



Release Document

***XACTstep* Version 5.2/6.0**
X-BLOX

October 1995

Read This Before Installation

XACT 6.0 Install for Windows

The following information supersedes the instructions in the *Getting Started & Installation Guide* for installing XACTstep 6.0 software on PCs running Microsoft Windows. These steps minimize XACTstep 6.0 problems caused by PC resource issues and Microsoft Windows configurations that are not optimal.

Note: After installation is complete, the install program will present you with required environment variable settings for your autoexec.bat program. In some cases this may include multiple paths to the PROSeries tools. The duplicate entry will not prevent the software from running, and can be deleted.

For complete configuration instructions, please see the “Setting Up the Xilinx Environment” chapter of the *Getting Started & Installation Guide*.

1. Before starting the XACTstep 6.0 installation process, Xilinx recommends that you make backup copies of the win.ini and system.ini files located in your Windows directory.
2. Before starting Windows, run a batch file called rmwin32s.bat in DOS to remove any previous version of the Microsoft WIN32S driver that may be installed on your PC.

The latest version (v1.25.142.0) is necessary for use with XACTstep 6.0 and is compatible with all previous versions. If you are not sure if WIN32S is installed on your PC, you can still run this program. If the driver is not installed, rmwin32s.bat tries to delete certain files and then reports that these files could not be found.

To run `rmwin32s.bat`, first identify the CD-ROM drive letter, for example, `D:\`.

From the DOS command prompt, type the following:

```
d:\xbbs\utils\rmwin32s.bat
```

If the WIN32S driver is found, all associated files are removed from your system, and you are prompted to manually remove `winmm16.dll` and `device=...W32S.386` from your `system.ini` file.

3. Next, start Microsoft Windows.
4. In Windows, select **File** → **Run** from the Program Manager.
5. In the command line box, type the following:

```
d:\win32s\disk1\setup.exe
```

This step installs the latest version of the Microsoft WIN32S driver needed for XACTstep 6.0 applications.

6. Select **File** → **Run** from the Program Manager.
7. In the command line box, type the following:

```
d:\xbbs\utils\xinfo\xinfo.exe
```

XINFO is a Xilinx utility that analyzes your computer's system resources for compatibility with the XACTstep software. Review the "Hints" page for suggestions on changes that you should make to your PC configuration to allow XACTstep 6.0 to run more efficiently on your PC.

8. Select **File** → **Run** from the Program Manager.
9. To begin the installation of the XACTstep 6.0 toolset, type the following in the command line box:

```
d:\setup.exe
```

Note: Preliminary calculations of disk space requirements may be inaccurate, depending on how your hard disk is formatted. To identify required disk space accurately, the Install program must calculate disk space on the basis of the selected products list. Using Custom Install, you can correctly calculate the required disk space by turning on the Analyze Disk Space option after making your selections. Using Quick Install, simply continuing the installation process by selecting the Install button correctly calculates the available disk space.

10. After installation is complete, exit from Windows.

11. Using a text editor, load `c:\autoexec.bat`.

Certain *XACTstep* tools require that the temporary variable be set. If you do not see a line such as “`set temp=c:\temp`” in your `autoexec.bat` file, add it to this file. (The location of the temporary directory is not important; only the existence of the variable and a valid path are important.)

12. Next, reboot your PC to ensure that all environment variables have been set correctly.

Refer to the *Getting Started & Installation Guide* for information on other topics, such as environment variables and disk space requirements.

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 `acoread` command as follows:

To view Xilinx Online Documents, use the command:

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

or

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

To view Xilinx Application Information, use the command:

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

or

```
acoread /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	1-1
Software.....	1-2
Documentation.....	1-2
Online Documentation	1-2
Maintenance and Support.....	1-3

Chapter 2 Features in This Release

XC5200 Support.....	2-1
X-BLOX Reference/User Guide Changes.....	2-1
ACCUM.....	2-1
ADD_SUB.....	2-1
ANDBUS.....	2-1
BIDIR_IO	2-1
COMPARE.....	2-2
DATA_REG	2-2
INC_DEC	2-2
INPUTS.....	2-2
SHIFT	2-2
TRISTATE	2-2
SRAM	2-2
Increased Clock-to-Clock Performance	2-3
Reduced Memory Usage	2-3
Summary of X-BLOX Symbols.....	2-4

Chapter 3 Device and Package Support

Device and Package Support.....	3-1
---------------------------------	-----

Chapter 4 Known Issues

Software	4-1
Design Entry	4-1
DRC Errors Reported by XNFPrep After X-BLOX Is Run...	4-1
ONE_HOT Data on SEL Is Not Checked for Validity	4-1
Merging of TRISTATE Module into Arithmetic RPMs	4-2
TRISTATE Support	4-2
Outputs (C_OUT and OVFL) of ACCUM Are Not Registered by X-BLOX	4-2
X-BLOX Does Not Check for Incorrectly Specified Attributes	4-3
X-BLOX Does Not Check for Incorrectly Specified RLOC_RANGE Values	4-3
X-BLOX May Insert a BUFGS Sourced by an IBUF	4-3
X-BLOX Issues Error #20107 on Incorrect “PROG” Statements	4-4
Wide LFSR Counters May Take a Long Time to Process ..	4-4
INVMASK and DECODEMASK Attributes Are Insignificant	4-5
X-BLOX May Trim a Signal With a TNM Parameter	4-5
Functional Simulation Requires All X-BLOX Buses Be Named	4-5
Do Not Assign PADNAME=INPUTS or PADNAME=OUTPUTS on I/O Symbols	4-6
Data Type Propagation	4-6
X-BLOX Incorrectly Issues Warning #20315 for Smallest Negative FORCE Module Value	4-6
X-BLOX Incorrectly Issues Error #20184 About Conflicting Data Types for SEL_CODES Signal	4-7
Error # 20042 if the Lower Index on a BUS_INTERFACE Is Not 0	4-7
Translation to XNF	4-8
Suppress Generation of RLOCs if PPR Cannot Place a Design	4-8
TNMs Do Not Propagate Through BUS_IFxx, ELEMENT, and SLICE Symbols	4-8
Implementation	4-9
Processing Pre-X-BLOX V5.0 Designs with X-BLOX V5.0 or Higher	4-9
X-BLOX V5.1 or Higher Implements Counters Differently From X-BLOX V5.0	4-9

X-BLOX Issues Error # 20224 When Using a SLICE Extract	4-10
STYLE=BINARY Counter Produces an Error if COUNT_TO Is Not a Decimal Number.....	4-10
Optimizations and Improvements	4-11
X-BLOX Disregards CLBMAPs When Merging Flip-Flops Into IOBs.....	4-11
Trimming Logic Synthesized by X-BLOX	4-11
Loadless BUFGS, GCLK, or ACLK May Cause XNFPrep to Issue Error #3673	4-11
X-BLOX Trims Unconnected Nets, Error #20013.....	4-12
Simulation	4-12
Warnings Might Not Be Reported for a Subsequent Run of XSimMake on an X-BLOX Design	4-12
XSIMMAKE Incorrectly Simulates STYLE=ILD DATA_REG	4-13
Functional Simulation Error of BUS_IF Symbol in ViewSim.....	4-13
Functional Simulation	4-13
Same BUS_IF Input and Output Names Cause Contention During Functional Simulation	4-13
Documentation.....	4-14
X-BLOX Reference/User Guide	4-14
Clarification of Design Examples and Tutorials Directory....	4-14
Data Values	4-14
Pull-Up and Pull-Down Resistors	4-15
Using XACT-Performance Attributes and PROM — Programmable Read-Only Memories	4-15
COUNTER — Universal Counter	4-15
INC_DEC	4-16
INPUTS, OUTPUTS	4-16
Data Type	4-16
COMPARE — Comparators	4-17
Johnson Counter	4-17
DATA_REG and SHIFT	4-18
Shift Universal Shift Register	4-19
TRISTATE Edge Constraints LOC Support.....	4-19
Controlling the Placement of RPMs.....	4-19
ADD_SUB OVFL Description	4-20
RLOC_ORIGIN Restrictions	4-21
RLOC_ORGIN Is Ignored for the TRISTATE Symbol	4-21
LOC and LOC[i] Should Refer to TRISTATE.....	4-21

Incorrect Logic in Cascaded Counters Drawings.....	4-22
STYLE=RIPPLE for COMPARE Is Valid for XC4000	4-22
SIM=XNF and simdir Option Explanations	4-22
LOC=LOCKED.....	4-23
USE_RLOC=TRUE.....	4-23
Missing from X-BLOX Documentation	4-23
INELEM, OUTELEM, INSLICE, and OUTSLICE Modules .	4-24
Power-Up Reset and Initialization.....	4-24
COUNTER — Universal Counter.....	4-24
DATA_REG — Data Register.....	4-25
STYLE Options Override mergeio=false Option.....	4-25

Chapter 5 Xilinx Customer Support Information

Registration, Authorization, and Customer Service	5-1
Technical Support	5-1
Training	5-2

Introduction

Welcome to the X-BLOX Interface from Xilinx!

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

- DS = Development System (new system)
- SC = Support Contract (update to 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 X-BLOX (DS-380) product.

Contents

The Development System (DS) product you received contains software and documentation. Update products also have software and documentation.

Hardware

No hardware is included with the Xilinx X-BLOX Interface.

X-BLOX requires a programmable key for operation on a PC. This key is supplied with Xilinx PC (Standard, Extended, and DS-VLS-BAS) packages or Core FPGA (DS-502) software. If you purchased X-BLOX and do not already have a key, contact Xilinx Customer Service.

Software

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

- Installation Program
- X-BLOX (DS-380)

Documentation

The following documentation is available in print for Xilinx FPGA and EPLD Core products.

- *Getting Started & Installation Guide*
- *Additional Products & Services Packet*

Online Documentation

The following online documents is included with your Xilinx FPGA and EPLD Core products.

- *Libraries Guide*
- *Libraries Supplement Guide*
- *X-BLOX Reference/User Guide*
- *Floorplanner Reference/User Guide*
- *Design Manager/Flow Engine Reference/User Guide*
- *Timing Analyzer Reference User Guide*
- *Hardware Debugger Reference/User Guide*
- *PROM File Formatter Reference/User Guide*
- *Development System User Guide*
- *Development System Reference Guide, Vols 1-3*
- *Hardware & Peripherals User Guide*
- *XEPLD Reference Guide*
- *XEPLD Design Guide*
- *XEPLD Schematic Design Guide*

- *XEPLD Reference Guide (for Windows)*
- *XEPLD Schematic Design Guide (for Windows)*

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.

Features in This Release

This section discusses new and upgraded X-BLOX features.

XC5200 Support

X-BLOX supports the XC5200 device family for all platforms.

X-BLOX Reference/User Guide Changes

The following modules have changed to support the XC5200 architecture. Modules which are not included are valid as described in the *X-BLOX Reference/User Guide*. The XC5200 attribute information in this section will be incorporated in the next revision of the *X-BLOX Reference/User Guide*.

ACCUM

Attribute: STYLE may be set to ALIGNED, UNALIGNED, or RIPPLE.

ADD_SUB

Attribute: STYLE may be set to ALIGNED, UNALIGNED, or RIPPLE.

ANDBUS

Attribute: STYLE is not supported.

BIDIR_IO

Attribute: NODELAY attribute can now be attached.

COMPARE

Attribute: STYLE may be set to ARITH, TREE, or RIPPLE.
STYLE=WIRED is not supported.

DATA_REG

Attribute:

STYLE=FD is added for implementation using CLB flip-flops.

STYLE=LD is added for implementation using CLB latches.

STYLE=CLB is not supported, use STYLE=FD or STYLE=LD.

STYLE=IOB is not supported.

STYLE=ILD is not supported.

STYLE=IFD is not supported.

STYLE=OFD is not supported.

Note: For XC3000 and 4000 designs, the NODELAY attribute can be added to the DATA_REG symbol. For XC5200 designs, attach the NODELAY attribute to either the INPUTS or BIDIR_IO symbols.

INC_DEC

Attribute: STYLE may be set to ALIGNED, UNALIGNED, or RIPPLE.

INPUTS

Attribute: The NODELAY attribute can now be attached.

SHIFT

Attributes: remain unaffected.

TRISTATE

Attribute: FLOAT_VAL is not supported.

SRAM

The SRAM symbol is not supported for XC5200.

Increased Clock-to-Clock Performance

This section describes changed criteria for merging flip-flops into IOBs.

Platform: All

Architecture: XC4000A/D, XC3000A/L, XC3100A

The default criteria for merging registers into IOBs have changed to improve CLOCK to CLOCK performance. If you do not specify otherwise, X-BLOX pushes a flip-flop into an IOB where it improves the CLOCK to OUT performance according to the following rules.

- X-BLOX merges a flip-flop into an OUTFF in an IOB if one or both of the following conditions apply.
 - The flip-flop D pin is sourced (either directly or indirectly) by *non-combinational* logic. The indirect case occurs where one or more buffers and/or inverters, and no other logic, are found between the non-combinational source and the D pin.
 - The source X-BLOX symbol (if the flip-flop was generated from an X-BLOX symbol) has a parameter that indicates it should be implemented in an IOB, for example, STYLE=IOB.
- Criteria for merging flip-flops into INFFs in an IOB have not changed.

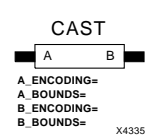
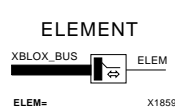
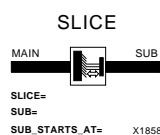
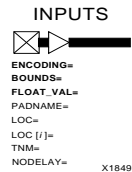
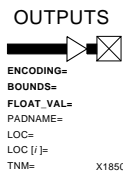
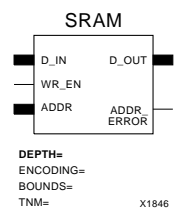
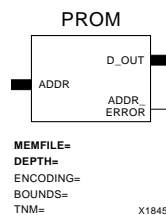
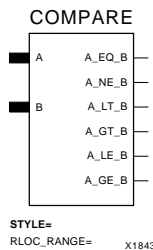
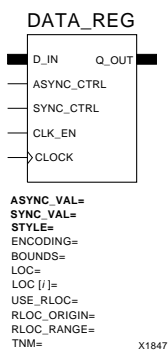
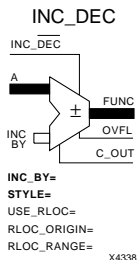
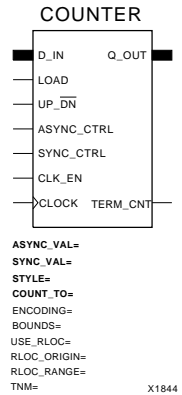
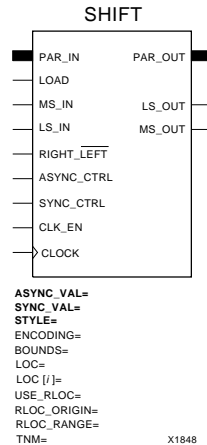
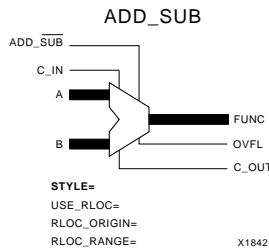
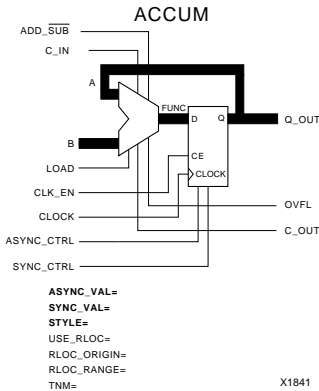
Note: In a design compiled with a previous version of X-BLOX, the clock-to-clock timing might be faster if STYLE is not specified. However, in some cases the clock-to-pad speed might be slower. If clock-to-pad speed is critical in your design, use registers in the IOB. For example, use the STYLE=IOB or STYLE=-OFD definitions for DATA_REG.

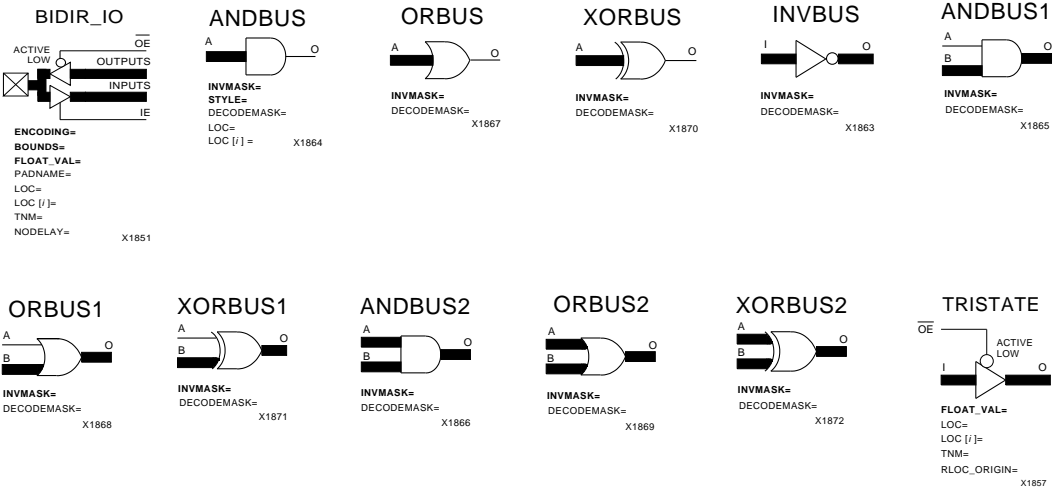
Reduced Memory Usage

For all platforms and architectures, memory saving enhancements made to the X-BLOX functional simulation mode lead to significant reductions in memory usage, by as much as 50 percent for certain styles of designs.

Summary of X-BLOX Symbols

Attributes that appear on schematic symbols are in bold type. Optional attributes appear in plain text. The X-BLOX symbols that are not displayed are valid as shown on the X-BLOX Symbols Reference Card.





Chapter 3

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 workarounds and known issues.

Software

Design Entry

DRC Errors Reported by XNFPrep After X-BLOX Is Run

Platform: All
Architecture: XC4000, XC3100A, XC3000A/L
Design Step: Design Entry
Reference Number: 16689

X-BLOX trims sourceless or loadless signals in a design. XNFPrep may report Design Rule Check (DRC) errors for the design file.

Consult the .blx report file generated by X-BLOX to see what trimming occurred prior to XNFPrep. The XNFPrep DRC error could be a result of this trimming. One example is an FMAP that is left without any inputs after X-BLOX trims away sourceless and loadless signals.

ONE_HOT Data on SEL Is Not Checked for Validity

Platform: All
Architecture: XC4000, XC3100A, XC3000A/L
Design Step: Design Entry
Reference Number: 12556

X-BLOX does not check the validity of ONE_HOT data on the select (SEL) input of the MUXBUS(x) and DECODE modules. When none of

the bits is set, or when more than one bit is set, X-BLOX fails to flag the error.

Merging of TRISTATE Module into Arithmetic RPMs

Platform: All

Architecture: XC4000

Design Step: Design Entry

Reference Number: Not Available

X-BLOX merges the TRISTATE module into arithmetic RPMs if, and only if, the following are true.

1. The arithmetic module (ADD_SUB, ACCUM, or COUNTER with STYLE=BINARY) driving the input to the TRISTATE has an RLOC_ORIGIN parameter.
2. The TRISTATE module does not have a USE_RLOC=FALSE parameter.
3. Xblox_merge_tristate=FALSE does not exist in your xactinit.dat file.

TRISTATE Support

The RLOC_ORIGIN attribute is not supported on the TRISTATE symbol on all platforms for the XC3000A/L, XC3100A, XC4000A/H/D/E, and XC5200 architectures.

Outputs (C_OUT and OVFL) of ACCUM Are Not Registered by X-BLOX

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Design Entry

Reference Number: Not Available

The carry out (C_OUT) and overflow (OVFL) outputs of the accumulator are not registered. Indication of a carry out and an overflow are lost when the accumulator is clocked.

These connections can be registered by connecting an X-BLOX DATA_REG module to each of the C_OUT and OVFL outputs.

X-BLOX Does Not Check for Incorrectly Specified Attributes

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Design Entry

Reference Number: 8143

X-BLOX ignores incorrectly specified attributes if they are not required attributes of a module. X-BLOX issues an error only if a required attribute has been incorrectly specified or misspelled.

Check the X-BLOX-generated report file, design.blx, which includes the errors reported by the software.

X-BLOX Does Not Check for Incorrectly Specified RLOC_RANGE Values

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Design Entry

Reference Number: 10289

X-BLOX does not check or report invalid RLOC_RANGE values specified in a design. PPR, however, detects these errors and reports them.

Check for errors in the report file generated by PPR. The report file is called design.rpt and includes the errors reported by PPR.

X-BLOX May Insert a BUFGS Sourced by an IBUF

Platform: All

Architecture: XC4000

Design Step: Design Entry

Reference Number: 16457

X-BLOX may put a net with a high fanout onto a BUFGS. However, in the case where the net sources both clock and non-clock pins, it adds an input buffer (IBUF) between the input pad and the BUFGS, which results in an additional delay on the net through the input buffer. The non-clock pins will not be sourced by the BUFGS.

You can manually put high fanout nets onto global buffers (BUFGS or BUFGP). There are four BUFGSs and four BUFGPs available on an XC4000 device.

X-BLOX Issues Error #20107 on Incorrect “PROG” Statements

Platform: All
Architecture: XC4000, XC3100A, XC3000A/L
Design Step: Design Entry
Reference Number: 16708

X-BLOX issues error #20107 if it encounters incorrect “PROG” statements in an XNF file created from pre-XACT 5.0 software.

Make sure that you are using Version 5.0 or higher of the translator that generates the XNF file.

Wide LFSR Counters May Take a Long Time to Process

Platform: All
Architecture: XC4000, XC3100A, XC3000A/L
Design Step: Design Entry
Reference Number: 13810

LFSR counters with the Q_OUT or D_IN pin connected to wide buses and having a small COUNT_TO value might take a long time to process. Although it might appear that your machine is hung; it is not.

To correct this problem, cascade two smaller counters using the cascading technique shown on page 4-45 of the *X-BLOX Reference/User Guide*.

INVMASK and DECODEMASK Attributes Are Insignificant

Platform: All
Architecture: XC4000, XC3100A, XC3000A/L
Design Step: Design Entry
Reference Number: 13995

The INVMASK and DECODEMASK attributes on an XORBUS2 are insignificant, because the polarity of both signals does not change the overall XOR result.

X-BLOX May Trim a Signal With a TNM Parameter

Platform: All
Architecture: XC4000
Design Step: Design Entry
Reference Number: 11431

A TNM parameter on a signal may be lost if the signal is trimmed. Make sure that you put TNM parameters only on clock pins or on those signals that you know will not be trimmed. As an alternate solution, put TNMs on symbols.

Functional Simulation Requires All X-BLOX Buses Be Named

Platform: All
Architecture: XC4000, XC3100A, XC3000A/L
Design Step: Design Entry and Simulation
Reference Number: 16608

XSimMake fails if there are unnamed X-BLOX buses or nets that are more than 1-bit wide in a schematic.

Be sure that all X-BLOX buses have names attached to them.

Do Not Assign PADNAME=INPUTS or PADNAME=OUTPUTS on I/O Symbols

Platform: All
Architecture: XC4000, 3100A, 3000A/L
Design Step: Design Entry
Reference Number: 20314

The input bus to the OUTPUTS symbol is called “outputs.” If you assign “outputs” as the value for this symbol’s PADNAME attribute, X-BLOX connects the input and output of the underlying buffer together. This connection causes contention, making the bus value indeterminate during simulation. To avoid this problem, use a PADNAME other than “outputs.” A similar problem occurs if an INPUTS symbol is given the PADNAME “inputs.”

Data Type Propagation

X-BLOX Incorrectly Issues Warning #20315 for Smallest Negative FORCE Module Value

Platform: All
Architecture: XC4000, 3100A, 3000A/L
Design Step: Data Type Propagation
Reference Number: 17974

X-BLOX issues warning message #20315 even though the FORCE symbol bounds and encoding values are within the limit bounds. Warning #20315 indicates that the symbols have attributes which exceed the data type and the most-significant bits of the value are truncated.

X-BLOX incorrectly gives this warning message for the smallest negative number that can be represented in the given bandwidth. Ignore the warning message. The .xg file contains the correct information.

X-BLOX Incorrectly Issues Error #20184 About Conflicting Data Types for SEL_CODES Signal

Platform: All
 Architecture: XC4000, 3100A, 3000A/L
 Design Step: Data Type Propagation
 Reference Number: 16859

X-BLOX incorrectly issues error #20184 about conflicting data types. Error #20184 indicates that the signal is connected to pins with conflicting data types. Use a CAST symbol between the two differently encoded buses if you need to change the data type, a bus, or slice from a bus.

Dir	Pin	Type	Symbol	Data Type
output	D_OUT	DECODE	\$1I174	one_hot(7:0)
input	XBLOX_BUS	ELEMENT	\$1I176/ELEM_7	ubin(7:0)
input	XBLOX_BUS	ELEMENT	\$1I176/ELEM_6	ubin(7:0)

In the example in the “CAST Symbol Data Types” table, X-BLOX infers the output of DECODE is one-hot, then infers the ELEMENTs as UBIN. To correct this problem, attach a BUS_DEF to the output of DECODE or use a CAST symbol between the two differently encoded buses.

Error # 20042 if the Lower Index on a BUS_INTERFACE Is Not 0

Platform: All
 Architecture: XC4000, XC3100A, XC3000A/L
 Design Step: Data Type Propagation
 Reference Number: 15033

This error occurs if the lower index of the bus that you connected to the X-BLOX bus port of a bus interface (BUS_IFxx) symbol — which is used when converting an X-BLOX bus to a non-X-BLOX bus and vice-versa — is not 0. This error means that the BOUNDS on that bus must be x:0, where x is the upper index.

If you need to interface to an X-BLOX bus with bounds other than x:0, insert a CAST symbol between the X-BLOX bus and the BUS_IF symbol to translate the bus into one that has bounds of x:0 or create a custom BUS_IF macro. For more information on the CAST symbol,

refer to the “CAST — Data Type Symbol” section of the “Module Definitions” chapter of the *X-BLOX Reference/User Guide*. For information on custom BUS_IF macros, refer to the “Creating a Custom BUS_IFxx Macro” section of the “Creating an X-BLOX Design” chapter of the *X-BLOX Reference/User Guide*.

Translation to XNF

Suppress Generation of RLOCs if PPR Cannot Place a Design

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Translation to XNF

Reference Number: 15564

If PPR cannot place your design, suppress the automatic generation of RLOC sets on X-BLOX modules that do not generate carry logic; use the command-line option “REG_RLOCS=FALSE”. Although it improves place and route results for most designs, the automatic generation of these parameters can create a placement problem that cannot be resolved by PPR. This option does not affect user-generated RPMs in the design. As an alternative to using this option, you can selectively attach a USE_RLOC=FALSE attribute to X-BLOX modules for which you want to suppress the automatic generation of RLOCs.

TNMs Do Not Propagate Through BUS_IFxx, ELEMENT, and SLICE Symbols

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Translation to XNF

Reference Number: Not Available

X-BLOX does not propagate TNMs and TSids attributes through BUS_IFxx, ELEMENT, and SLICE symbols. These attributes must be placed on buses ahead of these symbols in the data stream to propagate properly. Refer to the “TMN Attribute” section of the “Creating an X-BLOX Design” chapter of the *X-BLOX Reference/User Guide* for more information.

Implementation

Processing Pre-X-BLOX V5.0 Designs with X-BLOX V5.0 or Higher

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Implementation

Reference Number: Not Available

For existing designs created with pre-XACT 5.0 libraries:

Viewlogic — Always include the shm4000 library in your view-draw.ini file, whether or not you used any hard macros from the standard Xilinx hard macro library. In addition, if you created any custom hard macros, you should run HM2RPM on each one of them before processing the design.

Mentor Graphics — Run pld_men2xf8 then pld_fncsim8 or pld_timsim8 (for functional and timing simulation, respectively). If your design contains hard macros that you built, make sure you run HM2RPM on each of them before processing your design.

OrCAD — If your design contains hard macros that you built, run HM2RPM on each of them before processing the design.

X-BLOX V5.1 or Higher Implements Counters Differently From X-BLOX V5.0

Platform: All

Architecture: XC4000

Design Step: Implementation

Reference Number: 18389

Counters with six bits or more are implemented by using RPMs with carry logic. A COUNTER with the STYLE, BINARY and USE_RLOC attributes not set, does not use carry logic on XC4000 designs.

However, the counter uses more CLBs and timing predictability decreases.

X-BLOX Issues Error # 20224 When Using a SLICE Extract

Platform: All
Architecture: All
Design Step: Implementation
Reference Number: 19537

X-BLOX issues error # 20224 which reads:

```
XBLOX: ERROR 20224:  
INTERNAL ERROR  
Please contact Xilinx Technical Support. Please  
provide the following details to the support  
personnel. This information will be needed by the  
software developers.  
  
1. representation_error(is(_10269,+(abs(-(-  
2147483647,2147483647)),1)),2,' integer overflow')
```

This error can occur if a SLICE symbol is used to extract a single bit (bit 0) from a bus. To correct this problem, change the SLICE symbols to an ELEMENT.

STYLE=BINARY Counter Produces an Error if COUNT_TO Is Not a Decimal Number

Platform: All
Architecture: All
Design Step: Implementation
Reference Number: 17072

If a counter STYLE attribute is set to BINARY and COUNT_TO is not a decimal number, X-BLOX issues an error message.

Optimizations and Improvements

X-BLOX Disregards CLBMAPs When Merging Flip-Flops Into IOBs

Platform: All

Architecture: XC3100, 3000A/L

Design Step: Optimization and Improvements

Reference Number: 19382

X-BLOX does not look at a CLBMAP during architecture optimization. It tries to merge DFF into an IOB even if an explicit CLBMAP on that flip-flop exists, which causes XNFMAP to fail on the net.

To correct this problem, place the “X” flag on the net between the flip-flop and its I/O buffer.

Trimming Logic Synthesized by X-BLOX

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Optimization and Improvements

Reference Number: 12227

X-BLOX synthesizes logic from the X-BLOX modules used in the design but does not trim logic from the resulting design. XNFPrep, which is run after X-BLOX, will trim any redundant logic, and as a result, you might notice a significant amount of logic trimming, mainly buffers and inverters, occurs after the design has been synthesized by X-BLOX.

Look at the report file generated by XNFPrep. The report is called design.prp and contains a record of the nets and logic that were trimmed after X-BLOX was run.

Loadless BUFGS, GCLK, or ACLK May Cause XNFPrep to Issue Error #3673

Platform: All

Architecture: XC4000, XC3000A, XC3100A

Design Step: Optimization and Improvements

Reference Number: 15900

If your design contains a loadless BUFGS, X-BLOX can introduce up to four more BUFGS symbols in your design, bringing the total

number over four. These additional BUFGS symbols cause XNFPrep to issue Error #3673. The loadless BUFGS should have been trimmed away, leaving the total number of BUFGS symbols at four. The same is true for a GCLK or an ACLK buffer in the XC3000A family.

Correct the loadless BUFGS, GCLK, or ACLK in your design by either removing it entirely or connecting a load to it. Then, reprocess your design.

X-BLOX Trims Unconnected Nets, Error #20013

Platform: All

Architecture: All

Design Step: Optimization and Improvements

Reference Number: 19176

To prevent X-BLOX from trimming unconnected nets (for example, in an iterative design flow), attach an S flag to those nets and add a `savesig=TRUE` option in XNFPREP.

Simulation

Warnings Might Not Be Reported for a Subsequent Run of XSimMake on an X-BLOX Design

Platform: All

Architecture: XC4000, XC3100A, XC3000A/L

Design Step: Simulation

Reference Number: 16445

X-BLOX warnings reported for an initial run of XSimMake might not be reported for a subsequent run of XSimMake, which does not mean the warnings do not still apply.

To see all warnings for a design, rerun XSimMake, using the `-r` option that forces a re-execution of all programs. This option forces a regeneration of the X-BLOX simulation models, which ensures that all warning messages are reported.

XSIMMAKE Incorrectly Simulates STYLE=ILD DATA_REG

Platform: All
Architecture: All
Design Step: Simulation
Reference Number: 17355

XSIMMAKE incorrectly simulates STYLE=ILD DATA_REG during functional simulation. These level-sensitive latches behave like edge-triggered flip-flops during simulation.

Functional Simulation Error of BUS_IF Symbol in ViewSim

Platform: All
Architecture: All
Design Step: Simulation
Reference Number: 19151

When the output of a TRISTATE module is the input of a BUS_IFXX symbol, erroneous values are reported after functional simulation, when the TRISTATE is inactive. The output of the TRISTATE, “z,” is correct, but the output of the BUS_IF symbol is, “x,” an unknown. A similar result occurs with OBUFs.

Performing timing simulation (instead of functional) resolves this problem, because the BUS_IF symbol no longer exists in the underlying simulation model.

Functional Simulation

Same BUS_IF Input and Output Names Cause Contention During Functional Simulation

Platform: All
Architecture: All
Design Step: Functional Simulation
Reference Number: 18693

Contention occurs during functional simulation if the BUS_IF symbol is used to interface an X-BLOX output to a Viewlogic bus and the input and output buses have the same name.

To correct this problem, change the name of either the input or the output bus of BUS_IF.

Documentation

X-BLOX Reference/User Guide

Clarification of Design Examples and Tutorials Directory

Platform: All
Architecture: Not Available
Design Step: All
Reference Number: 10733

Directories of design examples and files (page 1-2) you use to run the tutorial are provided for your convenience. The path for the design examples is \$XACT/examples/interface design. The directory \$XACT/examples contains a README file that documents the example designs in that directory. The path for the tutorial files is \$XACT/tutorial/interface/design. The directory \$XACT/tutorial contains a README file that documents the tutorials in that directory.

Data Values

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10513, 14915

In the fifth paragraph of the “Data Values” section in the “Customizing an X-BLOX Module” section of the “Creating an X-BLOX Design” chapter, the last part of the sentence “but don’t care digits are not allowed in twos complement base” should start the following, new sentence: “Don’t care digits are allowed in numbers with radices that are powers of two.”

The sixth paragraph states, “To represent fractional data, a radix point (period) can be used. For example, 2#100.11# represents 4.75.” It should read, “X-BLOX version 5.0 ignores the decimal point and all characters that follow it. Only the integer portion of a numeric value is used and is right-aligned in the available bits. For example, 4.75 for

ENCODING=UBIN, BOUNDS=3:-2 yields a bit pattern of 000100 because the .75 is ignored and the value 4 is represented in 6 bits as 000100, which corresponds to the value of 1. To get the value 4.75, scale your constant by the number of fraction bits, that is 4.75 scaled by two fraction bits is $4.75 \times 4 = 19$. This value is represented as 010011 in 6 bits.” This description applies Reference #10513 on page 2-7.

The preceding description supersedes the descriptions in the “Big-Endian vs. Little-Endian” section within the “Bus Data Types” section of the “Creating an X-BLOX Design” chapter as well. (Reference #14915) on pages 2-18 and 2-19.

Pull-Up and Pull-Down Resistors

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10525

On page 2-12, “16#?3#” should be “4#?3#” in the first paragraph of the “FLOAT_VAL Attribute” section.

Using XACT-Performance Attributes and PROM — Programmable Read-Only Memories

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10531

On pages 2-33 and 4-86, respectively, the references to TNMs on PROMs are incorrect. X-BLOX issues Error #20138 if a TNM parameter has been specified on a PROM module. Remove all TNM parameters on X-BLOX PROM modules.

COUNTER — Universal Counter

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10661

On page 4-40, the description of the binary COUNT_TO attribute values should read as follows, “The COUNT_TO attribute values are any number between 2 and 2^n inclusive.”

INC_DEC

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10637, 10645

On page 4-64 (Reference #10637), the first heading under “Inputs” should read INC_DEC, not NC_DEC.

On page 4-65 (Reference #10645), the second line of the FUNC output description refers to a B port. The INC_DEC module does not have a B port. This sentence should read, “The data type of this port is the same as the A input port.”

INPUTS, OUTPUTS

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10628

On pages 4-70 and 4-82 the third sentence under the “Constraints File” section of both modules should read:

“However, if you specified the PADNAME attribute, the input pads of the INPUTS module — or output pads of the OUPUTS module — have names of the form SYM/PADNAME<0>, SYM/PADNAME<1>, SYM/PADNAME<-1>...”

Data Type

Platform: All
Architecture: Not Available
Design Step: Data Type Propagation
Reference Number: 10557

On page 2-16, the first Data Type paragraph states that both the BOUNDS and ENCODING attributes must be defined to establish a data type. In fact, it is possible to define only one of the two attributes. The second note on page 2-17 explains the different settings.

COMPARE — Comparators

Platform: All

Architecture: Not Available

Design Step: Data Type Propagation

Reference Number: 10522, 10540

The note at the bottom of the page 4-33 (Reference #10522) states that only the A_EQ_B and A_NE_B outputs can be connected when the encoding of the buses attached to the COMPARE module is ONE_HOT. This note is no longer true. Any or all of the output pins can be connected when the ENCODING is ONE_HOT.

On page 4-34 (Reference #10540), the RIPPLE style applies to all X-BLOX designs for devices supported by X-BLOX, not just to the XC3000A/L and XC3100A designs.

The ARITH style description should read, “Builds the COMPARE module from arithmetic modules. For the XC4000 family, fast carry logic is used.”

Johnson Counter

Platform: All

Architecture: Not Available

Design Step: Data Type Propagation

Reference Number: 10440

On page 4-43, the paragraph describing Johnson Counters, starting with the sentence, “One bit in the count sequence changes per clock cycle,” should read as follows. “One bit in the count sequence changes per clock cycle if the default COUNT_TO=2n is used. If COUNT_TO is assigned by the user to 2n – 1, the 2 bits changes during one clock cycle in the middle of the sequence, as in the following 3-bit example with COUNT_TO=5.

```
0 0 0
1 0 0
1 1 0
0 1 1
0 0 1"
```

DATA_REG and SHIFT

Platform: All

Architecture: Not Available

Design Step: Data Type Propagation

Reference Number: Not Available, 10548 (for page 4-54)

The descriptions for the LOC and LOC[i] attributes on pages 4-51 and 4-98 are incorrect.

The LOC attribute description should read, “Use the LOC attribute to place the module flip-flops in a specific CLB or IOB location.”

The following are examples of location constraints for XC4000 devices.

```
LOC=BL or LOC=P13 for an IOB
LOC=CLB_R1C1.X for a CLB
```

The following are examples of location constraints for XC3000A/L and XC3100A devices.

```
LOC=P13 for an IOB
LOC=BC for a CLB
```

Use the LOC[i] attribute to specify the LOC attribute for a desired module instance. For example, if the module refers to LOC[0]=CLB_R1C1, the module instance 0 should be constrained to the CLB at column 1 of row 1 on the device. Refer to the *Development System Reference Guide* for information on the XC3000 syntax and to the *Libraries Guide* for the XC4000 syntax.

In Table 4-13, on page 4-54 (Reference #10548) row FAST, MEDFAST... in the Attributes column, the text in the IOB column should read, “associated with output flip-flop.” On page 4-55, change the second sentence under FAST, MEDFAST, MEDSLOW, or SLOW as follows, “These attributes are associated with the IOB output register created by X-BLOX.”

Shift Universal Shift Register

Platform: All

Architecture: Not Available

Design Step: Data Type Propagation

Reference Number: 10543

On page 4-95, in the description of MS_IN, change “Circular Style” to “Circular or Arith styles.” The table entry for STYLE=ARITH on page 4-97 should read, “The MSB is the sign bit. X-BLOX issues an error if MS_IN is connected when this style is used.”

TRISTATE Edge Constraints LOC Support

Platform: All

Architecture: Not Available

Design Step: Data Type Propagation

Reference Number: 10513

On page 4-111, the description for the LOC attribute is incorrect. The TRISTATE module does not support edge constraints of the type LOC=TL. Instead, location constraints must be specified in the following format.

```
XC4000 devices: LOC=TBUF_R2C3.1
XC3000 devices: LOC=TBUF_BC.1
```

Controlling the Placement of RPMs

Platform: All

Architecture: Not Available

Design Step: Data Type Propagation

Reference Number: 10563, 10518

On page 5-3 (Reference #10563, 10518), the description of the USE_RLOC attribute should read, “This attribute is recognized only in XC4000 families by the ADD_SUB, ACCUM, INC_DEC, SHIFT, DATA_REG, COMPARE, and COUNTER modules. It has no effect on XC3000A/L or XC3100A designs.”

For all of the previously mentioned modules, the following rules apply.

- When the USE_RLOC attribute is set to FALSE, RPMs are not generated, which implies that carry logic is not used.

- When the USE_RLOC attribute is set to TRUE, X-BLOX generates RPMs. Carry logic is used where appropriate.
- If USE_RLOC is not set, X-BLOX chooses the fastest implementation depending on the bus width.

These rules apply to the COUNTER module when it uses the BINARY style.

- If USE_RLOC is set to FALSE, X-BLOX does not generate RPMs nor does it use carry logic.
- If USE_RLOC is set to TRUE, X-BLOX always generates RPMs and uses carry logic.
- If USE_RLOC is not set, X-BLOX chooses the fastest implementation style depending on the bus width. If the bus width is 6 bits or more, X-BLOX generates RPMs and uses carry logic.

If the COUNTER style is other than BINARY and USE_RLOC is FALSE, X-BLOX does not implement the flip-flops as an RPM; that is, no RLOCs are placed on the generated flip-flops. By default, RLOCs are placed on the flip-flops to force their placement in the same column. It is redundant to set this attribute to TRUE if the COUNTER style is other than BINARY.

The RLOC_ORIGIN description should read, “RLOC_ORIGIN is recognized by the ADD_SUB, ACCUM, INC_DEC, DATA_REG, SHIFT, COUNTER, and TRISTATE modules.”

The RLOC_RANGE description should read, “This attribute is recognized by the ADD_SUB, ACCUM, COMPARE, DATA_REG, INC_DEC, SHIFT, and COUNTER modules.”

ADD_SUB OVFL Description

Platform: All

Architecture: Not Available

Design Step: Data Type Propagation

Reference Number: 10812

On page 4-15, the second sentence of the OVFL section should read, “The OVFL output is Low when the result of the operation exceeds the precision of the adder/subtractor.” If the module is in subtract mode and the inputs are UBIN, the OVFL = C_OUT. In subtract mode, C_OUT is Low.

RLOC_ORIGIN Restrictions

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10536

On page 6-10, the first sentence should read, “An RLOC_ORIGIN must be specified such that the resulting RPM can be placed in the given part type.”

RLOC_ORIGIN Is Ignored for the TRISTATE Symbol

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10790

On page 4-112, the RLOC_ORIGIN attribute is ignored for the TRISTATE symbol, because X-BLOX does not create RPMs for it.

LOC and LOC[i] Should Refer to TRISTATE

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10791

On page 4-111, the LOC and LOC[i] descriptions refer to single and multiple input/output pad-buffers, but should refer to TRISTATE buffers. The LOC description should read, “The LOC attribute specifies the pin location for a single 3-state buffer, for example: LOC=A1, or the attribute specifies the placement of all TBUFs on a specific edge or corner of the chip, for instance, LOC=TL for the top-left corner of the chip.”

The LOC[i] description should read, “The LOC[i] attribute specifies the pin locations for multiple 3-state buffer pair locations, for example: LOC[6]=A11, LOC[7]=P9.”

Incorrect Logic in Cascaded Counters Drawings

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10772

On pages 4-4 and 4-46, in the “Cascading Counters with Clock Enable” and “Cascading Counters without an Initial Clock Enable” figures in the “ONE_HOT” section of the “COUNTER — Universal Counter” chapter, the logic connected to the net labeled, “To Next CE” does not look at the TERM_CNT of both counters. The second counter in each figure, the TERM_CNT should not be High for one cycle.

STYLE=RIPPLE for COMPARE Is Valid for XC4000

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10769

STYLE=RIPPLE is a valid Compare module style for XC4000 designs. On page 4-34, the “COMPARE — Implementation Styles” table in the “STYLE” section of the “COMPARE — Comparators” chapter, incorrectly indicates that it applies to only XC3000 and 3100A/L designs.

SIM=XNF and simdir Option Explanations

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10705

On page A-2, the SIM=XNF and simdir command-line option descriptions should both read, “This option is not used in the implementation flow, and is provided only for use by simulation interface programs (such as XSIMMAKE). It is not available from XDM.

LOC=LOCKED

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10700

On page 6-6, the second and third paragraphs of the “Global Buffers” section of the “Understanding X-BLOX Operations” chapter should be removed.

USE_RLOC=TRUE

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10664

The current X-BLOX Reference/User Guide reads (page 4-41), “It is redundant to set this attribute to TRUE.” Carry logic is only implemented with counters with six bits or more. If the counter is five bits, it is not redundant to use RLOC=TRUE, because a five-bit counter defaults to non-carry logic.

Missing from X-BLOX Documentation

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10523

SET, RESET, VCC, GND, ALL_0, and ALL_1 are valid values for the following X-BLOX attributes: COUNT_TO, ASYNC_VAL, SYNC_VAL, INC_BY, FLOAT_VAL, INVMASK, and DECODE-MASK.

- RESET, GND, and ALL_0 set the attribute to a bit pattern of all zeroes.
- SET, VCC, and ALL_1 set the attribute to a bit pattern of all ones.

INELEM, OUTELEM, INSLICE, and OUTSLICE Modules

Platform: All
Architecture: Not Available
Design Step: Design Entry
Reference Number: 10617

The INELEM, OUTELEM, INSLICE, and OUTSLICE modules are not documented in the *X-BLOX Reference/User Guide*. (Page 4-67 and 4-83)

Power-Up Reset and Initialization

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10449, 10512

The following should be added to the “Power-Up and Initialization” paragraph (Reference #10449, page 2-9), “When the Global Set Reset (GSR) is activated, the register-based modules, ACCUM, COUNTER, DATA_REG, and SHIFT, are loaded with the setting given by the ASYNC_VAL attribute. The same is true for the Global Reset (GR) in the XC3000A/L devices.”

The description for making custom X-BLOX bus interface symbols on page 2-29 (Reference #10512) does not explain that custom bus interface symbols are needed whenever you use a bus that does not start or end at index 0 due to the fact that the ELEMENTs within the Xilinx-supplied symbols are called out starting at ELEM=0.

COUNTER — Universal Counter

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10610, 10563

On page 4-39 (Reference #10610), the third sentence of the ASYNC_VAL attribute description and the second sentence of the SYNC_VAL attribute description should read as follows, “If this attribute is not specified, a default value of one is used for ONE_HOT counters, and a default value of zero otherwise.”

On page 4-41 (Reference #10563), the description of the USE_RLOC attribute is incomplete. For the full description, refer to reference #10518 “Controlling the Placement of RPMs” in this chapter.

DATA_REG — Data Register

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10511

The Q_OUT output port of the DATA_REG module must be connected or XNFPrep issues an error message. (Page 4-50)

STYLE Options Override mergeio=false Option

Platform: All
Architecture: Not Available
Design Step: Implementation
Reference Number: 10735

The mergeio=false command-line option (pages 4-55, 56, and 57) is overridden by STYLE=CLB, IOB, ILD, IFD, and OFD. The following notes should be added for the corresponding implementation styles. The command-line option overrides some values of the STYLE attribute, but not others.

- STYLE=CLB

For STYLE=CLB, X-BLOX implements a DATA_REG module in CLBs only. This attribute value overrides the mergeio=false command-line option.

- STYLE=IOB, STYLE=ILD, STYLE=IFD, and STYLE=OFD

The STYLE=*attribute* overrides the mergeio=false command-line option.

- STYLE=none

No changes are necessary. It is already clear that the mergeio=false command-line option overrides the attribute.

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



0401426

Printed in U.S.A.