PCI 64/32 Interface v3 and v4

Features

- Fully PCI 3.0 compliant LogiCORE™, 64/32-bit, 66/33 MHz interface
- Customizable, programmable single-chip solution
- Pre-defined implementation for predictable timing
- Incorporates Xilinx Smart-IP™ technology
- 3.3V operation at 0–66 MHz
- 5.0V operation at 0–33 MHz
- Fully verified design tested with Xilinx proprietary test bench and hardware
- Available through the Xilinx CORE Generator™ v8.2i with applicable service pack
- CardBus compliant
- Supported initiator functions:
  - Configuration read, configuration write
  - Memory read, memory write, MRM, MRL
  - Interrupt acknowledge, special cycles
  - I/O read, I/O write
- Supported target functions:
  - Type 0 configuration space header
  - Up to three base address registers (MEM or I/O with adjustable block size from 16 bytes to 2 GB)
  - Medium decode speed
  - Parity generation, parity error detection
  - Configuration read, configuration write
  - Memory read, memory write, MRM, MRL
  - Interrupt acknowledge
  - I/O read, I/O write
  - Target abort, target retry, target disconnect

LogiCORE Facts				
Resource Utilization(1)	**PCI64 v4**	**PCI32 v4**	**PCI64 v3**	**PCI32 v3**
Slice Four Input LUTs	565	506	724	553
Slice Flip-Flops	404	333	732	566
IOB Flip-Flops	270	150	176	97
IOBs	94	55	89	50
TBUFs (1)			352	288
GCLKs (2)	2	2	1	1
Provided with Core				
Documentation	*PCI v4 Getting Started Guide* *PCI v3 Getting Started Guide* *PCI v4 User Guides* *PCI v3 User Guides*			
Design File Formats	Verilog/VHDL Simulation Model NGO Netlist			
Constraints Files	User Constraints File (UCF) Guide File (NCD)			
Example Design	Verilog/VHDL Example Design			
Design Tool Requirements				
Tested Entry and Verification Tools (3)	Synplicity Synplify Xilinx XST (4) Model Technology ModelSim® Exemplar LeonardoSpectrum Cadence NC-Verilog			
Xilinx Tools	ISE v8.2i			
Support				
Provided by Xilinx @ www.xilinx.com\support				

Notes

1. Resource utilization depends on configuration of the interface and the user design. Unused resources are trimmed by the Xilinx technology mapper. The utilization figures reported in this table are representative of maximum configuration.
2. Designs running at 66 MHz in Virtex-E and Spartan-II devices require one GCLKIOB and two GCLKs. Virtex-5 and Virtex-4 implementations require additional BUFG for 200 MHz reference clock.
3. XST is command line option only. See the *PCI Getting Started Guide* for more information.

© 2006 Xilinx, Inc. All rights reserved. XILINX, the Xilinx logo, and other designated brands included herein are trademarks of Xilinx, Inc. All other trademarks are the property of their respective owners. Xilinx is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Xilinx makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Xilinx expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

Table 1: **Core Implementation**

Supported Device	Core Version	Signaling Environment
PCI64/66		
Virtex-5 VLX50-FF1153-2 (regional clock based)[1,2]	v4	3.3V only
Virtex-4 XC4VFX20-FF672-11C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VLX25-FF668-11C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VSX35-FF668-11C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-II Pro XC2VP50-FF1152-6C/I	v3	3.3V only
Virtex-II Pro XC2VP40-FF1152-6C/I	v3	3.3V only
Virtex-II Pro XC2VP30-FF1152-6C/I	v3	3.3V only
Virtex-II Pro XC2VP20-FF1152-6C/I	v3	3.3V only
Virtex-II Pro XC2VP7-FF672-6C/I	v3	3.3V only
Virtex-II XC2V1000-FG456-5C/I	v3	3.3V only
Virtex-E XCV1000E-FG680-6C	v3	3.3V only
Virtex-E XCV300E-BG432-6C	v3	3.3V only
Virtex XCV1000-FG680-6C	v3	3.3V only
Virtex XCV300-BG432-6C	v3	3.3V only
Spartan-II XC2S150-FG456-6C	v3	3.3V only
Spartan-II XC2S200-FG456-6C	v3	3.3V only
Spartan-IIE XC2S300E-FG456-6C	v3	3.3V only
PCI64/33		
Virtex-5 XC5VLX50-FF1153-1 (global clock based)[2]	v4	3.3V only
Virtex-5 XC5VLX50-FF1153-1 (regional clock based)[2]	v4	3.3V only
Virtex-4 XC4VSX35-FF668-10C/I[1,2] (global clock based)	v3	3.3V only
Virtex-4 XC4VLX25-FF668-10C/I[1,2] (global clock based)	v3	3.3V only
Virtex-4 XC4VFX20-FF672-10C/I[1,2] (global clock based)	v3	3.3V only
Virtex-4 XC4VSX35-FF668-10C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VLX25-FF668-10C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VFX20-FF672-10C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-II Pro XC2VP50-FF1152-5C/I	v3	3.3V only
Virtex-II Pro XC2VP40-FF1152-5C/I	v3	3.3V only

Table 1: **Core Implementation *(Continued)***

Supported Device	Core Version	Signaling Environment
Virtex-II Pro XC2VP30-FF1152-5C/I	v3	3.3V only
Virtex-II Pro XC2VP20-FF1152-5C/I	v3	3.3V only
Virtex-II Pro XC2VP7-FF672-5C/I	v3	3.3V only
Virtex-II XC2V1000-FG456-4C/I/M	v3	3.3V only
Virtex-E XCV1000E-FG680-6C	v3	3.3V only
Virtex-E XCV300E-BG432-6C	v3	3.3V only
Virtex-E XCV100E-BG352-6C	v3	3.3V only
Virtex XCV1000-FG680-5C	v3	3.3V, 5.0V
Virtex XCV300-BG432-5C	v3	3.3V, 5.0V
Spartan-3E XC3S1200E-FG400-4C/I[2]	v3	3.3V only
Spartan-3 XC3S1000-FG456-4C/[2]	v3	3.3V only
Spartan-IIE XC2S300E-FG456-6C	v3	3.3V only
Spartan-IIE XC2S200E-FG456-6C	v3	3.3V only
Spartan-IIE XC2S150E-FG456-6C	v3	3.3V only
Spartan-IIE XC2S100E-FG456-6C	v3	3.3V only
Spartan-II XC2S200-FG456-6C	v3	3.3V, 5.0V
Spartan-II XC2S150-FG456-6C	v3	3.3V, 5.0V
PCI32/66		
Virtex-5 XC5VLX50-FF1153-2 (regional clock based)[1]	v4	3.3V only
Virtex-4 XC4VSX35-FF668-11C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VLX25-FF668-11C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VFX20-FF672-11C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-E XCV400E-FG676-6C	v3	3.3V only
Virtex-E XCV200E-FG256-6C	v3	3.3V only
Virtex XCV200-FG256-6C	v3	3.3V only
PCI32/33		
Virtex-5 XC5VLX50-FF1153-1 (regional clock based)[1]	v4	3.3V only
Virtex-5 XC5VLX50-FF1153-1 (global clock based)[1]	v4	3.3V only
Virtex-4 XC4VSX35-FF668-10C/I[1,2] (global clock based)	v3	3.3V only
Virtex-4 XC4VLX25-FF668-10C/I[1,2] (global clock based)	v3	3.3V only
Virtex-4 XC4VFX20-FF672 -10C/I[1,2] (global clock based)	v3	3.3V only

Table 1: **Core Implementation *(Continued)***

Supported Device	Core Version	Signaling Environment
Virtex-4 XC4VSX35-FF668-10C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VLX25-FF668-10C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-4 XC4VFX20-FF672-10C/I[1,2] (regional clock based)	v3	3.3V only
Virtex-II Pro XC2VP7-FF672-5C/I	v3	3.3V only
Virtex-II XC2V1000-FG456-4C/I/M	v3	3.3V only
Virtex-E XCV1000E-FG680-6C	v3	3.3V only
Virtex-E XCV300E-BG432-6C	v3	3.3V only
Virtex-E XCV100E-BG352-6C	v3	3.3V only
Virtex XCV1000-FG680-5C	v3	3.3V, 5.0V
Virtex XCV300-BG432-5C	v3	3.3V, 5.0V
Spartan-3E XC3S1200E-FG400-4C/I[(7)]	v3	3.3V only
Spartan-3 XC3S1000-FG456-4C/I[(7)]	v3	3.3V only
Spartan-IIE XC2S300E-PQ208-6C	v3	3.3V only
Spartan-IIE XC2S200E-PQ208-6C	v3	3.3V only
Spartan-IIE XC2S150E-PQ208-6C	v3	3.3V only
Spartan-IIE XC2S100E-PQ208-6C	v3	3.3V only
Spartan-IIE XC2S50E-PQ208-6C	v3	3.3V only
Spartan-II XC2S200-PQ208-5C	v3	3.3V, 5.0V
Spartan-II XC2S150-PQ208-5C	v3	3.3V, 5.0V
Spartan-II XC2S100-PQ208-5C	v3	3.3V, 5.0V
Spartan-II XC2S50-PQ208-5C	v3	3.3V, 5.0V

1. Virtex-5, Virtex-4, and Spartan-3E solutions require silicon stepping 1 or later.
2. Requires 200 MHZ reference clock.
3. Virtex and Spartan-II not recommended for CardBus.
4. Commercial devices; 0oC < Tj < 85oC.
5. For additional Part/Package combinations in Virtex-II Pro and older families, see the UCF Generator in the PCI Lounge. For Virtex-4 and newer families, please use UCF Generator in Core Generator.
6. XC2V1000 is supported over Military Temp. range.
7. Spartan-3, Spartan-3E, Virtex-4, and Virtex-5 devices do not contain TBUFs. The Xilinx tools automatically translate TBUFs to LUTs, and they are included in the worst case LUT count listed.
8. Virtex-5, Virtex-4, Virtex-II Pro, Spartan-3, and Spartan-3E devices are supported over commercial and industrial temperatures ranges.

Applications

- Embedded applications in networking, industrial, and telecommunication systems
- PCI add-in boards such as frame buffers, network adapters, and data acquisition boards
- Hot Swap CompactPCI boards
- CardBus compliant
- Any applications requiring a PCI interface

General Description

The PCI interface is a pre-implemented and fully tested module for Xilinx FPGAs. The pinout for each device and the relative placement of the internal logic are pre-defined. Critical paths are controlled by constraints and guide files to ensure predictable timing. This significantly reduces engineering time required to implement the PCI portion of your design. Resources can instead be focused on your unique user application logic in the FPGA and on the system-level design. As a result, Xilinx PCI products minimize your product development time.

The core meets the setup, hold, and clock to timing requirements as specified in the PCI specification, and is verified through extensive simulation.

Other FPGA resources that enable efficient implementation of a PCI system include

- Block SelectRAM™ memory. Blocks of on-chip ultra-fast RAM with synchronous write and dual-port RAM capabilities. Used in PCI designs to implement FIFOs.
- SelectRAM memory. Distributed on-chip ultra-fast RAM with synchronous write option and dual-port RAM capabilities. Used in PCI designs to implement FIFOs.
- Internal three-state bus capability for data multiplexing

The interface is optimized for best possible performance and utilization in Xilinx FPGA devices.

Smart-IP Technology

Drawing on the architectural advantages of Xilinx FPGAs, the Smart-IP technology ensures the highest performance, predictability, reproducibility, and flexibility in PCI designs. The Smart-IP technology is incorporated in every PCI Interface.

The Xilinx Smart-IP technology leverages the Xilinx architectural advantages, such as look-up tables and segmented routing, and floorplanning information, such as logic mapping and location constraints. This technology provides the best physical layout, predictability, and performance. Additionally, these features allow for significantly reduced compile times over competing architectures.

To guarantee the critical setup, hold, minimum clock-to-out, and maximum clock-to-out timing, the PCI interface is delivered with Smart-IP constraints files that are unique for a device and package combination. These constraints files guide the implementation tools so that the critical paths always are within specification. Xilinx provides Smart-IP constraints files for many device and package combinations. Constraints files for unsupported device and package combinations may be generated using the web-based constraints file generator.

Functional Description

Figure 1 illustrates the PCI Interface partitioned into five major blocks and a user application.

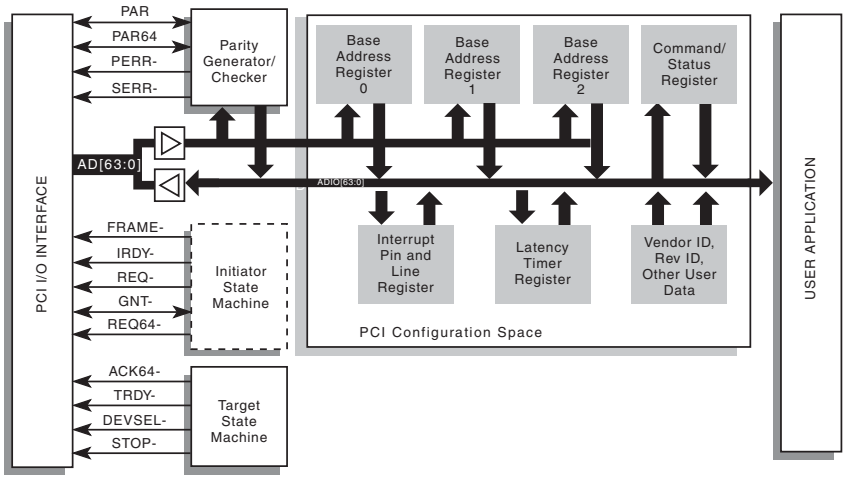


Figure 1: **PCI Interface Block Diagram**

PCI I/O Interface Block

This block handles the physical connection to the PCI bus including all signaling, input and output synchronization, output three-state controls, and all request-grant handshaking for bus mastering.

User Application

The PCI Interface provides a simple, general-purpose interface for a wide range of applications.

PCI Configuration Space

This block provides the first 64 bytes of Type 0, v3.0 Configuration Space Header (Table 2) to support software-driven *Plug-and-Play* initialization and configuration. This includes information for Command, Status, and three Base Address Registers (BARs). The user application interface is capable of extending configuration space. This capability, including the ability to implement a capabilities pointer in configuration space, allows you to implement functions such as power management and message signaled interrupts in the user application.

Table 2: **PCI Configuration Space Header**

31	16	15	0	
Device ID		Vendor ID		00h
Status		Command		04h
Class Code			Rev ID	08h
BIST	Header Type	Latency Timer	*Cache Line Size*	0Ch
Base Address Register 0 (BAR0)				10h
Base Address Register 1 (BAR1)				14h
Base Address Register 2 (BAR2)				18h
Base Address Register 3 (BAR3)				1Ch

Table 2: **PCI Configuration Space Header *(Continued)***

Base Address Register 4 (BAR4)				20h
Base Address Register 5 (BAR5)				24h
Cardbus CIS Pointer				28h
Subsystem ID		Subsystem Vendor ID		2Ch
Expansion ROM Base Address				30h
Reserved			CapPtr	34h
Reserved				38h
Max Lat	Min Gnt	Int Pin	Int Line	3Ch
Reserved				40h-FFh

Note: Shaded areas are not implemented and return zero.

Parity Generator/Checker

This block generates and checks even parity across the AD bus, the CBE# lines, and the parity signals. It also reports data parity errors via PERR# and address parity errors via SERR#.

Initiator State Machine

This block controls the PCI interface initiator functions. The states implemented are a subset of those defined in Appendix B of the *PCI Local Bus Specification*. The initiator control logic uses one-hot encoding for maximum performance.

Target State Machine

This block controls the PCI interface target functions. The states implemented are a subset of those defined in Appendix B of the *PCI Local Bus Specification*. The target control logic uses one-hot encoding for maximum performance.

Interface Configuration

The PCI Interface can be easily configured to fit unique system requirements using the Xilinx CORE Generator GUI or by changing the HDL configuration file. The following customization options, among many others, are supported by the interface and described in the *PCI User Guide.*

- Device and vendor ID
- Base Address Registers (number, size, and type)

Burst Transfer

The PCI bus derives its performance from its ability to support burst transfers. The performance of any PCI application depends largely on the size of the burst transfer. Buffers to support PCI burst transfer can efficiently be implemented using on-chip RAM resources.

Supported PCI Commands

Table 3 illustrates the PCI bus commands supported by the PCI Interface.

Table 3: **PCI Bus Commands**

CBE [3:0]	Command	PCI Initiator	PCI Target
0000	Interrupt Acknowledge	Yes	Yes
0001	Special Cycle	Yes	Ignore
0010	I/O Read	Yes	Yes
0011	I/O Write	Yes	Yes
0100	Reserved	Ignore	Ignore
0101	Reserved	Ignore	Ignore
0110	Memory Read	Yes	Yes
0111	Memory Write	Yes	Yes
1000	Reserved	Ignore	Ignore
1001	Reserved	Ignore	Ignore
1010	Configuration Read	Yes	Yes
1011	Configuration Write	Yes	Yes
1100	Memory Read Multiple	Yes	Yes
1101	Dual Address Cycle	No	Ignore
1110	Memory Read Line	Yes	Yes
1111	Memory Write Invalidate	No	Yes

Bandwidth

The PCI Interface supports fully compliant zero wait-state burst operations for both sourcing and receiving data. This interface supports a sustained bandwidth of up to 528 MBytes/sec. The design can be configured to take advantage of the ability of the PCI Interface to do very long bursts. The flexible user application interface, combined with support for many different PCI features, gives users a solution that lends itself to use in many high-performance applications. The user is not locked into one DMA engine; hence, an optimized design that fits a specific application can be designed.

Recommended Design Experience

The PCI Interface is pre-implemented, allowing engineering focus on the unique user application functions of a PCI design. However, PCI is a high-performance design that is challenging to implement in any technology. For this reason, previous experience building high-performance, pipelined FPGA designs using Xilinx implementation software, constraints files, and guide files is recommended. The challenge to implement a complete PCI design including user application functions varies depending on the configuration and functionality of your application. Contact your local Xilinx representative for a closer review and estimation for your specific requirements.

Timing Specifications

The maximum speed at which the user design is capable of running can be affected by the size and quality of the design. The following tables show the key timing parameters for the PCI Interface.

Table 4 defines the Timing Parameters in the 66 MHz implementations and Table 5 defines the Timing Parameters in the 33 MHz implementations.

Table 4: **Timing Parameters, 66 MHz Implementations**

Symbol	Parameter	Min	Max
T_{cyc}	CLK Cycle Time	15[1]	30
T_{high}	CLK High Time	6	-
T_{low}	CLK Low Time	6	-
T_{val}	CLK to Signal Valid Delay (bussed signals)	2[2]	6[2]
T_{val}	CLK to Signal Valid Delay (point to point signals)	2[2]	6[2]
T_{on}	Float to Active Delay	2[2]	-
T_{off}	Active to Float Delay	-	14[1]
T_{su}	Input Setup Time to CLK (bussed signals)	3[2,3]	-
T_{su}	Input Setup Time to CLK (point to point signals)	5[2,3]	-
T_{h}	Input Hold Time from CLK	0[2,3]	-
T_{rstoff}	Reset Active to Output Float	-	40

Notes

1. Controlled by timespec constraints, included in product.
2. Controlled by SelectIO configured for PCI66_3.
3. Controlled by guide file, included in product.

Table 5: **Timing Parameters, 33 MHz Implementations**

Symbol	Parameter	Min	Max
T_{cyc}	CLK Cycle Time	30[1]	-
T_{high}	CLK High Time	11	-
T_{low}	CLK Low Time	11	-
T_{val}	CLK to Signal Valid Delay (bussed signals)	2[2]	11[2]
T_{val}	CLK to Signal Valid Delay (point to point signals)	2[2]	11[2]
T_{on}	Float to Active Delay	2[2]	-
T_{off}	Active to Float Delay	-	28[1]
T_{su}	Input Setup Time to CLK (bussed signals)	7[2]	-
T_{su}	Input Setup Time to CLK (point to point signals)	10[2]	-
T_{h}	Input Hold Time from CLK	0[2]	-
T_{rstoff}	Reset Active to Output Float	-	40

Notes
1. Controlled by timespec constraints, included in product.
2. Controlled by SelectIO configured for PCI33_3 or PCI33_5.

Ordering Information

The PCI Core is accessed through the Xilinx CORE Generator v8.2i or higher. The Xilinx CORE Generator is bundled with the ISE Foundation v8.2i software at no additional charge. To purchase the Xilinx PCI core, please contact your local Xilinx sales representative.

Part Numbers

- DO-DI-PCI-AL
  - Access to the v3 32/64-bit PCI Lounge IP only core supporting Spartan and Virtex Families
  - Equivalent to purchasing the DO-DI-PCI64-IP and DO-DI-PCI32-IP
- DX-DI-PCI64-IP
  - Upgrade from PCI32 IP only to PCI 64 and PCI32 IP only core

Revision History

Date	Version	Revision
08/30/02	1.2	Style update
12/18/02	1.3	Updated build to v3.0.103; updated Xilinx tool to v5.Ii
3/4/03	1.4	Updated build to v3.0.105; v5.2i; changed document date to 3/7/03; updated date in footer legal content to 2003.
4/14/03	1.5	Updated build to v3.0.106; In LogiCORE Facts table, updated PCI32/66 and PC32/33 product listings to include Spartan-3 device support.

Date	Version	Revision
5/8/03	1.6	Updated Xilinx tools to 5.2i SP2; added Note 10.
8/11/03	1.7	In Supported Products table, PCI64/66 section, Virtex-II Pro XC2VP7FF672-7 was -7c; updated to incorporated Corporate documentation standards.
9/17/03	1.8	Updated build to v3.0.113; in LogiCORE Facts table, Xilinx Tools v6.1i SP1 was v5.2i SP2; date was August 11, 2003.
11/10/03	1.9	Updated build to v3.0.116; in Supported Products table, PCI64/66 section, Virtex-II Pro XC2VP7FF672-6C was -7C.
12/17/03	1.10	Updated device support information in Facts Table (added Virtex-II Pro devices).
1/7/04	1.11	Updated build to v3.0.117; in Supported Products table, sections PCI64/66 and PCI64/33, added Virtex-II Pro XC2VP20...through XC2VP50; updated Xilinx tools to 5.2i SP3.
1/30/04	1.12	Updated build to v3.0.122, changed date to January 25, 2004.
4/9/04	1.13	Updated build to v3.0.126; updated Xilinx tools to 6.2i SP1; added notes 11 and 12 to Supported devices table; added suffix /I to all Virtex-II Pro devices.
4/26/04	1.14	Updated build to v3.0.128, updated Xilinx tools to 6.2i SP2, changed date to April 26, 2004.
7/15/04	1.15	Updated build to v3.0.129, and support for Xilinx software v6.2i SP3. Also, created data sheet in the new template.
11/11/04	1.16	Updated support for Xilinx tools v6.3i; updated PCI spec to v3.0; added Exemplar LeonardoSpectrum and Cadence NC-Verilog entry and verification tools.
12/8/04	1.17	Updated to build 3.0.140 and Virtex-4 support.
3/7/05	1.18	Updated to Xilinx tools 7.1i and PCI build 3.0.145.
5/13/2005	2.0	Updated build to 3.0.150, added support for Spartan-3E, addition of SP2.
8/15/05	3.0	Updated build to 3.0.151, updated SP2 to SP3 for 7.1i
9/12/05	4.0	Updated build to 3.0.152, updated SP3 to SP4 for 7.1i, removed reference to Spartan-3 as pending production speed file. Moved Table 3 to follow text reference.
1/18/06	5.0	Updated build to 3.0.155, ISE Software to v8.1i, release date.
2/14/06	5.5	Updated build to 158, added SP2 support to ISE v8.1i, removed support for Spartan-II XC2S100 device.
7/13/06	6.0	Added v4.0 core, Virtex-4 support, updated build to 160, ISE v8.2i, release date.