations

Architecture: XC4000

Design Step: Timing Simulation

Reference Number: 26620

The FTGS VSS simulation model for the XC4000 RAM32X1 cell is incorrect; this can result in inaccurate timing simulation at higher clock frequencies.

Contact the Xilinx Technical Support Hotline for corrected model files for this cell.

Documentation

Synopsys (XSI) for FPGAs Interface/Tutorial Guide

Xilinx-supplied Hard Macros are Automatically Incorporated into Your Designs

Platform: Workstations
Architecture: XC4000/A/D/H
Design Step: Not Available
Reference Number: 22906

In “Appendix B” of the current version of *Synopsys (XSI) for FPGAs Interface/Tutorial Guide*, the following is stated:

“If you have Xilinx-supplied hard macros in an existing design, you must copy the appropriate XNF file from the \$XACT/data/hmllib directory to your design directory.”

In the current release, this is no longer applicable. The hard macros are automatically incorporated into your designs.

Promdata.xnf File Must Be Copied to Promdata.sxnf File

Platform: Workstations
Architecture: XC4000/A/H
Design Step: Not Available
Reference Number: 22905

In the “Using the FPGA Compiler” and the “Using the Design Compiler” chapters in the *Synopsys (XSI) for FPGAs Interface/Tutorial Guide*, the “Using MemGen” section is missing the following step:

After you have created the promdata.xnf file with the MemGen program, you must copy this file to a promdata.sxnf file.

Path Reference to VSS Simulation Models is Incorrect

Platform: Workstations

Architecture: XC3000, XC4000

Design Step: Not Available

Reference Number: 23670

In the “Simulating Your FPGA Design” chapter in the *Synopsys (XSI) for FPGAs Interface/Tutorial Guide*, the library path definitions for the FTGS models are incorrect. In the current release of the book, the path is:

XC4000 : \$DS401/synopsys/libraries/vss/lib/xc4000

XC3000 : \$DS401/synopsys/libraries/vss/lib/xc3000

Replace “vss” with “sim” to create the following correct path definitions:

XC4000 : \$DS401/synopsys/libraries/sim/lib/xc4000

XC3000 : \$DS401/synopsys/libraries/sim/lib/xc3000

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



0401434

Printed in U.S.A.