XILINX®

Release Document

Xilinx Synopsys Interface
Version 3.3
Software, Interface, and Libraries

June 1995

Read This Before Installation

Software Versions

Program	Version	Program	Version
APR	5.1	XDelay	5.1
APRLOOP	5.1	XDM	5.1
HM2RPM	5.1	XEMake	5.1
LCA2XNF	5.1	XMake	5.1
MakeBits	5.1	XNFBA	5.1
MakePROM	5.1	XNFCvt	5.1
MAP2LCA	5.1	XNFMAP	5.1
MemGen	5.1	XNFMerge	5.1
PPR	5.1	XNFPrep	5.1
XChecker	5.1	XSimMake	5.1
XDE	5.1		

XILINX®

Contents

Introduction

Welcome to the Synopsys Interface and Package from Xilinx! 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 Base, Standard, and Extended packages contain hardware, software, and documentation. Interface and Update products have software and documentation only.

Hardware[1]

- XChecker Download and Readback Cable Set

Software

Xilinx software for PC platforms is provided on CD-ROM. You can also place an order for 3 1/2" floppies.

Xilinx software for Sun4 series workstations and HP700 workstations is provided on CD-ROM. Software for other supported workstations is provided on appropriate cartridge tape media.

Documentation[1]

- Getting Started Packet
- Additional Products and Services Packet
- XACT Installation Guide
- XACT Libraries Guide
- Design Migration Guide
- Xilinx Synopsys Interface FPGA User Guide
- Xilinx Synopsys Interface EPLD User Guide
- XACT User Guide
- XACT Reference Guide, Volumes 1, 2, and 3

1. *Not included in maintenance and upgrade packages*

- XACT Hardware and Peripherals Guide
- XEPLD Design Guide
- XEPLD Reference Guide
- XEPLD Software Quick Reference Card
- XEPLD Hardware Quick Reference Card
- The Programmable Logic Data Book
- XACT Design Implementation Core Tools — Known Issues

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 last page 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.

Features in This 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 XC52000 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 and VHDL System Simulator (VSS) simulation libraries are included for the new XC5200 family, including support for LUT mapping using the FPGA compiler. 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. The default output from Synlibs specifies gate-level Design Compiler libraries. To access the new FPGA Compiler LUT cell libraries, use the -fc option in Synlibs. For example, when using the FPGA Compiler to target the XC5210-5, type:

```
synlibs -fc 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 = {standard.sldb}
```

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.

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

```
synlibs -fc 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 vhdl 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
```

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 3.3

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).

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 *XSI FPGA User 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, ILD, IOBUF, IOBUF_N_F, IOBUF_N_S, OBUFT, OBUFT_F, OFDT, and OFDT_F. BUFG_F, ACLK_F, and GCLK_F

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 XC4000 Carry Logic Primitives

The CY4 cell and 42 CY4_ mode cells have been added to the xgen_4000.db generic library for improved post-implementation analysis. This library addition allows you to use the Read command's -xnf option to allow the FPGA Compiler to read post-route XNF files. After the design is read, you can perform timing analysis or synthesize the 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 have been enhanced to support the XC5200 family.
- Syn2XNF has been enhanced to support XC7000 devices.
- XNF2VSS now supports LCANET version numbers up to 6.
- A bug that caused XNF2VSS to misinterpret a device as tristate has been fixed.
- A bug that caused a 32 x 1 RAM cell to be misinterpreted as a 16 x 1 RAM cell has been fixed.
- Synlibs has been enhanced to provide link, target, synthetic, and symbol library information for XC5200 and XC7000 families. Also, the default library information is for the Design Compiler for the XC3000 and XC5200 families, and for the FPGA Compiler for the XC4000 families. Use the -fc option to specify FPGA Compiler library information for the XC3000 and XC5200 families.

XC5200 Primitives

Table 1 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

Table 2 Inverter

Name	Outputs	Inputs	Notes
INV	O	I	No delay

Table 3 Buffer

Name	Outputs	Inputs	Notes
BUF	O	I	No delay

Table 4 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.

Table 5 Decoders

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

* Indicates that you must instantiate this primitive.

Table 6 Decoders Implementing Carry Logic

Name	Outputs	Inputs
DEC_CC4	O	C_IN, A<3:0>
DEC_CC8	O	C_IN, A<7:0>
DEC_CC16	O	C_IN, A<15:0>

Table 7 Resistor to V_{CC} for Inputs, Open-Drain and 3-State Outputs

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

Table 8 Resistor to Ground for Inputs

Name	Outputs	Inputs	Notes
PULLDOWN	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.

Table 9 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

Table 10 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.

Table 11 Clock Buffers

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

* 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.

Table 12 Input Buffers

Name	Outputs	Inputs	Notes
IBUF	O	I	
IBUF_U*	O	I	Unbonded pad

* Indicates that you must instantiate this primitive.

Table 13 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*	O	I	Unbonded pad

* Indicates that you must instantiate this primitive.

Table 14 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*	O	I, T	Unbonded pad

* Indicates that you must instantiate this primitive.

Table 15 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.

Table 16 Boundary-Scan Logic Controller

Name	Outputs	Inputs	Notes
BSCAN	TDO, DRCK, IDLE, SEL1, SEL2, RESET, UPDATE, SHIFT	TDI, TMS, TCK, TDO1, TDO2,	No delay
MD1		O	Output pad for BSCAN
TDO		O	Output pad for BSCAN
MD0	I		Input pad for BSCAN

Name	Outputs	Inputs	Notes
MD2	I		Input 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. You must connect MD0 and MD2 to an IBUF symbol. Similarly, you must connect an MD1 pad to an OBUF symbol.

Table 17 LCA Bitstream Readback Boundary-Scan Logic Controller (for Readback Function)

Name	Outputs	Inputs	Notes
RDBK	DATA, RIP	TRIG	No delay

Table 18 Readback Controller (for Readback Function)

Name	Outputs	Inputs	Notes
RDCLK		I	No delay

Table 19 Readback Function

Name	Outputs	Inputs	Notes
READBACK	DATA, RIP	CLK, TRIG	No delay

Table 20 Startup and Configuration Controller

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

Table 21 Internal 5-Frequency Clock Signal Generator

Name	Outputs	Inputs	Notes
OSC5	OSC1, OSC2		No delay

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

Table 22 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.
HMAP_PUC		I3, I2, I1, O	Pins unlocked from signals; function generator closed to additional logic.
F5_MUX	O	DI, I2, I1	Used to connect two FMAPs to form a 5-input function.

Table 23 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.

Table 24 Power/Ground

Name	Outputs
VCC	VCC
GND	GROUND

Table 25 Carry Logic

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

System Requirements

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

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

- A 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

- An 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

XSI Installation

This section explains how to install XSI DS-401 software. Your XSI software must be installed on the same platform as the Synopsys software. However, the Xilinx Development System, XACT V5.1 software, can be installed on the same network or on a different network (platform); for example, your XACT software might reside on a PC while your XSI and Synopsys tools reside on a UNIX-based workstation.

Important: The Synopsys Standard package (DS-SY-STD) is not available on the CD as a selectable package, however you can install this package by selecting its components.

To install the DS-SY-STD package, first install the XACT 5.1 core tools from the XACT 5.1 CD-ROM by selecting the DS-3PA-STD package. Next, install the DS-401 software from the XSI 3.3 CD-ROM into a different directory than the XACT 5.1 tools.

Note: If installing the DS-401 standalone package, you can ignore the licensing messages that appear on your screen during the installation process. See the "Authorization Codes and the License Manager" section in this release note for more information.

Starting the Installation Program

Installation Requirements

The DS-401 installation requires a cartridge tape or CD-ROM drive. The drive can be on the local machine or remote-login accessible (via network). The installation does not require that you are logged in as root unless you are installing the software in a directory that requires root privilege. You should be logged in as the owner of the installation directory.

Remote Tape Operation

To access a remote tape drive, establish a remote session on the machine with the tape drive using the rlogin command. It is important that the remote tape drive is the same type as your workstation. For example, if you are installing DS-401 on a Sun workstation, the tape drive must be a Sun tape drive. If you cannot mount your file system on the remote machine, perform one of the following.

- Ask your system administrator to mount your file system on the target machine and repeat the rlogin command.
- Use a remote shell to connect to the host and execute the specified commands. To access a host by means of a remote shell, you must be able to execute commands from the remote shell to the target machine and not receive any messages. The following commands should not result in any messages displayed on your screen.

  For Sun workstation installation, enter the following.

  `rsh` *remote* **`echo -n ""`**

  For HP-PA workstation installation, enter the following.

  `remsh` *remote* **`echo -n ""`**

Selecting an Installation Directory

For best results, install the DS-401 software in a new subdirectory on your system. If you are an existing DS-401 customer, it is important that you create a completely new directory for the new DS-401 software instead of overwriting the existing DS-401 or XACT directory. Installing the current release of DS-401 in a new directory allows you to validate your new installation before removing the old software. It is recommended that you select a top directory where user programs are installed, such as /usr, /usr/dev, or /usr/local, and then create a separate subdirectory for the DS-401 programs.

The default install option is to install the new DS-401 software in the ./ds401_3.3 directory.

XACT 5.1 Installed on the Same Platform

If the XACT Development System V5.1 is installed, you should install the new DS-401 Version 3.3 software in a different directory than the existing XACT directory (such as /usr/local/ds401_3.3).

XACT 5.1 Not Installed or Installed on a Different Platform

If the XACT Development System is not installed or is installed on a different platform, you must create a new directory for the installation of DS-401. If DS-401 is installed on a different platform than XACT, refer to the *Xilinx Synopsys Interface FPGA User Guide* for instructions on how to translate and transfer files between different platforms.

Program Versions

You will install the following versions of the XSI programs.

Program Name	Version
Syn2XNF	3.6.0
SEDIF2XNF	5.1
Speedcheck	3.3.0
Synlibs	3.3.0
XNFMerge	5.1
XSIFix	3.3.0
XNF2VSS	3.3.0
VMH2VSS	3.2.0
Syn2EPLD	3.2.0

Authorization Codes and the License Manager

The current version of the DS-401 software does not require any licensing, nor do you need to obtain any authorization codes to use DS-401. It is not necessary to restart your Synopsys license manager for the DS-401 V3.3 tools.

Installation Screen Messages

Licensing

DS-401 does not require a license. You can ignore the licensing messages that appear on your screen during the installation process.

If you are installing a DS-SY-STD system, refer to the *XACT Installation Guide* for licensing information.

Optional Components

There are five installation components: required data files, required binary files, optional XC3000 files, optional XC4000 files, and optional XC7000 files. The XC3000 files contain libraries for XC3000/A/L and XC3100/A devices. The XC4000 files contain libraries for XC4000/A/D/H devices. The XC5200 files contain libraries for the XC5200 devices. The XC7000 files contain libraries for all XC7000 EPLDs.

Tape Drive Specification

When you are prompted to specify if the tape drive is local or remote, enter “L” if you performed a remote login or used a remote shell.

Xilinx X Window System

You can ignore the following message during installation.

```
Before using Xilinx X Window System programs, the user
must issue the command ‘xhost +’ to allow Xilinx
programs such as xdm, xact or xdelay to connect to the
display. The most secure way to use xhost is `xhost +
<hostname>’, where <hostname> is the name of the
computer you're running on.
```

Installing From CD-ROM on a Sun UNIX System

1. Check the system requirements as defined in the “System Requirements” section of this release note.
2. Insert the CD-ROM into the drive.
3. Start a window manager (OpenWindows or X-Windows).
4. Execute the platform-specific commands.

   On a Sun4, enter the following.

   ```
   # mkdir /cdrom↵
   # mount -t hsfs -o ro /dev/sr0 /cdrom↵
   % /cdrom/install↵
   ```

   Note: The period must be typed as part of the install command.

   On an HP700, enter the following.

   ```
   # mkdir /cdrom↵
   # mount /dev/dsk/3s0 /cdrom↵
   % /cdrom/INSTALL\;1↵
   ```

   Note: Workstation users must have root privileges to use mount commands (# = root prompt, % = user prompt). Directories and device names can vary; therefore, check these names with your system administrator.

5. Follow the instructions on your screen.

   If you have enough space, Xilinx recommends that you install XSI 3.3 to a new directory. This procedure enables you to validate the new installation before removing the old software.

6. Refer to the “Setting Up the Xilinx Environment” appendix in the *XACT Installation Guide* for information on how to update your configuration file with the proper environment variables.

   Before running the software, you must install and start the network license manager. If the license manager is not running, the protected applications try to start it automatically.

7. Authorize your software using the information in the “License Management for Workstations” section in the *XACT Installation Guide*.

Installing From Tape on a Sun UNIX System

1. Log in to the target system. If you are at a node without a tape drive, log in to the machine with the tape drive as shown in the following command example.

   `/usr/ucb/rlogin` *machine_with_tape_drive*

2. Use the following command to create the new installation directory.

   `/bin/mkdir` *directory_name*

3. Move to the installation directory.

   `cd` ***full_path/****directory_name*

4. Insert the tape into the drive.
5. Load the installation script with the following commands.

   ```
   /bin/mt -f /dev/rst0 rewind
   /bin/tar xvf /dev/rst0
   ```

Note: A common name on Sun UNIX systems for a tape drive is /dev/rst0. Verify this name with your system administrator.

6. If you cannot mount your file system on a machine with a tape drive, do not rlogin. Instead, load the installation script from your system as shown in the following command. *Remote* is the remote node with the tape drive used for the installation.

   ```
   rsh remote /bin/mt -f /dev/rst0 rewind
   rsh remote /bin/dd if=/dev/rst0 bs=20b | /bin/tar
   xvf -
   ```

7. Run the installation script with the following command and follow the instructions that appear on your screen. The period and slash must be typed as part of the command.

   `./install`

8. Read the text that concludes the installation. The messages displayed by the installation program are stored in the “reminder” file. Your existing reminder file is saved as reminder.1.

Installing From Tape on an HP-PA System

1. Log in to the target system. If you are at a node without a tape drive, log in to the machine with the tape drive as shown in the following command example.

   `/usr/ucb/rlogin` *machine_with_tape_drive*

2. Use the following command to create the new installation directory.

   `/bin/mkdir` *directory_name*

3. Move to the installation directory.

   `cd` *full_path/directory_name*

Note: The software must be installed on a disk that is configured for long file names. If you attempt to install the software on disk that does not support long file names, the installation program stops and directs you to move to a different disk.

4. Insert the tape into the drive.
5. Load the installation script with the following commands.

   ```
   /bin/mt -t /dev/rmt/0m rewind
   /bin/tar Oxvf /dev/rmt/0m
   ```

   The tar options include “O” (capital O) to ensure that the installer owns the files, rather than just a Xilinx user identification.

   A common tape device name on HP-PA systems is /dev/rmt/0m for the 4-mm DAT tape drive (a zero in /dev/rmt/0m). Verify this name with your system administrator.

   The installation program expects to read from a device configured to be low density, no rewind, Berkeley-style close. Such devices often have names such as “/dev/rmt/c201d3lnb.” The last three characters are lower case LNB.

6. If you cannot mount your file system on a machine with a tape drive, do not rlogin. Instead, load the installation script from your system as shown in the following command. *Remote* is the remote node with the tape drive used for the installation.

   ```
   remsh remote /bin/mt -t /dev/rmt/0m rewind
   remsh remote /bin/dd if=/dev/rmt/0m bs=20b |
   /bin/tar Oxvf -
   ```

7. Run the installation script with the following command and follow the instructions that appear on your screen. The period and slash must be typed as part of the command.

   ```
   ./install
   ```

8. Read the text that concludes the installation. The messages displayed by the installation program are stored in the “reminder” file. Your existing reminder file is saved as reminder.1.

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_v3.3_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 list the environment variables you need to set depending on your system configuration/setup.

System Configuration	Environment Variable
DS-401 and XACT 5.1 are installed on same platform but different directories	**setenv XACT {***DS-401_v3.3_Dir***}:{***XACT_5.1_Dir***}**
DS-401 and XACT 5.1 are installed on different platforms	On DS-401 platform, enter the following. **setenv XACT {***DS-401_v3.3_Dir***}**
	On the XACT 5.1 platform, enter the following. **setenv XACT {***XACT_5.1_Dir***}**

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

   setenv DS401 {*DS-401_v3.3_Dir***}**

XSI Directory Tree Structure

After you have completed the installation, you should have the XSI directory tree structure and files shown as follows.

If you installed only the XC3000 files, XC4000 files, XC5200 files, or XC7000 files, you will not have all of the files listed in the following figure.

```
DS401_dir/
   bin/platform/
       synlibs
       syn2xnf
       sedif2xnf
       speedcheck
       xnfmerge
       syn2epld
       xnf2vss
       vmh2vss
       lib_compile
   data/synopsys/
       parttype.spd  <~44 .spd files>
       partlist.xct
       text.spd
       xmap_3000/ <~250 .xnf files>
       xmap_4000/ <~250 .xnf files>
       xmap_5200/ <~250 .xnf files>
       xprim_3000/ <~60 .xnf files>
       xprim_4000/ <~160 .xnf files>
       xprim_5200/ <~100 .xnf files>
       xprim_7000/ <~30 .xnf files>
       xunmap_3000/ <~250 .xnf files>
       xunmap_4000/ <~250 .xnf files>
       xunmap_5200/ <~250 .xnf files>
   synopsys/
       libraries/dw/lib/fpga
           <xblox_dw_modules>.o
           <xblox_dw_modules>.syn
           <xblox_dw_modules>.sim
           <xblox_dw_modules>.mra
       libraries/dw/lib/epld
           <dw_modules>.syn
           <dw_modules>.sim
           <dw_modules>.mra
       libraries/dw/src/fpga
           README
           install_dw.dc
           <xblox_dw_modules>.vhd.e
           <xblox_dw_modules>.vhd.e.update
       libraries/dw/src/epld
           README
           install_dw.dc
           install_xc7000.dc
           <dw_modules>.vhd.e
           <dw_modules>.vhd.e.update
       libraries/syn/
           xgen_family.db
```

```
            xfpga_family-speedgrade.db
            xprim_parttype-speedgrade.db
            xprim_family-speedgrade.db
            xio_parttype-speedgrade.db
            xdc_family-speedgrade.db
            xc3000.sdb
            xc4000.sdb
            xblox_4000.sldb
            xc7000.db
            xc7000.sdb
            xc7000.sldb
            xc5200.sdb
        libraries/sim/src/
            xc4000
                README
                install_xc4000.dc
                xc4000_FTGS.vhd.e
                xc4000_FTGS.vhd
            xc3000
                README
                install_xc3000.dc
                xc3000_FTGS.vhd.e
                xc3000_FTGS.vhd
            xc5200
                README
                install_xc5200.dc
                xc5200_FTGS.vhd.e
                xc5200_FTGS.vhd
        libraries/sim/lib/
            xc4000
                <vss4k_FTGS>.syn
                <vss4k_FTGS>.sim
                <vss4k_FTGS>.mra
            xc3000
                <vss3k_FTGS>.syn
                <vss3k_FTGS>.sim
                <vss3k_FTGS>.mra
            xc5200
                <vss5k_FTGS>.syn
                <vss5k_FTGS>.sim
                <vss5k_FTGS>.mra
    tutorial/synopsys/
        fpga/x4000/
            vhd
            verilog
        dc/x3000a/
            vhd
            verilog
```

```
    epld
        vhd
        verilog
    vss/xc4000
examples/synopsys/
    fc4k.synopsys_dc.setup
    dc4k.synopsys_dc.setup
    dc3k.synopsys_dc.setup
    7k.synopsys_dc.setup
    fpga.script
    dc.script
    fpga/
        xc4000/
            vhd/<design-directory>
            verilog/<design-directory>
        xc4000a/
            vhd/<design-directory>
            verilog/<design-directory>
        xc4000h/
            vhd/<design-directory>
            verilog/<design-directory>
    dc/
        xc3000/
            vhd/<design-directory>
            verilog/<design-directory>
        xc4000/
            vhd/<design-directory>
            verilog/<design-directory>
        xc4000a/
            vhd/<design-directory>
            verilog/<design-directory>
        xc4000h/
            vhd/<design-directory>
            verilog/<design-directory>
```

Synopsys Startup File and Library Setup

Setting the .synopsys_dc.setup File

Refer to the "Getting Started" chapter of the *Xilinx Synopsys Interface FPGA User Guide* for instructions on how to set up the Synopsys start-up file for FPGA designs.

For XC5200 designs, you can 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 *Xilinx Synopsys Interface EPLD User 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 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.

```
cd $DS401/synopsys/libraries/dw/src/fpga
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 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 $DS401/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. Substitute *family* with xc3000, xc4000, or xc7000 as follows.

```
cd $DS401/synopsys/libraries/sim/src/xc3000
dc_shell -f install_xc3000.dc
cd $DS401/synopsys/libraries/sim/src/xc4000
dc_shell -f install_xc4000.dc
cd $DS401/synopsys/libraries/sim/src/xc5200
dc_shell -f install_xc5200.dc
cd $DS401/synopsys/libraries/dw/src/epld
dc_shell -f install_xc7000.dc
```

Note: The simulation files for the XC7000 family are located in the $DS401 /synopsys/libraries/dw/src/epld directory.

The previous commands analyze the encrypted models and place the output files into the $DS401/synopsys/libraries/sim/lib/xc3000, $DS401/synopsys/libraries/sim/lib/xc4000, $DS401/synopsys/libraries/sim/lib/xc5200, and $DS401/synopsys/libraries/dw/lib/epld directories.

Device and Package Support

This is a master table for Xilinx devices. X-BLOX supports XC3000A/L, XC3100A/L, and XC4000 device families.

Device	Packages						Speed Grades				
XC2018	PC44	PC68	PC84	PG84	TQ100	VQ64	-33	-50	-70	-100	-130
XC2064	PC44	PC68	PD48	PG68			-33	-50	-70	-100	-130
XC2018L	PC84	VQ64	VQ100				-10				
XC2064L	PC68	VQ64					-10				
XC3020	CB100	CQ100	PC68	PC84	PG84	PQ100	-50	-70	-100	-125	
XC3030	PC44	PC68	PC84	PG84	PQ100	TQ100	-50	-70	-100	-125	
XC3042	CB100 PQ100	CQ100 TQ100	PC84	PG84	PG132	PP132	-50	-70	-100	-125	
XC3064*	PC84	PG132	PP132	PQ160			-50	-70	-100	-125	
XC3090*	CB164 PQ208	CQ164	PC84	PG175	PP175	PQ160	-50	-70	-100	-125	
XC3020A	CB100	PC68	PC84	PG84	PQ100		-6	-7			
XC3030A	PC44 VQ100	PC68	PC84	PG84	PQ100	VQ64	-6	-7			
XC3042A	CB100 TQ144	PC84 VQ100	PG84	PG132	PP132	PQ100	-6	-7			
XC3064A*	PC84	PG132	PP132	PQ160	TQ144		-6	-7			
XC3090A*	CB164 TQ176	PC84	PG175	PP175	PQ160	PQ208	-6	-7			
XC3020L	PC84						-8				
XC3030L	PC84	VQ64	VQ100				-8				
XC3042L	PC84	TQ144	VQ100				-8				
XC3064L*	PC84	TQ144					-8				
XC3090L*	PC84	TQ176					-8				
XC3120	CB100	PC68	PC84	PG84	PQ100		-3	-4	-5		
XC3130	PC44	PC68	PC84	PG84	PQ100	TQ100	-3	-4	-5		
XC3142	CB100 TQ100	PC84 TQ144	PG84	PG132	PP132	PQ100	-3	-4	-5		
XC3164*	PC84	PG132	PP132	PQ160			-3	-4	-5		
XC3190*	CB164 PQ208	CQ164	PC84	PG175	PP175	PQ160	-3	-4	-5		
XC3195*	PC84	PG175	PG223	PP175	PQ160	PQ208	-3	-4	-5		
XC3120A	CB100	PC68	PC84	PG84	PQ100		-2	-3	-4	-5	

Device	Packages						Speed Grades				
XC3130A	PC44	PC68	PC84	PG84	PQ100	VQ64	-2	-3	-4	-5	
	VQ100										
XC3142A	CB100	PC84	PG84	PG132	PP132	PQ100	-2	-3	-4	-5	
	TQ144	VQ100									
XC3164A*	PC84	PG132	PP132	PQ160	TQ144		-2	-3	-4	-5	
XC3190A*	CB164	PC84	PG175	PP175	PQ160	PQ208	-2	-3	-4	-5	
	TQ176										
XC3195A*	PC84	PG175	PG223	PP175	PQ160	PQ208	-2	-3	-4	-5	
XC4003	PC84	PG120	PQ100				-5	-6			
XC4005*	CB164	PC84	PG156	PQ160	PQ208		-4	-5	-6	-6B	-10
XC4006*	PC84	PG156	PQ160	PQ208			-4	-5	-6		
XC4008*	MQ208	PC84	PG191	PQ160	PQ208		-4	-5	-6		
XC4010*	BG225	CB196	MQ208	PC84	PG191	PQ160	-4	-5	-6	-10	
	PQ208										
XC4013*	BG225	MQ208	MQ240	PG223	PQ160	PQ208	-4	-5	-6		
	PQ240										
XC4002A	PC84	PG120	PQ100	VQ100			-5	-6			
XC4003A	CB100	PC84	PG120	PQ100	VQ100		-4	-5	-6		
XC4004A*	PC84	PG120	PQ160	TQ144			-5	-6			
XC4005A*	PC84	PG156	PQ160	PQ208	TQ144		-4	-5	-6		
XC4010D*	PC84	PQ160					-5	-6			
XC4013D	PQ160						-5	-6			
XC4003H	PG191	PQ208					-5	-6			
XC4005H*	MQ240	PG223	PQ240				-5	-6			
XC7236	PC44						-25	-30			
XC7272	PC68	PC84	PG84				-25	-30			
XC7318	PC44	PQ44					-5	-7			
XC7336	PC44	PQ44					-5	-7	-10	-12	-15
XC7354	PC44	PC68					-7	-10	-12	-15	-20
XC7372	PC68	PC84	PG84	PQ100			-7	-10	-12	-15	-20
XC73108	BG225	PC84	PG144	PQ100	PQ160		-7	-10	-12	-15	-20
XC73144	BG225	PG184	PQ160				-7	-10	-12	-15	
XC7236A	PC44						-16	-20	-25		
XC7272A	PC68	PC84	PG84				-16	-20	-25		

*Not supported in Base Packages.

Known Issues

X-BLOX DesignWare Library Must Be Analyzed

Platform: Workstations
Architecture: XC4000/A/D/H
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 *DS-401-Directory*/synopsys/libraries/dw/src/fpga and *DS-401-Directory*/synopsys/libraries/dw/src/epld directories.

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.

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.

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 *XSI FPGA User Guide* for detailed information on slew rates.

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.

The I/Os May Not Default to the Correct Slew Rate

Platform: Workstations
Architecture: XC4000/A/D/H
Design Step: I/O Insertion and Design Optimization
Reference Number: Not Available

Xilinx strongly recommends that you set your pad types to the device defaults. To set your I/O pads to the default values, type in the following command before you execute the Set Port Is Pad and Insert Pads commands.

For XC4000/A/D devices, enter the following.

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

For XC4000H devices, enter the following.

```
set_pad_type -slewrate HIGH all_outputs()
set_pad_type -voh 4.75 -vol 0.6 all_outputs()
```

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 or Clear Flip-Flops

Platform: Workstations
Architecture: XC3000/A/L, XC3100/A, XC4000/A/D/H, and 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 design, or a preset flip-flop in an 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.

Prevent Multiple OBUFs Driven by Same Net in EPLD Designs

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

If the synthesizer produces a netlist containing two or more output ports (OBUFs) driven by the same net, fitnet produces the following fatal error.

```
ppi2051:[Error] The net 'node_name' is driving multiple
output buffers...
```

If you did not intend the same signal to be driven onto multiple output pins of the EPLD, you should check for design errors or remove the extraneous output ports. If you want to implement redundant output pins, you can prevent the error by setting the following attribute in DC shell and re-compiling the design.

```
compile_fix_multiple_port_nets=true
```

Synopsys automatically inserts extra buffers before the OBUFs to produce the desired signal on each output.

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.

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 BUFGPs 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 *Xilinx Synopsys Interface FPGA User Guide* for more information.

TDO Pad is Removed by the Compiler

Platform: Workstations
Architecture: XC4000/A/D/H
Design Step: Implementation
Reference Number: 19735

To incorporate the XC4000 boundary scan capability in a configured FPGA using Synopsys tools, you must manually instantiate the boundary scan library primitives at the source code level. These primitives include TDI, TMS, TCK, TDO, and BSCAN. You must assign a Synopsys Don’t Touch attribute to the net connected from the TDO port on the boundary scan symbol to the TDO pad before you execute the Insert_pads and Compile commands.

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.

Xilinx-supplied Hard Macros are Automatically Incorporated into Your Designs

Platform: Workstations
Architecture: Not Available
Design Step: Not Available
Reference Number: 22906

In "Appendix B" of the current version of *XSI User 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/hmlib 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: Not Available
Design Step: Not Available
Reference Number: 22905

In the "Using the FPGA Compiler" and the "Using the Design Compiler" chapters in the *XSI User 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: Not Available
Design Step: Not Available
Reference Number: 23670

In the “Simulating Your FPGA Design” chapter in the *XSI User 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
  - Southwest 1-408-879-4917
  - Northwest/British Columbia 1-408-879-5150
  - Northeast/Mid-Atlantic/Canada (except British Columbia) 1-408-879-4939
  - Central 1-408-879-5321
  - Southeast 1-408-879-5383
- Europe 1-408-879-5383
- Japan 1-408-879-5321
- Southeast Asia/All Other Countries 1-408-879-4917
- Facsimile Transmission 1-408-559-0115
- International customers may also contact their local sales representative.

Technical Support

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

Training

- United States and Canada
  - Xilinx Training Administrator 1-408-879-5090
- International customers contact your local sales representative.

The Programmable Logic Company℠

2100 Logic Drive, San Jose CA 95124-3400

Tel: (408) 559-7778 FAX: (408) 559-7114

Printed in U.S.A.

PN 0401331 01