Section Titles

Table of Contents

Introduction

LogiCORE Products

AllianceCORE Products

1. Introduction

2. LogiCORE Products

3. AllianceCORE Products

4. LogiBLOX

5. Reference Designs

Introduction

February 8, 1998 Product Specification

Welcome to the Second Edition

We are excited to present the second edition of the CORE Solutions Data Book. We were overwhelmed by the popularity of the first edition, printed just seven months ago.

Since then we have added 43 new products from both Xilinx and our AllianceCORE partners. New product highlights include:

- Version 2.0 of the LogiCORE PCI interface that achieves fully compliant, zero wait-state operation for 132 MBps sustained burst bandwidth.
- LogiCORE and AllianceCORE products that support the new Spartan family of low-cost, ASIC-replacement FPGAs.
- Communications and networking products, including ATM, HDLC and T1 Framer cores.
- Complete RISC Processor solutions, including Xilinx-based prototyping boards and development software.

Because there are so many product descriptions in this book, we have tried to organize things to help you find the product you need faster. The Product Listing by Application Segment section will be the best place to start your search.

Xilinx and our partners are constantly adding to, and improving the products in the CORE Solutions portfolio. These constant changes means that the information in this book will go out of date in a hurry. We strongly recommend that you consult the Xilinx web site for the latest new products and information.

Background

The ASIC core industry has been developing for over a decade. Today there exists a wealth of intellectual property (IP) that is readily available from numerous sources. During this time, however, programmable logic did not have the density or the performance needed to accommodate large IP cores.

Today, things have changed considerably. Xilinx is shipping FPGAs with usable densities up to 500,000 gates. Now, not only is the use of pre-defined logic functions in programmable logic a possibility, it is becoming a requirement to meet ever-shrinking product development cycles.

As a result, many ASIC core vendors and system designers are beginning to look at using cores for their programmable logic designs. It is for this reason that Xilinx created the CORE Solutions portfolio of products.

CORE Solutions Products

CORE Solutions products are broken into five groups, supporting four application areas as shown in Figure 1. The application areas are as follows:

Application Focus	LogiCORE™	XILINX® AllianceCORE	Reference Designs	LogiBLOX	Design Centers
Standard Bus Interface	●	●	●		●
DSP Functions	●	●	●		●
Communications and Networking		●			●
Base-Level Functions	●	●	●	●	●

Figure 1: Solutions Products

- **Standard Bus Interfaces** - such as PCI, PCMCIA, USB and IIC.
- **DSP Functions** - These range from small building blocks such adders, registers and multipliers, to larger system-level functions such as FIR filters and transforms.
- **Communications and Networking** - such as HDLC, T1 framers and ATM functions.
- **Base-Level Functions** - a broad category of functions used across many application segments. These include the very small parameterizable LogiBLOX macros up through larger functions such as UARTs and RISC processors.

A separate section of this databook is dedicated to each CORE Solutions group.

LogiCORE™ Products

Xilinx was the first programmable logic company to embrace the concept of cores for FPGAs with the LogiCORE™ PCI Interface. Today LogiCORE is the most successful core program in the programmable logic industry. Through it, Xilinx has gained considerable experience developing, selling and servicing FPGA cores.

LogiCORE products are pre-implemented and verified system level logic functions sold and supported directly by Xilinx.

The LogiCORE development team is also working on specific core-enabling technologies designed to make cores easier to develop and use. The first of these products includes the CORE Generator tool. Over time, Xilinx will continue to lead the industry with innovative core-based tools.

The LogiCORE Products section of this databook has complete datasheets on all released LogiCORE products.

AllianceCORE™ Program

Xilinx' goal is to expand the availability of quality cores for Xilinx programmable logic. In order to offer the broadest selection of cores to our customers, we have teamed up with leading third-party IP providers through the Xilinx AllianceCORE™ program.

Xilinx works closely with these partners by sharing what we have learned through our own LogiCORE development efforts. AllianceCORE partners work to provide complete solutions for core-based programmable logic design. Individual datasheets for all support tools, such as development software and Xilinx-based prototyping boards are included in the AllianceCORE Products Section. AllianceCORE products are sold and supported directly by our partners.

The AllianceCORE products section of this databook includes complete datasheets on all released AllianceCORE products. Profiles of each partner and their particular areas of expertise are also included.

Reference Designs

Reference Designs are free examples that provide a good starting point for the designer learning how to implement popular building-block and system-level functions.

The Reference Designs section of this databook lists all of the Xilinx Reference Designs that include design files, and where they can be found on the Xilinx world wide web site.

LogiBLOX™ Products

LogiBLOX is a graphical interactive tool for creating simple modules, such as counters, shift registers and multiplexers. LogiBLOX includes a library of generic modules with graphical user interfaces for parameterization, and is shipped with all Xilinx standard development system software.

The LogiBLOX Products section of this databook provides a brief overview of the modules that are available.

Design Centers

The Design Center program was created to provide local technical support from qualified third-party design firms for customers using LogiCORE and AllianceCORE products. Currently, the LogiCORE PCI Interface is the only product supported by certified Xilinx Design Centers. Information for them is included in the LogiCORE PCI datasheets.

The Effect of PLD Architecture on Cores

December 5, 1997

Background

Intellectual property (IP) in the form of pre-defined functions, or cores, has been a hot topic in the electronics industry. The ultimate goal is to be able to select cores off-the-shelf in the form of virtual components from multiple sources and combine them into a system-level design using any EDA tool of preference. The resulting design could then be targeted to any silicon vendors process.

That is the ideal situation and someday we may get there. Unfortunately, there are still tool and silicon dependencies today that effect the performance of a core. This is particularly true for programmable logic. A core that has been tuned for a particular programmable logic device (PLD) architecture will achieve higher performance and better utilization than would a generic, synthesized version of the same core.

Architecture, then, plays a critical role in the implementation of cores in programmable logic. This paper will look at how PLD features impact the performance, consistency, power consumption and efficiency of cores. We will also address some aspects of the impact of software.

PLD Architectures

Most PLDs consist of an array of logic cells that contain a look-up-table and a flip-flop. In Xilinx FPGAs for example, the basic element is known as a Configurable Logic Block, or CLB as shown in Figure 1. The logic cells in PLDs are larger than the basic logic element of an ASIC style gate array, and this difference is the main cause for the less efficient and lower performance results obtained for PLDs through a typical synthesis flow.

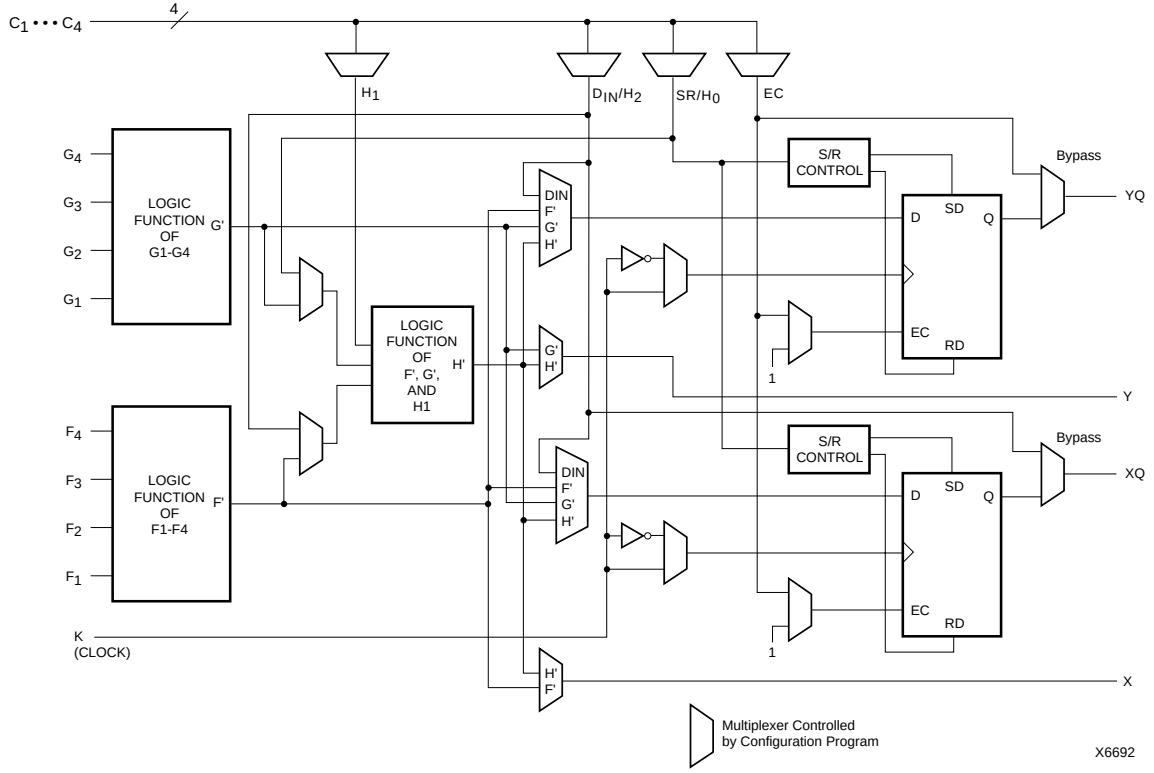


Figure 1: Xilinx XC4000 CLB Structure

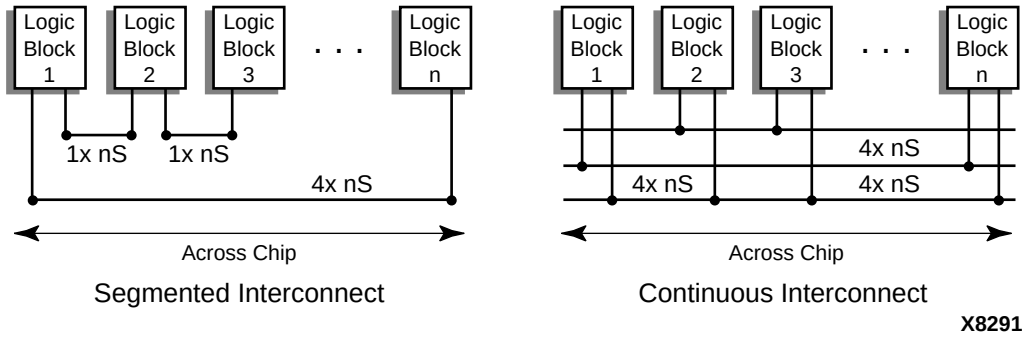


Figure 2: Segmented and Continuous Interconnect

There are differences in each PLD vendor's logic block structure as well as differences in features such as I/O and carry logic, but these involve more detail than will be dealt with here. Instead, we will focus on differences in routing resources and on-chip RAM since these have the most dramatic impact on the use of cores.

Routing Resources

PLDs generally provide routing resources in one of two forms - segmented or continuous (non-segmented) interconnect - as shown in Figure 2.

Xilinx supports segmented interconnect, and also provides global and long-line resources in all of its FPGAs. Segmented interconnect is used for short signals between adjacent CLBs. Long line and global routing resources are used for buses, high fan-out signals, or signals that must travel across the device.

Continuous interconnect architectures rely on long-line resources for signal routing. The touted advantage is to lower the decision-making burden on the place and route tools, reducing overall place and route compilation times. It has, however, a detrimental effect on any design, including those that use cores, as we will see later.

On-Chip RAM

RAM in PLDs appears either distributed across the logic array, or in dedicated, embedded blocks. For example, within each CLB of the Xilinx XC4000 family of FPGAs are two 4-input look-up tables that can be used for logic, or configured as single- or dual-port RAMs in a distributed RAM architecture.

Comparative Analysis

Let's examine, then, the impact that these features have on designs that use cores. We will focus on three important areas: 1) performance consistency, or predictability, 2) efficiency of resource utilization and 3) power consumption.

Performance Consistency

The interconnect scheme of the PLD has considerable impact on how predictable a core's performance will be. Cores designed using relational placement constraints and segmented routing instead of long-line resources can be placed anywhere in the logic array, and will perform exactly as specified regardless of surrounding logic. This is very important when a design uses multiple cores, or cores plus other logic.

If a core is not delivered with a physical layout that leverages segmented routing, then its performance will be impacted by interaction with surrounding logic during the place and route phase of the design flow. This yields an unpredictable slow down in the performance of that core.

Figure 3 shows the impact of these affects using a 12x12 multiplier core as a benchmark. With the segmented routing architecture of the XC4000 family and the physical layout provided with the Xilinx LogiCORE multiplier core, performance remains virtually constant even when multiple instances of the core are included on the same device. In

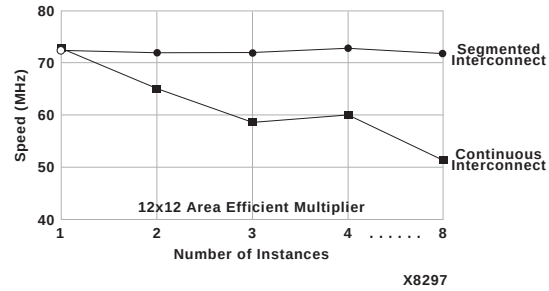


Figure 3: Segmented versus Continuous Interconnect performance for multiple instances of a 12x12 Multiplier Core

contrast, for a PLD with continuous interconnect and cores that only define logic and not layout, note how performance degrades considerably as more cores (or additional user defined logic) are added.

Migrating to a larger device only compounds this problem. Larger devices have longer metal lines with additional capacitance. For a segmented architecture, cores that leverage the interconnect achieve virtually consistent performance regardless of the density of the device. In contrast, cores in a continuous interconnect architecture suffer from a 30 percent performance degradation between the smallest and largest device in the family. This degradation occurs *on top of* that shown in Figure 3.

Performance figures for cores in continuous interconnect architecture PLDs are usually stated for a single instance of that core, implemented in the smallest member of the family in order to show the highest speed. The problem is that it is impossible to know the actual performance of this core until after the design is placed and routed together with all of the other cores and user logic in the target device.

Efficiency

Device utilization is a measure of how efficiently a core uses the logic resources of the PLD. Careful tuning of a design can yield a dramatic reduction in area, resulting in a core that is both very small, and very fast. PLD architecture can have a great impact on this as well.

The availability of abundant routing resources helps a lot. Higher densities can be achieved by offering both global and segmented routing options. Short, isolated signals can be routed onto local interconnect without wasting longer lines. This option is not available with continuous interconnect PLDs.

Many applications, such as in Digital Signal Processing (DSP), take advantage of distributed RAM. PLD-based DSP uses a unique technique known as distributed arithmetic for which a large array of small RAMs is advantageous. PLDs that have no distributed RAM or only block RAM are limited to using flip-flops. In many cases, the sheer number of flip flops required exceeds the number available in the largest device.

For example, a 16-bit shift register that leverages distributed RAM uses only three logic cells in a Xilinx XC4000 device. In a PLD without distributed RAM, the same shift register would take 16 logic cells. This makes functions such as multipliers, transforms and filters take as much as three times the logic resources as they would using distributed RAM. Figure 4 shows comparative examples of this for various 16-bit FIR filters.

Distributed RAM also allows the flexibility to build single- or dual-port memory structures of any size, anywhere in the logic array. Functions like PCI leverage this to obtain 132 Mbyte/sec sustained, zero wait-state burst transfer rates

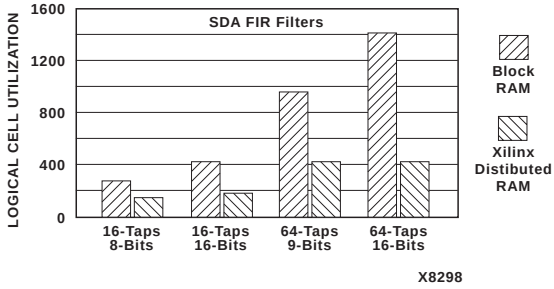


Figure 4: FIR Filter Efficiency in Distributed versus Block RAM PLDs

that are far beyond that which current block-RAM architectures can achieve.

Power Consumption

All device packages have a thermal limit that, when exceeded, will result in thermal breakdown of the device. Since faster clock speeds result in higher power consumption, the thermal limit of the package represents a literal brick wall to the speed at which the chip inside can operate (see Figure 5). Obviously then, a PLD architecture that consumes less power for a given clock frequency can run at a higher speed in the same package.

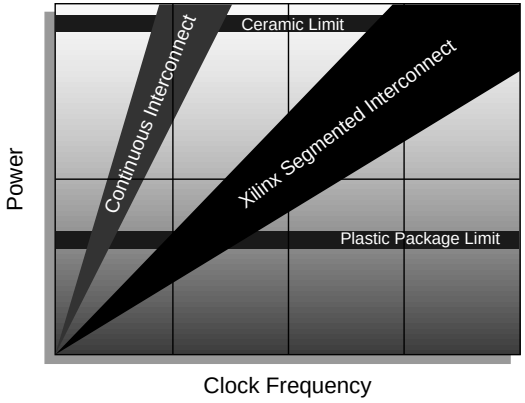


Figure 5: Power Consumption of Segmented versus Continuous Interconnect PLDs

PLDs based on continuous interconnect rely on long metal lines for signal routing. If a signal only needs to travel a short distance on the chip, it must be routed using a longer piece of metal than is necessary, as shown in Figure 2. This introduces additional parasitic capacitance that increases the switching current required. At high frequencies, this causes significantly higher power dissipation. The problem

only gets worse for higher density devices. This is why high density, continuous interconnect PLDs are often only available in large, expensive ceramic packages.

In contrast, a segmented PLD architecture provides the option to route short distance signals on short pieces of metal. This minimizes the load on that signal as well as the current needed to drive it. Cores that are designed using the local interconnect of segmented PLDs will consume less power. This allows the design to run at much higher speeds for a particular package, or to run at the same speed and consume less power than possible using a continuous interconnect architecture. The effect of this is shown in Figure 5. The higher efficiencies achieved through segmented interconnect and distributed RAM further reduce power consumption by allowing a design to be implemented in a smaller device. This allows even higher performance as indicated by the ranges shown in the figure.

Conclusions

The main reason for using cores is to cut development time and allow designers to focus on those portions of the system that they know best. Cores are chosen for features, functionality and performance. When it comes to performance, it is best to know exactly what to expect from a core before the design is started. Anything that increases the possibility of design delays, or even failure, should be avoided.

Xilinx FPGAs, such as the XC4000X architecture, offer capacities reaching the half-million system-gate level. There is no way to maintain the time-to-market value of programmable devices of this capacity efficiently without the use of cores. Xilinx FPGAs offer superior architectural features that allow you to implement cores with guaranteed timing, maximum efficiency and at minimum power allowing higher performance. This comes as the result of silicon features that other PLD architectures cannot alter through upgrades to place and route software.

In addition, all Xilinx LogiCORE functions delivered through the CORE Generator tool are unique in that they include both the logic and layout required to take advantage of the availability of segmented routing, distributed RAM and other architectural features not covered in this paper. This makes them consistent, predictable and fast while minimizing power consumption.

Finally, Xilinx HardWire products provide the fastest available migration to a true high-volume, low-cost ASIC. Conversion to a HardWire device requires no redesign or test vector generation. In this case, the FPGA serves as the prototype and early production vehicle. Once the FPGA design is frozen, Xilinx takes over the conversion process while work can begin on the next design. Only Xilinx owns and offers a risk-free program like this.

No other PLD vendor today offers such a rich set of silicon and software features that makes designing with cores for programmable logic so easy. For more information on Xilinx devices, the CORE Generator tool, or CORE Solutions products, contact your local Xilinx sales representative, send email to logicore@xilinx.com, or look us up on the web at:

http://www.xilinx.com/products/logicore/logicore.htm

XC4000-Series FPGAs: The Best Choice for Delivering Cores

Feb. 8, 1998 XBRF 007 (Version 2.0) Application Brief

Summary

Reusable logic provide an efficient means of embedding common logic functions in high-density FPGA designs. The rich feature set of the XC4000-Series FPGA devices makes them the ideal choice for core-based system design.

Xilinx Family

XC4000E, XC4000EX, XC4000XL, XC4000XV

Introduction

Reusable designs, called cores, have been available to designers using traditional mask-programmed gate arrays for several years. Typically, these cores are pre-designed, pre-tested implementations of widely-used system functions that can be “dropped” into a design and integrated with the application-specific functions of that design. Examples of functions commonly available as cores include bus interfaces, microcontrollers, and DSP functions such as digital filters. The use of such pre-designed modules accelerates time-to- market and allows the designer to focus his or her time and resources on the system’s unique functions.

Now that Field Programmable Gate Arrays (FPGAs) have reached usable densities in the hundreds of thousands of gates, FPGAs have become an ideal “platform” for the development and use of cores and other intellectual property. Specifically, the Xilinx XC4000-Series FPGAs, consisting of the XC4000E, XC4000EX, XC4000XL and XC4000XV family devices, deliver the best combination of performance, density, features, and flexibility to support the use of cores in high-density FPGA design.

Today’s designer can select from a variety of high-density FPGA devices and architectures. However, the successful implementation of system-level designs requires much more than just “raw gates”. The FPGA architecture must include the system-integration features that allow the efficient integration of pre-defined cores and application-specific functions. The XC4000-Series is unique in its incorporation of so many of the features required for the successful implementation and use of cores.

XC4000 Architectural Features

The XC4000-Series silicon provides more that just gates and flip-flops—it provides a system-level solution. The following represent unique XC4000 FPGA features.

Flexible Clocking

Clock structures are important, especially in high-density designs. The XC4000-Series offers up to eight dedicated low-skew, high fan-out clock distribution networks. Any, or all, of the flip-flops can be driven from one of these networks.

However, some applications require more than eight clocks. Each XC4000-Series logic block clock input can be driven by an individual clock signal, allowing literally hundreds of clocks in a single design.

Advanced On-Chip RAM (and ROM)

Many systems require RAM memory, either as standard read/write memory or as FIFOs. XC4000-Series FPGAs provide ample distributed SelectRAM to build small blocks of efficient storage on-chip. On-chip RAM results in better integration and improved system performance.

The XC4000-Series RAM is included in every logic block. Larger memory arrays are built by connecting multiple logic blocks together. Each memory array optimally uses the available device resources. In contrast, FPGAs with only large dedicated RAM blocks often suffer from poor efficiency. Not every design uses all of the RAM available in a large block. Xilinx was first to provide on-chip RAM in a programmable logic device. Since then, other vendors have followed, but do not offer the XC4000’s advanced RAM features.

XC4000-Series RAM supports true dual-port access, providing superior performance in FIFO and data buffering applications. The XC4000 software provides a memory compiler to quickly and easily build large RAM arrays. Plus, the memory compiler can also build ROM structures for implementing control stores, system configuration information, or state machine code.

Internal Bi-Directional Bussing

Most system-level designs contain a bi-directional data bus. Consequently, the XC4000-Series provides an efficient internal bussing structure for optimal bi-directional data flow within the device. In other FPGA technologies, busses must be re-implemented using multiplexers. Buses

implemented using multiplexers consume more resources, resulting in slower system performance.

Numerous and Flexible Output Enables

Large system designs typically require bi-directional I/O. These three-state outputs are controlled by an output enable signal. The XC4000-Series FPGAs provide up to one output-enable per I/O for maximum flexibility. These output-enables are plentiful and operate independently of the clock resources. A Global Three-State forces all outputs to high-impedance for test purposes or for power-on reset functions.

Definable Power-Up State

In many applications, the power-up state of the design is important. Enable lines must be held disabled until the remainder of the system is functioning. Counters must power-up in a known state.

The XC4000-Series FPGAs automatically initialize all of the device flip-flops to a specified state on power-up or after a Global Reset. Each flip-flop can be individually programmed to power-up in the set or reset state.

Furthermore, each of the XC4000 I/O pins can be configured with either a pull-up or pull-down resistor, holding them at a pre-determined logic level until driven.

Built-in JTAG Support

JTAG (IEEE 1148.1) test logic is an important consideration for high-density system designs, especially when using aggressive surface-mount technology like Ball Grid Arrays. The XC4000-Series FPGAs have JTAG support built into every device. No additional logic is required.

Reprogrammable

XC4000 FPGAs are infinitely reprogrammable—even in system. Design modifications can be made quickly and easily, without throwing away devices. A design in the field can even be enhanced with new features or design updates.

XC4000-Series Product Features

Scalable, Broad Product Family

The XC4000-Series is the world's most-popular and the industry's broadest FPGA family. Its scalable architecture and design software provide system solutions ranging from 3,000 gates up to 500,000 gates and beyond, offering the ultimate in design flexibility. An XC4000-based core can be integrated with the user's application and migrated between a wide range of devices, providing the right solution for the design.

Advanced Packaging, Footprint Compatibility

The XC4000-Series devices are available in a wide variety of packaging options, including pin grid array, ball grid array, quad flat pack, and thin quad flat pack. Furthermore, there is footprint compatibility between various XC4000 device densities in the same package plus compatibility with other Xilinx logic technologies. This allows the designer to migrate to higher density devices without having to change the board layout.

Pinout Stability

Often, the design and procurement of the printed circuit board (PCB) is in the critical path for completion of a system design. Thus, designers prefer to fix the pinout of their FPGAs early in the design cycle and maintain that pinout configuration during subsequent design iterations.

The XC4000-Series devices have extra interconnect around the periphery of the device to enhance pinout stability between design revisions. In contrast, other FPGAs ca become "unroutable" as a result of even small design changes if the pinouts are locked in place.

High Quality

XC4000-Series FPGAs are 100%-tested standard products manufactured in high volumes with significantly higher quality than that achievable in most ASIC designs.

Proven Risk-Free Cost-Reduction Path

For high-volume applications, a thoroughly system-verified FPGA design can be cost-reduced using Xilinx HardWire technology. The HardWire conversion process offers risk-free 100% pin- and function-compatibility with the equivalent FPGA device, preserving the same PC board. Plus, thorough test vectors are generated automatically, saving

the engineer from creating complex simulation files. With HardWire technology, the engineer needs to design the XC4000-Series FPGAs: The Best Choice for Delivering Logic Cores, system only once. There is no need to redesign specifically for another technology.

XC4000 Design Software

Successfully implementing a design requires more than just silicon. Design software is an important consideration in any logic design. Xilinx offers complete software development solutions which enable the implementation of designs in Xilinx Programmable Logic devices.

A Wide Variety of EDA Tools

Xilinx offers complete software development solutions which enable the implementation of designs in Xilinx Programmable Logic devices. These development solutions combine powerful technology with a flexible, easy to use graphical interface to help you achieve the best possible designs within your project schedule, regardless of your experience level. Xilinx development solutions are divided into two solution series. The Alliance Series integrates the implementation software with design software from the industry's most extensive set of EDA partners. And our Foundation Series is the Xilinx complete, ready-to-use design solution based on industry-standard HDLs and push-button design flows.

Xilinx M1 Technology

The M1 technology represents Xilinx's next generation baseline technology for implementation and integration functions. This advanced technology enables digital system designers to increase design performance, leverage standards-based, high-level design methodologies, and quickly receive future software updates and device support for Xilinx FPGA and CPLD solutions.

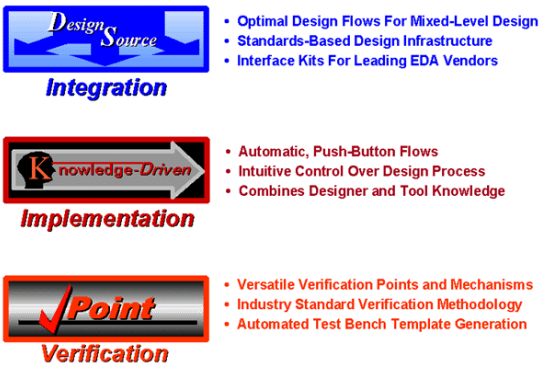
The M1 technology also represents the Xilinx commitment to developing the next generation software platform. It provides the core technology for both our Alliance and Foundation Series, and depicts the next generation performance advancement needed by designers using our ongoing high density CPLD and High Volume efforts. M1 gives you the power to deliver your design on time.

The M1 technology provides dramatically improved design performance through advanced placement and routing algorithms, powerful "auto-interactive" design tools delivering the choice between a push-button or manually-directed design flows, and support for standards-based design methodologies including EDIF, SDF, VHDL (Vital) and Verilog.

Powerful, Simple Flows

The M1 development process insured that we made not only push button flows, but powerful, simple flows. This led us to break the design process into three simple elements. The first being design source integration, which allows you to use mixed mode or multi-level design flows to leverage a standards based design infrastructure. You can then integrate all your design source inputs, including schematic capture, languages, existing netlists, and intellectual property such as LogiCORE and AllianceCORE products.

The second element of the flow is knowledge driven implementation. Here, the development was guided by a simple premise: The belief that understanding the minute details of a given architecture is the responsibility of the tools, not the designer. This frees the designer to concentrate on design structures and performance requirements. We combine



these two strengths to provide intuitive control over the design process, combining the designer's knowledge with a powerful tool.

And finally, check point verification, which delivers versatile verification points and mechanisms based on industry standards. This allows a user to include the verification process into the overall design flow, as an integration, not an after thought. That is why M1 is able to provide automatic test bench template generations from your actual language.

Proven Solution

The hardware and software features discussed above make for a good message. However, theory must be demonstrated in practice before these features have any bearing in reality. Xilinx has proven that the XC4000-Series FPGAs is an ideal delivery vehicle for cores through two programs.

LogiCORE Products

Xilinx has been shipping cores for the XC4000 family for over two years, starting with the highly successful LogiCORE PCI Interface, the industry's highest-performance, fully PCI-compliant interface core for programmable logic. The LogiCORE portfolio of products has since expanded to include many small to complex cores for common and digital signal processing applications. All of the LogiCORE products are designed to fully exploit the XC4000 silicon and software features to achieve predictable performance while using minimal logic resources.

AllianceCORE Program

The AllianceCORE program forms partnerships between Xilinx and third-party intellectual property providers who are also producing cores that exploit the unique features of the Xilinx XC4000 solution. This program helps leverage the system-level background of many industry experts to broaden the range of cores that are available for Xilinx programmable logic.

Support

World-Class Technical Support

A working design takes more than silicon and software. It sometime takes support, too. Xilinx is committed to providing world-class technical support through its world-wide network of expert field applications engineers, telephone hotline support engineers, World-Wide Web and E-mail links, design center, and training classes.

Standard Bus Interfaces

Peripheral Component Interconnect Bus

Function	CORE Solution	Source	Page
PCI Master & Slave Interfaces 2.0	LogiCORE	Xilinx	2-11
PCI Master & Slave Interfaces 1.2.0	LogiCORE	Xilinx	2-19
PCI32 Spartan Master & Slave Interfaces	LogiCORE	Xilinx	2-29
Synthesizable PCI Bridge Design Example	LogiCORE	Xilinx	2-35

PC-Card Bus (PCMCIA)

Function	CORE Solution	Source	Page
PCMCIA Fax/Modem Macro	AllianceCORE	Mobile Media Research	3-15
PCMCIA Library R 1.2	AllianceCORE	Mobile Media Research	3-19
PCMCIA Prototyping Card	AllianceCORE	Mobile Media Research	3-23
PCMCIA Card Debugger/Exerciser	AllianceCORE	Mobile Media Research	3-25
PCMCIA CIS Generator 1.2	AllianceCORE	Mobile Media Research	3-27

Universal Serial Bus

Function	CORE Solution	Source	Page
Low-Speed USB Function Controller	AllianceCORE	Mentor	3-29
Full-Speed USB Function Controller	AllianceCORE	Mentor	3-33
3-Port USB Hub Controller	AllianceCORE	Mentor	3-37
USB Function Evaluation Board	AllianceCORE	Mentor	3-41
USB Hub Evaluation Board	AllianceCORE	Mentor	3-43
USB Simulation Model	AllianceCORE	Mentor	3-45

Other Standard Bus Products

Function	CORE Solution	Source	Page
ISA Plug and Play Interface	Reference Design	Xilinx	5-3
XF-TWSI Two-Wire Serial Interface (IIC)	AllianceCORE	Memec Design Services	3-9

Digital Signal Processing

Correlators

Function	CORE Solution	Source	Page
One Dimensional RAM-Based Correlator	LogiCORE	Xilinx	2-45
One Dimensional ROM-Based Correlator	LogiCORE	Xilinx	2-41

Filters

Function	CORE Solution	Source	Page
Comb Filter	LogiCORE	Xilinx	2-49
16-Tap, 8-Bit FIR Filter	Reference Design	Xilinx	5-3
Serial Distributed Arithmetic FIR Filter	LogiCORE	Xilinx	2-51
Dual-Channel Serial Distributed Arithmetic FIR Filter	LogiCORE	Xilinx	2-55
Parallel Distributed Arithmetic FIR Filter	LogiCORE	Xilinx	2-59

Transforms

Function	CORE Solution	Source	Page
DFT Core (Real Data In, Complex Data Out)	LogiCORE	Xilinx	2-65
FFT Core (1024 Points)	LogiCORE	Xilinx	2-69

DSP Building Blocks

Function	CORE Solution	Source	Page
SDA FIR Control Logic	LogiCORE	Xilinx	2-73
Sine/Cosine	LogiCORE	Xilinx	2-75
Non-Symmetric, 16-Deep Time Skew Buffer	LogiCORE	Xilinx	2-77
Non-Symmetric, 32-Deep Time Skew Buffer	LogiCORE	Xilinx	2-81
Symmetric, 16-Deep Time Skew Buffer	LogiCORE	Xilinx	2-85

Communications and Networking

Asynchronous Transfer Mode

Function	CORE Solution	Source	Page
Cell Assembler (CC-201)	AllianceCORE	CoreEl Microsystems	3-49
Cell Delineation (CC-200)	AllianceCORE	CoreEl Microsystems	3-53
CRC10 Generator and Verifier (CC-130)	AllianceCORE	CoreEl Microsystems	3-57
CRC32 Generator and Verifier (CC-131)	AllianceCORE	CoreEl Microsystems	3-61
UTOPIA Slave (CC-141)	AllianceCORE	CoreEl Microsystems	3-85

Forward Error Correction

Function	CORE Solution	Source	Page
Reed-Solomon Decoder	AllianceCORE	Integrated Silicon Systems	3-71
Reed-Solomon Encoder	AllianceCORE	Integrated Silicon Systems	3-77
Viterbi Decoder	AllianceCORE	CAST	3-91

Telecommunications

Function	CORE Solution	Source	Page
HDLC Protocol Core	AllianceCORE	Integrated Silicon Systems	3-65
MT1F T1 Framer	AllianceCORE	Virtual IP Group	3-81

Base-Level Functions

Basic Elements

Function	CORE Solution	Source	Page
Clock Divider	LogiBLOX	Xilinx	4-4
Comparator	LogiBLOX	Xilinx	4-4
Constant	LogiCORE	Xilinx	2-91
Constant	LogiBLOX	Xilinx	4-4
Counter	LogiBLOX	Xilinx	4-4
Loadable Binary Counter	Reference Design	Xilinx	5-4
Ultra-Fast Synchronous Counter	Reference Design	Xilinx	5-4
Accelerating Loadable Counters	Reference Design	Xilinx	5-4
Data Register	LogiBLOX	Xilinx	4-4
Decoder	LogiBLOX	Xilinx	4-4
Frequency/Phase Comparator for PLL	Reference Design	Xilinx	5-5
Simple Gates	LogiBLOX	Xilinx	4-4
Harmonic Frequency Synthesizer and FSK Modulator	Reference Design	Xilinx	5-5
Input/Output	LogiBLOX	Xilinx	4-4
Multiplexer	LogiBLOX	Xilinx	4-4
Multiplexers, Barrel Shifters	Reference Design	Xilinx	5-3
Two Input MUX	LogiCORE	Xilinx	2-93
Three Input MUX	LogiCORE	Xilinx	2-95
Four Input MUX	LogiCORE	Xilinx	2-97
Parallel to Serial Converter	LogiCORE	Xilinx	2-99
Pulse-Width Modulation	Reference Design	Xilinx	5-5

Basic Elements

Function	CORE Solution	Source	Page
Register	LogiCORE	Xilinx	2-101
Serial Code Conversion between BCD and Binary	Reference Design	Xilinx	5-5
Shift Register	LogiBLOX	Xilinx	4-4
Tristate	LogiBLOX	Xilinx	4-4

Math Functions

Function	CORE Solution	Source	Page
1's and 2's Complement	LogiCORE	Xilinx	2-103
Accumulator	LogiBLOX	Xilinx	4-4
Scaled by 1/2 Accumulator	LogiCORE	Xilinx	2-105
Adder/Subtracter	LogiBLOX	Xilinx	4-4
Adders, Subtractors, Accumulators	Reference Design	Xilinx	5-3
Registered Adder	LogiCORE	Xilinx	2-107
Registered Loadable Adder	LogiCORE	Xilinx	2-109
Registered Scaled Adder	LogiCORE	Xilinx	2-111
Registered Serial Adder	LogiCORE	Xilinx	2-113
Integrator	LogiCORE	Xilinx	2-115
Constant Coefficient Multiplier	LogiCORE	Xilinx	2-117
Constant Coefficient Multiplier (Pipelined)	LogiCORE	Xilinx	2-119
Parallel Multipliers, Area Optimized	LogiCORE	Xilinx	2-121
Parallel Multipliers, Performance Optimized	LogiCORE	Xilinx	2-125
Square Root	LogiCORE	Xilinx	2-127
Registered Subtracter	LogiCORE	Xilinx	2-129
Registered Loadable Subtracter	LogiCORE	Xilinx	2-131

Memories

Function	CORE Solution	Source	Page
Delay Element	LogiCORE	Xilinx	2-133
FIFOs in XC4000 RAM	Reference Design	Xilinx	5-4
Register-Based FIFO	Reference Design	Xilinx	5-4
Synchronous FIFO	LogiCORE	Xilinx	2-135
16-Word Deep Registered Look-Up Table	LogiCORE	Xilinx	2-139
32-Word Deep Registered Look-Up Table	LogiCORE	Xilinx	2-141
Registered Synchronous RAM	LogiCORE	Xilinx	2-143
Registered ROM	LogiCORE	Xilinx	2-145
ROM, RAM, Synch-RAM, Dual Port RAM	LogiBLOX	Xilinx	4-4
4Mb Virtual SPROM	Reference Design	Xilinx	5-5

Processor Products

Function	CORE Solution	Source	Page
Dynamic Microcontroller	Reference Design	Xilinx	5-4
TX400 Series RISC CPU Cores	AllianceCORE	T7L Technology Inc.	3-97
RISC CPU Core Design Base Board	AllianceCORE	T7L Technology Inc.	3-103
Scalable Development Platform Integrated Software	AllianceCORE	T7L Technology Inc.	3-107
V8 uRISC 8-bit RISC Microprocessor	AllianceCORE	VAutomation	3-109
IntelliCore™ Prototyping System	AllianceCORE	VAutomation	3-115

Processor Peripherals

Function	CORE Solution	Source	Page
C2910A Microprogram Controller	AllianceCORE	CAST	3-129
Configuring FPGAs over a Processor Bus	Reference Design	Xilinx	5-5
M8237 DMA Controller	AllianceCORE	Virtual IP Group	3-133
M8254 Programmable Timer	AllianceCORE	Virtual IP Group	3-143
M8255 Programmable Peripheral Interface	AllianceCORE	Virtual IP Group	3-147
XF8255 Programmable Peripheral Interface	AllianceCORE	Memec Design Services	3-151
XF8256 Multifunction Microprocessor Support Controller	AllianceCORE	Memec Design Services	3-155
M8259 Programmable Interrupt Controller	AllianceCORE	Virtual IP Group	3-159
XF8279 Programmable Keyboard Display Interface	AllianceCORE	Memec Design Services	3-163
XF9128 Video Terminal Logic Controller	AllianceCORE	Memec Design Services	3-167
DRAM Controller	AllianceCORE	NMI Electronics	3-173

UARTs

Function	CORE Solution	Source	Page
XF-8250 Asynchronous Communications Element	AllianceCORE	Memec Design Services	3-137
M16450 - Universal Asynchronous Receiver/ Transmitter	AllianceCORE	Virtual IP Group	3-121
M16550A - UART With RAM	AllianceCORE	Virtual IP Group	3-125

Generic Development Tools

Function	CORE Solution	Source	Page
GVA-100 DSP Prototyping Platform	AllianceCORE	GV & Associates	3-179
MDS FPGA Development Module	AllianceCORE	Memec Design Services	3-183
Microprocessor-Based Core Evaluation Card	AllianceCORE	NMI Electronics	3-185
Xilinx CORE Generator	LogiCORE	Xilinx	2-7

LogiCORE Products

LogiCORE Products

LogiCORE products are sold, licensed and supported by Xilinx. They are developed internally by Xilinx or jointly with a partner.

The cores that Xilinx provides as LogiCORE products typically fall into one of two categories. The first are high-performance interface cores that require a thorough understanding and control of the FPGA technology and implementation software in order to achieve the desired performance and complexity. An example of a core in this category is the LogiCORE PCI interface.

The second category are cores that benefit from a very specialized implementation in the FPGA. An example is the LogiCORE DSP modules that are implemented using a unique algorithm, Distributed Arithmetic. This algorithm fits the lookup-table-based architecture of the FPGA. The result is outstanding performance and device utilization, often more than 10 times better than generic HDL descriptions.

Xilinx CORE Generator

In addition to actual cores, Xilinx is committed to develop enabling design tools and methodologies to facilitate usage of cores with FPGAs. The first of this product type is the CORE Generator tool. This innovative methodology for acquiring and using cores allow to combine the benefits of 1) a firm core with predictable performance, and 2) the flexibility of system level design, facilitated by behavioral description languages such as VHDL and Verilog.

LogiCORE products are customized to fit your specific application using an intuitive graphical user interface. Based on your inputs, the CORE Generator generates a proven core with predictable timing that can be integrated using a VHDL-, Verilog- or schematic-based design flow. As a result, you can put several, individually proven cores with given performance, into one system on a single FPGA. Because each core is already verified, the time-to-market benefits are maintained for high-complexity FPGAs.

Xilinx PCI solutions

The Xilinx PCI solution includes the necessary devices, tools and cores to build a cost-effective single-chip PCI system in record time.

- LogiCORE PCI - the highest performing PCI core with predictable timing
- XC4000E/XLT - the industry's fastest FPGAs that allow you to integrate a PCI interface plus 5 to 60 thousand gates
- Spartan - the industry's lowest cost FPGA; on-chip RAM allows high performance for the most cost-effective programmable solution for PCI.
- HardWire - an automatic migration path to a low-cost chip for volume production
- Web-based configuration - for easy configuration and integration of the LogiCORE PCI module
- 3rd party Design Centers - with PCI expertise available for special applications and customization of the core

PCI is an extremely high-performance and complex specification that is challenging to meet in any technology. To meet the stringent PCI specification the core is carefully hand-tuned for the targeted architecture. Placement and routing for the critical parts of the core are locked down to ensure that timing can be met every time the core are implemented.

To achieve our goals, the LogiCORE development team is working closely with both IC and Software teams. New methodologies for characterizing and modeling our FPGAs have been developed as a result of this teamwork. You will have access to state of the art technology and expertise, allowing you to complete your PCI application in record time.

Xilinx has sold hundreds of LogiCORE PCI licenses and has built up a solid knowledge about PCI. We are committed, and will continuously improve our PCI products to remain state of the art.

Xilinx DSP Solutions

Using an FPGA to implement DSP functions often allow a radical performance advantage over fixed processors while maintaining maximum flexibility and shortest time-to-market. Until now, tools to automate the design process have been lacking and most designs have been completed manually by experienced FPGA designers.

Xilinx' DSP LogiCORE products are complex parameterized DSP building blocks that deliver performance and density equal to or better then a hand-tuned implementation. LogiCORE DSP modules can be used with VHDL, Verilog or schematic capture design methodologies.

Higher level DSP cores are available from our AllianceCORE partners.

Acquiring LogiCORE Products

LogiCORE products are available from your local Xilinx sales representative similar to other Xilinx software products. Xilinx and your local sales representative will also be your primary source for support of the core, the devices and the design tools. You can also send email questions to: logicore@xilinx.com.

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

The Xilinx CORE Generator is an FPGA productivity enhancement tool that generates parameterized building blocks (cores) compatible with standard Xilinx FPGA design methodologies.

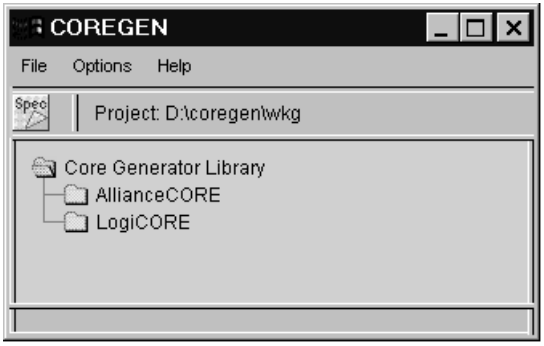


Figure 1: The CORE Generator Window

Features

- Compatible with VHDL, Verilog, and schematic capture top-level design methodologies.
- Supports PC and Workstation platforms and permits web-based operation.
- Supports Xilinx LogiCOREs and third-party AllianceCOREs.
- Easy to use, intuitive point and click operation.
- Plug & Play-COREs can be added at any time by downloading new cores from the Xilinx web page.
- Supplies the link from system level design tools to Xilinx FPGA silicon.
- Compatible with Xilinx Foundation, Alliance, and XACT development systems.

CORE Generator with Xilinx LogiCOREs

- Automatically generates parameterized cores and HDL behavioral models.
  - Wide range of core complexity from basic logic building block to system level functions.
  - Supports the XC4000E, EX, XL, XV, and Spartan FPGA families.
- Unique methodology produces a logic design and a layout (floorplan) for each core.
- Generates optimal designs (best performance and density) for FPGAs.
- Performance and size are known and specified before the core is generated.
- Consistent performance independent of Xilinx FPGA device size and independent of percent utilization.
- Reduces overall FPGA implementation time; cores come mapped and relatively placed.

Introduction

The Xilinx CORE Generator generates parameterized cores optimized for Xilinx FPGAs. It serves as a cataloging and delivery vehicle for Xilinx designed and supported LogiCOREs, third-party designed and supported AllianceCOREs, data sheets, and application notes.

The CORE Generator fills the need created by the introduction of the next generation high-capacity FPGAs that use advanced semiconductor process technology. A 100,000-gate design can be built from a collection of cores in a matter of hours.

FPGA LogiCORE: Matrix of CLBs

Xilinx FPGAs consist of an expandable matrix of “Configurable Logic Blocks” (CLBs). A LogiCORE is delivered as a pre-defined group of CLBs that maintain their relative locations when placed anywhere in an FPGA.

Unique Design Methodology: CORE Generator and LogiCOREs

The output of the CORE Generator is a netlist that specifies the logic design and a physical layout or floorplan. This logic and layout methodology, combined with the segmented routing architecture of the Xilinx XC4000 family, generates designs with guaranteed performance. The size (number of CLBs) and the performance (maximum operating frequency) are specified in the data sheet for each core.

When installed in an FPGA, the cores meet the same performance specifications independent of device size and

independent of how many cores are used in a large FPGA device.

Traditionally, cores for most other FPGAs are “soft” cores because the physical layout and characteristics of the design cannot be known prior to placement and routing. Every placement and routing iteration results in a different physical layout and as a result, the performance is not predictable.

The CORE Generator takes advantage of the Xilinx segmented architecture and does not use any of the global routing resources except for the global clock distribution network. The core can be placed anywhere in any size FPGA and then interconnected with standard routing software. The performance will always be the same for a given speed device.

Without this predictability it is not possible to know whether a design will work until the final phases of a project, when it may be too late to move to an alternative approach. With the CORE Generator approach, the theoretical design matches the actual design.

Designs optimized for FPGAs

The CORE Generator produces designs that are optimized for FPGAs in two ways. First, the LogiCORE implementations are generated with FPGA architecture in mind and take advantage of the look-up table logic and distributed RAM. Second, they include mapping information and relative location constraints for each CLB so that the layout (floorplan) is always optimal.

The efficiency (number of CLBs) and performance of each parameterized LogiCORE is comparable to a hand-packed design.

The FPGA device power dissipation is also improved through the use of cores. Designs with optimal layout use less programmable interconnect and thus less dynamic power.

Compatible with standard tools

The CORE Generator can be used with any HDL top-level design methodologies or schematic capture design tools (Xilinx Foundation or Viewlogic).

For HDL environments the CORE Generator produces VHDL or Verilog code that can be spliced into the top-level HDL design. This instantiation code directs the synthesis tool to the associated core netlist during the HDL synthesis process and the Xilinx place and route tools combine all of the netlists into a single design. The CORE Generator also delivers a behavioral model for simulation.

For schematic capture flows a symbol is automatically generated. The user-parameterized core can then be included in the overall schematic along with traditional schematic elements.

CORE Generator Design Flow

When the Xilinx CORE Generator is initiated, a COREGEN window appears with expandable folders that contain lists of available cores and lower level functional building blocks. The hierarchical list can be expanded until the desired core is located and its parameterization screen appears. Enter the parameters and click on the generate button to initiate the core generation process.

After the parameterized core is generated, the schematic symbol is placed in the schematic capture library or the HDL instantiation code is placed in a file.

The parameterized core can be used with either Foundation or Alliance series flows.

LogiCORE™

PCI Master & Slave Interfaces Version 2.0

December 15, 1997 Data Sheet

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Introduction

With Xilinx LogiCORE PCI Master and Slave interface Version 2.0, a designer can build a customized, 32 bit, 33 MHz fully PCI compliant system with the highest possible performance, 132 Mbytes/s, and up to 124,000 system gates in a XC4000XLT FPGA.

Features

- Fully 2.1 PCI compliant 32 bit, 33 MHz PCI Interface
  - Master (Initiator/Target)
  - Slave (Target-only)
- Programmable single-chip solution with customizable back-end functionality
- Pre-defined implementation for predictable timing in Xilinx XC4000XLT FPGAs or HardWire™ FpgASICs (see *LogiCORE Facts* for listing of supported devices)
- 3.3 V Operation with XC4000XLT devices
- Zero wait state operation
- Fully verified design
  - Tested with the Xilinx internal testbench
  - Simulated using VirtualChips™ PCI testbench
  - Tested in hardware (silicon proven)
- Configurable on-chip FIFO can be added for maximum burst speed (see *Xilinx Documents* section)
- Design Once™ - automatic conversion to HardWire for cost reduction
- Supported Initiator functions (PCI Master only)
  - Initiate Memory Read, Memory Write, Memory Read Multiple (MRM), Memory Read Line (MRL) commands
  - Initiate I/O Read, I/O Write commands
  - Initiate Configuration Read, Configuration Write commands
  - Bus Parking
  - Basic Host Bridging

LogiCORE™ Facts

Core Specifics		
Device Family	XC4000XLT	
CLBs Used	178 - 308	
IOBs Used[1]	53/51	
System Clock f_{max}	0 - 33MHz	
Device Features Used	Bi-directional data buses SelectRAM™ (optional user FIFO) Boundary scan (optional)	
Supported Devices/Resources Remaining		
	I/O[1,2]	**CLB**[3]
XC4013XLT PQ208	99/101	268 - 398
XC4013XLT HQ240	133/135	268 - 398
XC4028XLT HQ240	133/135	716 - 846
XC4062XLT HQ240	133/135	1996 - 2126
XC4062XLT BG432	293/295	1996 - 2126
Provided with Core		
Documentation	*User's Guide v2.0* *PCI Data Book v2.0*	
Design File Formats	VIEW*logic* schematics EDIF Netlist[4]	
Constraint Files	M1 User Constraint File (UCF) M1 Guide files	
Verification Tools	VIEW*logic* command files VHDL Testbench Verilog Testbench	
Schematic Symbols	VIEW*logic*, VHDL, Verilog	
Evaluation Model	VHDL, Verilog Simulation Model[4]	
Reference designs & application notes	Example design: Ping Reference Design[5]	
Additional Items	Reference book: *PCI System Architecture*	
Design Tool Requirements		
Xilinx Core Tools	M1.3.7	
Entry/Verification Tools	For CORE generator: VHDL, Verilog, Schematic For changing source files: Workview Office V7.1.2 or V7.2	

Notes: See next page.

LogiCORE™ Facts (cont.)

Support

Xilinx provides technical support for this LogiCORE™ product when used as described in the User's Guide or supporting Application Notes. Xilinx cannot guarantee timing, functionality, or support of the product if implemented in devices not listed above, or customized beyond that referenced in the product documentation, or if any changes are done in sections of the design marked as "DO NOT MODIFY".

Notes:
1. Master/Slave
2. The XLT devices use 8 I/O locations for Vtt pins
3. The exact number of CLBs depends on user configuration of the core and level of resource sharing with adjacent logic. Factors that can affect the size of the design are number and size of the BARs, zero vs. one wait state, and medium vs. slow decode. These numbers include a 16 x 32 FIFO.
4. Available on Xilinx Home Page, in the LogiCORE PCI VIP Lounge: www.xilinx.com/products/logicore/logicore.htm
5. Slave only

Features (cont.)

- Supported Target functions (PCI Master and Slave)
  - Type 0 Configuration Space Header
  - Up to 3 Base Address Registers (memory or I/O with adjustable block size from 16 bytes to 2 Gbytes, slow or medium decode speed)
  - Parity Generation (PAR), Parity Error Detection (PERR# and SERR#)
  - Memory Read, Memory Write, Memory Read Multiple (MRM), Memory Real Line (MRL), Memory Write, Invalidate (MWI) commands
  - I/O Read, I/O Write commands
  - Configuration Read, Configuration Write commands
  - 32-bit data transfers, burst transfers with linear address ordering
  - Target Abort, Target Retry, Target Disconnect
  - Full Command/Status Register
- Supported by Xilinx CORE Generator
  - Web-based configuration
  - Generation of proven design files

Applications

- PCI add-in boards such as graphic cards, video adapters, LAN adapters and data acquisition boards
- Embedded applications within telecommunication and industrial systems
- CompactPCI boards
- Other applications that need PCI

General Description

The LogiCORE™ Master and Slave Interfaces v2.0 are pre-implemented and fully tested modules for Xilinx XC4000XLT FPGAs (see *LogiCORE Facts* for listing of supported devices). The pin-out and the relative placement of the internal Configurable Logic Blocks (CLBs) are predefined. Critical paths are controlled by TimeSpec's to ensure that timing is always met. This significantly reduces engineering time required to implement the PCI portion of your design. Resources can instead be focused on the

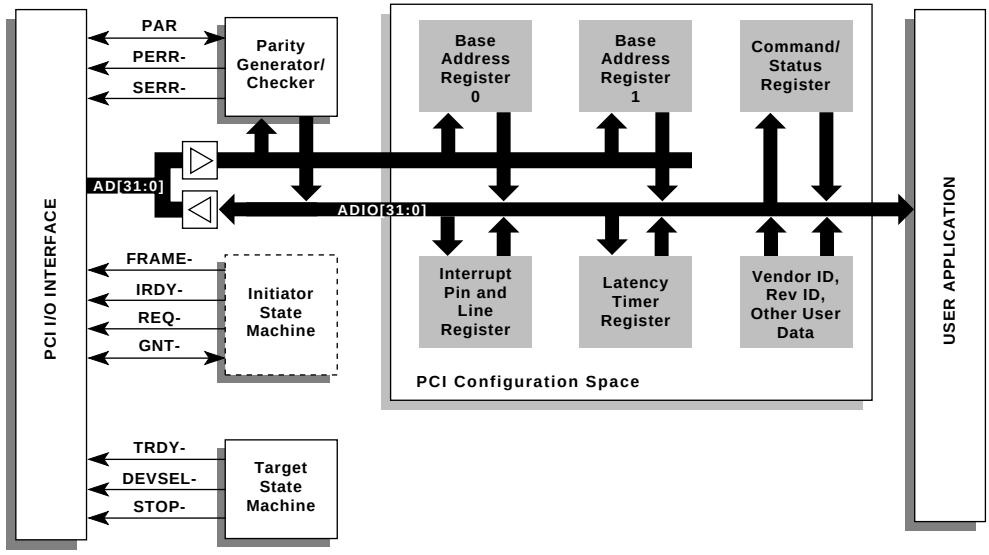


Figure 1: LogiCORE™ PCI Interface Block Diagram (BAR 2 not shown)

unique back-end logic in the FPGA and the system level design. As a result, the LogiCORE™ PCI products can cut your development time by several months.

Xilinx XC4000XLT Series FPGAs enables designs of fully PCI-compliant systems. The devices meet all required electrical and timing parameters including AC output drive characteristics, input capacitance specifications (10pF), 7 ns setup and 0 ns hold to system clock, and 11 ns system clock to output. These devices meet all specifications for 3.3 V PCI. Although the XLT devices have a 3V driver they can be used in a 5V PCI system and meet timing for up to 8 loads.

The XC4000XLT devices differ from regular XL devices by the addition of clamp diodes, required by the PCI 3.3 V electrical specification. For more details about this see the Application Note, *Using the XC4000XL for 3.3 V and 5 V PCI Applications*.

The PCI Compliance Checklist (In the *PCI Data Book*) has additional details about electrical compliance. Other features that enable efficient implementation of a complete PCI system in the XC4000XLT includes:

- Select-RAM™ memory: on-chip ultra-fast RAM with synchronous write option and dual-port RAM option. Used in the PCI Interfaces to implement the FIFO.
- Individual output enable for each I/O
- Internal 3-state bus capability
- 8 global low-skew clock or signal distribution networks
- IEEE 1149.1-compatible boundary scan logic support

See the latest Xilinx Programmable Logic Data Book for more details.

The module is carefully optimized for best possible performance and utilization in the XC4000XLT FPGA architecture. When implemented in the XC4013, more than 50% of the FPGA's resources remain for integrating a unique back-end interface and other system functions into a fully programmable one-chip solution. When implemented in a XC4062, 90% of the FPGA's resources remain.

Xilinx DesignOnce™ service allows an automatic conversion to a low cost HardWire™ device for high-volume production.

Functional Description

The LogiCORE PCI Master Interface is partitioned into five major blocks, plus the user application, shown in Figure 1. Each block is described below.

PCI I/O Interface Block

The I/O interface 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.

Parity Generator/Checker

Generates/checks even parity across the AD bus, the CBE lines, and the PAR signal. Reports data parity errors via PERR- and address parity errors via SERR-.

Target State Machine

This block manages control over the PCI interface for Target functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The controller is a high-performance state machine using state-per-bit (one-hot) encoding for maximum performance. State-per-bit encoding has narrower and shallower next-state logic functions that closely match the Xilinx FPGA architecture.

Initiator State Machine (PCI Master only)

This block manages control over the PCI interface for Initiator functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The Initiator Control Logic also uses state-per-bit encoding for maximum performance.

PCI Configuration Space

This block provides the first 64 bytes of Type 0, version 2.1, Configuration Space Header (CSH) (see Table 1) to support software-driven "Plug-and Play" initialization and configuration. This includes Command, Status, and three Base Address Registers (BARs). BAR 2 is not shown in figure 1. These BARs illustrate how to implement memory- or I/O-mapped address spaces. Each BAR sets the base address for the interface and allows the system software to determine the addressable range required by the interface. Using a combination of Configurable Logic Block (CLB) flip-flops for the read/write registers and CLB look-up tables for the read-only registers results in optimized packing density and layout.

With this release, the hooks for extending configuration space has been built into the backend interface. These hooks, including the ability to implement a CapPtr in configuration space, allows the user to implement functions such as Advanced Configuration and Power Interface (ACPI) in the backend design.

Table 1: 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
Base Address Register 4 (BAR5)				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	Interrupt Pin	Interrupt Line	3Ch
Reserved				40h-FFh

Note:
Italicized address areas are not implemented in the LogiCORE PCI Interface default configuration. These locations return zero during configuration read accesses.

User Application with Optional Burst FIFOs

The LogiCORE PCI Interface provides a simple, general-purpose interface with a 32-bit data path and latched address for de-multiplexing the PCI address/data bus. The general-purpose user interface allows the rest of the device to be used in a wide range of applications.

Typically, the user application contains burst FIFOs to increase PCI system performance (An Application Note is available, please see the *Xilinx Documents* section). An on-chip read/write FIFO, built from the on-chip synchronous dual-port RAM (SelectRAM™) available in XC4000XLT devices, supports data transfers in excess of 33 MHz.

Core Configuration

The LogiCORE PCI Interface can easily be configured to fit unique system requirements using Xilinx web-based graphical configuration tool or changing the VHDL, Verilog, or VIEW*logic* configuration file. The following customization is supported by the LogiCORE product and described in accompanying documentation.

- Initiator or target functionality (PCI Master only)
- Base Address Register configuration (1 - 3 Registers, size and mode)
- Configuration Space Header ROM
- Initiator and target state machine (e.g., termination conditions, transaction types and request/transaction arbitration)
- Burst functionality
- User Application including FIFO (back-end design)

Table 2: PCI Bus Commands

CBE [3:0]	Command	PCI Master	PCI Slave
0000	Interrupt Acknowledge	No[1]	Ignore
0001	Special Cycle	No[1]	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[1]	Ignore
1110	Memory Read Line	Yes	Yes
1111	Memory Write and Invalidate	No[1]	Yes

Note:
1. The Initiator can present these commands, however, they either require additional user-application logic to support them or have not been thoroughly tested.

Supported PCI Commands

Table 2 illustrates the PCI bus commands supported by the LogiCORE™ PCI Interface. The PCI Compliance Checklist, found in the *PCI Data Book*, has more details on supported and unsupported commands.

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. A FIFO to support PCI burst transfer can efficiently be implemented using the XC4000XLT on-chip RAM feature, SelectRAM™. Each XC4000XLT CLB supports two 16x1 RAM blocks. This corresponds to 32 bits of single-ported RAM or 16 bits of dual-ported RAM, with simultaneous read/write capability.

Wait States

In this version of the core, the wait state is switch selectable. The XC4013XLT-1 is able to run without wait states. The XC4028XLT-1 and the XC4062XLT-1 will require one wait state. Faster speed grades will allow zero wait state operation.

In the Zero Wait state mode, no wait states are inserted either while sourcing data or receiving data. This allows a 100% burst transfer rate in both directions with full PCI compliance. No additional wait states are inserted in response to a wait state from another agent on the bus. Either IRDY or TRDY is kept asserted until the current data phase ends.

In one wait state mode, the LogiCORE Interface automatically inserts a wait state when sourcing data (Initiator Write, Target Read) during a burst transfer. In this mode, the LogiCORE Interface can accept data at 100% burst transfer rate and supply data at 50%.

See Table 3 for a PCI bus transfer rates for various operations in either zero or one wait state mode.

Table 3: LogiCORE PCI Transfer Rates

Zero Wait State Mode	
Operation	**Transfer Rate**
Initiator Write (PCI ← LogiCORE)	3-1-1-2
Initiator Read (PCI → LogiCORE)	4-1-1-2
Target Write (PCI→ LogiCORE)	5-1-1-1
Target Read (PCI ← LogiCORE)	6-1-1-1
One Wait State Mode	
Operation	**Transfer Rate**
Initiator Write (PCI ← LogiCORE)	3-2-2-2
Initiator Read (PCI → LogiCORE)	4-1-1-2
Target Write (PCI→ LogiCORE)	5-1-1-1
Target Read (PCI ← LogiCORE)	6-2-2-2

Note:
Initiator Read and Target Write operations have effectively the same bandwidth for burst transfer.

Timing Specification

The XC4000XLT family, together with the LogiCORE PCI products enables design of fully compliant PCI systems. Backend design can affect the maximum speed your design is capable of. Factors in your back-end designs that can affect timing include loading of hot signals coming directly from the PCI bus, and gate count. Table 4 shows the key timing parameters for the LogiCORE PCI Interfaces that must be met for full PCI compliance.

Verification Methods

Xilinx has developed an internal testbench with numerous vectors to test the Xilinx PCI design. The LogiCORE PCI Interfaces have also been extensively simulated using the VirtualChips VHDL PCI test bench from Phoenix Technologies, Ltd. (not included with the LogiCORE PCI products). The Interface has also been verified in hardware in the XC4013XLT-1 PQ208C FPGA.

Included with the LogiCORE PCI Master and Slave Interface is an example design and a VIEW*logic* based PCI protocol test bench that verifies the PCI interface functions according to the test scenarios specified in the *PCI Local Bus Specification* see Figure 2. This test bench consists of 28 test scenarios, each designed to test compliance to a specific PCI bus operation. Refer to the checklist in the *PCI Data Book* for a complete list of test scenarios.

Table 4: Timing Parameters [ns]

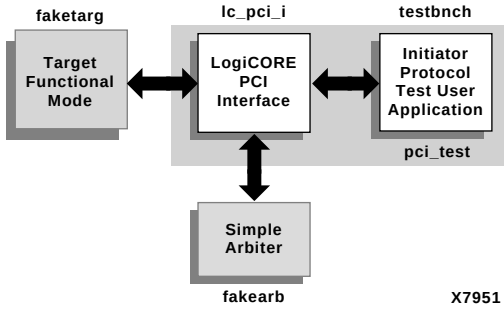
Parameter	Ref.	PCI Spec.		LogiCORE PCI, XC4000XLT-1	
		Min	Max	Min	Max
CLK Cycle Time		30	∞	30[1]	∞
CLK High Time		11		11	
CLK Low Time		11		11	
CLK to Bus Signals Valid[3]	T_{ICK-OF}	2	11	2[2]	8.5
CLK to REQ# and GNT# Valid[3]	T_{ICK-OF}	2	12	2[2]	8.5
Tri-state to Active		2		2[2]	
CLK to Tri-state			28		28[1]
Bus Signal Setup to CLK (IOB)	T_{PSD}		7		7
Bus Signal Setup to CLK (CLB)			7		7[1]
GNT# Setup to CLK	T_{PSD}		10		7
Input Hold Time After CLK (IOB)	T_{PHD}		0		0
Input Hold Time After CLK (CLB)			0		0[2]
RST# to Tri-state			40		40[2]

Notes:
1. Controlled by TIMESPECS, included in product
2. Verified by analysis and bench-testing
3. IOB configured for Fast slew rate

Ping Reference Design

The Xilinx LogiCORE™ PCI “PING” Application Example, delivered in VHDL and Verilog, has been developed to provide an easy-to-understand example which demonstrates many of the principles and techniques required to successfully use the LogiCORE™ PCI macro in a System On A Chip solution.

Figure 2: PCI Protocol Testbench



Synthesizable PCI Bridge Design Example

This synthesizable PCI bridge design is an example application bridge, delivered in Verilog and VHDL. This example demonstrates how to interface to the LogiCORE V2.0 PCI core and provides a modular foundation upon which to base other designs. See separate data sheet for details.

Recommended Design Experience

The LogiCORE PCI interface is pre-implemented allowing engineering focus at the unique back-end functions of a PCI design. Regardless, PCI is a high-performance system that is challenging to implement in any technology, ASIC or FPGA. Therefore, we recommend previous experience with building high-performance, pipelined FPGA designs using Xilinx implementation software, TIMESPECs, and guide files. The challenge to implement a complete PCI design including back-end functions varies depending on configuration and functionality of your application. Contact your local Xilinx representative for a closer review and estimation for your specific requirements.

Table 5: Part Numbers

Product	Part Number
LogiCORE PCI Master (Initiator/Target)	DO-DI-PCIM
LogiCORE PCI Slave (Target Only)	DO-DI-PCIS
LogiCORE PCI Master (Initiator/Target) Support Contract	SC-DI-PCIM-U
LogiCORE PCI Slave (Target Only) Support Contract	SC-DI-PCIS-U
LogiCORE PCI Slave to Master Upgrade	DX-DI-S2M
LogiCORE PCI Master (Initiator/Target) Software Renewal	SR-DI-PCIM
LogiCORE PCI Slave (Target Only) Software Renewal	SR-DI-PCIS

Ordering Information

Table 5 shows the part numbers for the LogiCORE™ products. Before placing an order, please read and sign the attached LogiCORE™ license agreement and fax it to Xilinx at +1 408-377-3259. For pricing and availability please contact your local Xilinx sales office.

Related Information

Recommended Design Centers

Listed below are design centers and design consultants that have experience with the LogiCORE™ PCI products.

HighGate Design
12380 Saratoga-Sunnyvale Road, Suite 8
Saratoga, CA 95070-3090, USA
Phone: +1 408-255-7160
Fax: +1 408-255-7162
E-mail: info@highgatedesign.com
URL: www.highgatedesign.com

Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, Arizona 85202, USA
Phone: +1 602-491-4311
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Comit Systems
1250 Oakmead Pkwy, Suite 210
Sunnyvale, CA 94088, USA
Phone: +1 408-988-2987
Fax: +1 408-988-2133
E-mail: preeth@comit.com
URL: www.comit.com

Austin Franklin
Dark Room Technologies, Inc.
126 Poor Farm Road
Harvard MA, 01451, USA
Phone: +1 508-772-9928
Fax: +1 508-772-4287
E-mail: darkroom@ix.netcom.com

PCI Special Interest Group (PCI-SIG) Publications

The PCI-SIG publishes various PCI specifications and related documents such as *PCI Local Bus Specification*, *PCI Compliance Checklist* and *PCI System Design Guide*.

PCI Special Interest Group
2575 NE Kathryn St. #17
Hillsboro, OR 97124
Phone: +1 800-433-5177 (inside the US)
+1 503-693-6232 (outside the US)
Fax: +1 503-693-8344
Office hours: 8:30am - 4:00pm PST
E-mail: info@pcisig.com
URL: www.pcisig.com

Xilinx Documents

More PCI related information is available on Xilinx Web:

www.xilinx.com/products/logicore/logicore.htm.

XILINX LOGICORE™ "PCI MASTER INTERFACE" AND "PCI SLAVE INTERFACE" LICENSE AGREEMENT

PLEASE READ THIS DOCUMENT CAREFULLY BEFORE USING THE **XILINX LOGICORE PCI MASTER INTERFACE** OR **PCI SLAVE INTERFACE DESIGN**, BY USING THE XILINX LOGICORE DESIGN, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE.

1. **License.** XILINX, INC. ("XILINX") hereby grants you a nonexclusive, nontransferable license to use the LOGICORE PCI MASTER/SLAVE INTERFACEdesign (the "Design"), solely for your use in developing designs for XILINX programmable logic devices or XILINX HardWire™ devices. Use of the Design in non-XILINX devices or technologies is prohibited unless you have entered into a separate written agreement with XILINX for such use. XILINX retains title to the Design and to any patents, copyrights, trade secrets and other intellectual property rights therein. To protect such intellectual property rights, you may not decompile, reverse engineer, disassemble, or otherwise reduce the Design to a human-perceivable form. You may not modify or prepare derivative works of the Design in whole or in part. This License allows you to make an unlimited number of copies of the Design for internal use only, provided you reproduce on each copy of the Design the copyright and any other proprietary legends that were on the original copy of the Design.
2. **Termination.** This License is effective until terminated. You may terminate this License at any time by destroying the Design and all copies thereof. This License will terminate immediately without notice from XILINX if you fail to comply with any provision of this License. Upon termination you must destroy the Design and all copies thereof.
3. **Governmental Use.** The Design is commercial computer software developed exclusively at Xilinx's expense. Accordingly, pursuant to the Federal Acquisition Regulations (FAR) Section 12.212 and Defense FAR Supplement Section 227.2702, use, duplication and disclosure of the Design by or for the Government is subject to the restrictions set forth in this License Agreement. Manufacturer is XILINX, INC., 2100 Logic Drive, San Jose, California 95124.
4. **Limited Warranty and Disclaimer.** THE DESIGN IS PROVIDED TO YOU "AS IS". XILINX AND ITS LICENSORS MAKE AND YOU RECEIVE NO WARRANTIES OR CONDITIONS, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, AND XILINX SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. XILINX does not warrant that the functions contained in the Design will meet your requirements, or that the operation of the Design will be uninterrupted or error free, or that defects in the Design will be corrected. Furthermore, XILINX does not warrant or make any representations regarding use or the results Design in terms of correctness, accuracy, reliability or otherwise.
5. **Limitation of Liability.** IN NO EVENT WILL XILINX OR ITS LICENSORS BE LIABLE FOR ANY LOSS OF DATA, LOST PROFITS, COST OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL OR INDIRECT DAMAGES ARISING FROM THE USE OR OPERATION OF THE DESIGN OR ACCOMPANYING DOCUMENTATION, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY. THIS LIMITATIN WILL APPLY EVEN IF XILINX HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THIS LIMITATION SHALL APPLY NOTWITHSTANDING THE FAILURE OF THE ESSENTIAL PURPOSE OF ANY LIMITED REMEDIES HEREIN.
6. **Export Restriction.** You agree that you will not export or re-export the Design, reference images or accompanying documentation in any form without the appropriate United States and foreign government licenses. Your failure to comply with this provision is a material breach of this Agreement.
7. **Third Party Beneficiary.** You understand that portions of the Design and related documentation may have been licensed to XILINX from third parties and that such third parties are intended third party beneficiaries of the provisions of this Agreement.
8. **Non-Transferable.** You may not provide design source information including, but not limited to, schematics, hardware description language source code, or netlist files, to a third party without prior written approval from XILINX. You may provide device programming files— XILINX bit-stream files or PROM files—of the resulting HardWire gate array to third-parties without prior approval.
9. **General.** This License shall be governed by the laws of the State of California, without reference to conflict of laws principles. If for any reason a court of competent jurisdiction finds any provision of this License, or portion thereof, to be unenforceable, that provision of this License shall be replaced to the maximum extent permissible so as to effectuate the intent of the parties, and the remainder of this License shall continue in full force and effect. This License constitutes the entire agreement between the parties with respect to the use of this Design and related documentation, and supersedes all prior or contemporaneous understandings or agreement, written or oral, regarding such subject matter.

I have read the Xilinx LogiCORE License Agreement, and I agreement to comply with the terms and conditions therein.

Name: _____ _____

(Signed) (Date)

Printed: _____

Title: _____ Company: _____

Address: _____ _____

E-mail: _____ PO#: _____

Fax to: +1 408-377-3259 (USA)

LogiCORE™

PCI Master & Slave Interfaces Version 1.2.0

May 25, 1997 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: Techsupport: hotline@xilinx.com
Feedback: logicore@xilinx.com
URL: www.xilinx.com

Features

- Fully 2.1 PCI compliant 32 bit, 33MHz PCI Interface
  - Master (Initiator/Target)
  - Slave (Target-only)
- Pre-defined implementation for predictable timing in Xilinx XC4000E FPGAs or HardWire (see *LogiCORE Facts* for listing of supported devices)
- Fully verified design
  - Simulated using VirtualChips™ PCI testbench
  - Tested in hardware (silicon proven)
- Configurable on-chip FIFO can be added for maximum burst speed (see *Xilinx Documents* section)
- Programmable single-chip solution with customizable back-end functionality
- Design Once™ - automatic conversion to HardWire for cost reduction
- Supported Initiator functions (PCI Master only)
  - Initiate Memory Read, Memory Write, Memory Read Multiple (MRM), Memory Read Line (MRL) commands
  - Initiate I/O Read, I/O Write commands
  - Initiate Configuration Read, Configuration Write commands
  - Bus Parking
- Supported Target functions (PCI Master and Slave)
  - Type 0 Configuration Space Header
  - Up to 2 Base Address Registers (memory or I/O with adjustable block size from 16 bytes to 256 Mbytes, slow decode speed)
  - Parity Generation (PAR), Parity Error Detection (PERR# and SERR#)
  - Memory Read, Memory Write, Memory Read Multiple (MRM), Memory Real Line (MRL), Memory Write, Invalidate (MWI) commands
  - I/O Read, I/O Write commands
  - Configuration Read, Configuration Write commands

LogiCORE™ Facts

Core Specifics	
Device Family	XC4000E
CLBs Used	152 - 268
IOBs Used[5]	53/51
System Clock f_{max}	0 – 33MHz
Device Features Used	Bi-directional data buses SelectRAM™ (optional user FIFO) Boundary scan (optional)

Supported Devices[1]/Resources Remaining		
	I/O[5]	CLB[2]
XC4013E PQ160[6]	-/76	308 - 424
XC4013E PQ208	107/109	308 - 424
XC4013E HQ240	141/143	308 - 424
XC4020E HQ208	107/109	516 - 632
XC4020E HQ240	141/143	516 - 632

Provided with Core	
Documentation	User's Guide PCI Interface Protocol Checklist V1.0 to V1.1 Comparison Ping Ref. Design Users Guide
Design File Formats	VIEW*logic* schematics XNF Netlist[3]
Constraint Files	TimeSpecs, RPMs, Guide files, sample synthesis script
Verification Tool	VIEW*logic* command files VHDL Testbench Verilog Testbench
Schematic Symbols	VIEWlogic, VHDL, Verilog
Evaluation Model	VHDL, Verilog Simulation Model[3]
Reference designs & application notes	Example design Ping Reference Design[4]
Additional Items	Reference book: PCI System Architectures

Design Tool Requirements	
Xilinx Core Tools	XACTstep 5.2.1/6.0.1
Entry/Verification Tools	For CORE generator: VHDL, Verilog, Schematic For changing source files: Workview Office V7.1.2 or V7.2

Notes: See next page.

LogiCORE™ Facts (cont.)

Support

Xilinx provides technical support for this LogiCORE product when used as described in the User's Guide or supporting Application Notes. Xilinx cannot guarantee timing, functionality, or support of the product if implemented in devices not listed above, or customized beyond that referenced in the product documentation, or if any changes are done in sections of the design marked as "DO NOT MODIFY".

Notes:
1. Speed grade is determined by PCI configuration and user back-end, see Figure 3.
2. The exact number of CLBs and depends on user configuration of the core and level of resource sharing with adjacent logic.
3. Available on Xilinx Home Page, in the LogiCORE PCI VIP Lounge: www.xilinx.com/products/logicore/logicore.htm
4. See heading "Ping Reference Design"
5. Master/Slave
6. Slave only

Features (cont.)

- 32-bit data transfers, burst transfers with linear address ordering
- Target Abort, Target Retry, Target Disconnect
- Full Command/Status Register

• Supported by Xilinx CORE Generator
- Web-based configuration
- Generation of proven design files

Applications

• Add-in boards such as graphic cards, video adapters, LAN adapters and data acquisition boards.
• Embedded applications within telecommunication and industrial systems.
• Other applications that need PCI

General Description

The LogiCORE™ Master and Slave Interfaces are pre-implemented and fully tested modules for Xilinx XC4000E FPGAs (see *LogiCORE Facts* for listing of supported devices). The pin-out and the relative placement of the internal Configurable Logic Blocks (CLBs) are pre-defined. Critical paths are controlled by TimeSpec's to ensure that timing is always met. This significantly reduces engineering time required to implement the PCI portion of your design. Resources can instead be focused on the unique back-end logic in the FPGA and the system level design. As a result, the LogiCORE PCI products can cut your development time by several months.

Xilinx XC4000E Series FPGAs enables designs of fully PCI-compliant systems. The devices meet all required electrical and timing parameters including AC output drive characteristics, input capacitance specifications (10pF), 7 ns setup and 0 ns hold to system clock, and 11 ns system clock to output. The *PCI Compliance Checklist XC4000E* has additional details (see the *Xilinx Documents* section). Other features that enable efficient implementation of a complete PCI system in the XC4000E includes:

• Select-RAM™ memory: on-chip ultra-fast RAM with synchronous write option and dual-port RAM option. Used in the PCI Interfaces to implement the FIFO.
• Individual output enable for each I/O

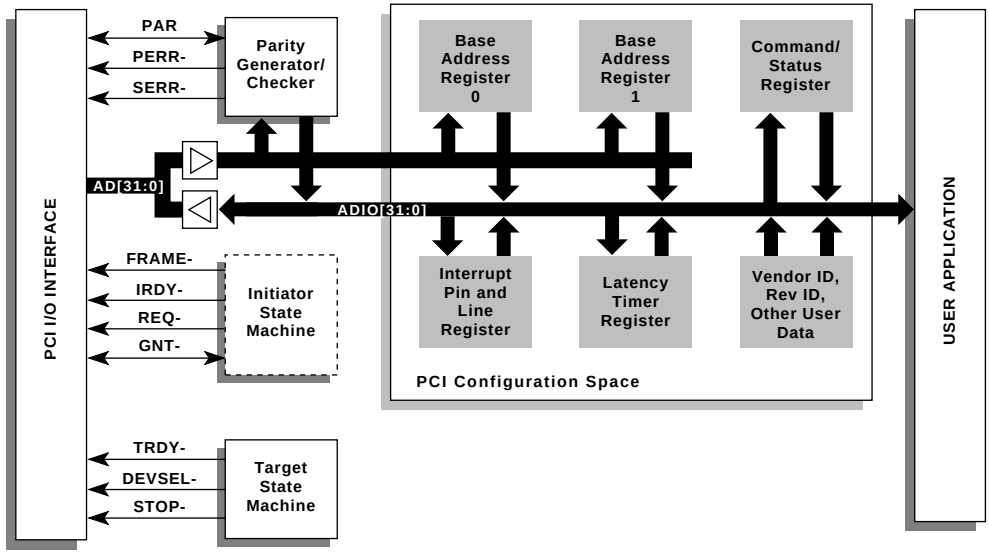


Figure 1: LogiCORE PCI Interface Block Diagram

- Internal 3-state bus capability
- 8 global low-skew clock or signal distribution networks
- IEEE 1149.1-compatible boundary scan logic support

See *Xilinx 1996 Data Book* for more details.

The module is carefully optimized for best possible performance and utilization in the XC4000E FPGA architecture. Implemented in the smallest supported FPGA, XC4013, more than 50% of the FPGA's resources remain for integrating a unique back-end interface and other system functions into a fully programmable one-chip solution.

Xilinx DesignOnce™ service allows an automatic conversion to a low cost HardWire™ device for high-volume production.

Functional Description

The LogiCORE PCI Master Interface is partitioned into five major blocks, plus the user application, shown in Figure 1. Each block is described below.

PCI I/O Interface Block

The I/O interface 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.

Parity Generator/Checker

Generates/checks even parity across the AD bus, the CBE lines, and the PAR signal. Reports data parity errors via PERR- and address parity errors via SERR-.

Target State Machine

This block manages control over the PCI interface for Target functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The controller is a high-performance state machine using state-per-bit (one-hot) encoding for maximum performance. State-per-bit encoding has narrower and shallower next-state logic functions that closely match the Xilinx FPGA architecture.

Initiator State Machine (PCI Master only)

This block manages control over the PCI interface for Initiator functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The Initiator Control Logic also uses state-per-bit encoding for maximum performance.

PCI Configuration Space

This block provides the first 64 bytes of Type 0, version 2.1, Configuration Space Header (CSH) (see Table 1) to support software-driven "Plug-and Play" initialization and configuration. This includes Command, Status, and two Base Address Registers (BARs). These BARs illustrate how to implement memory- or I/O-mapped address spaces. Each BAR sets the base address for the interface and allows the system software to determine the addressable range required by the interface. Using a combination of Configurable Logic Block (CLB) flip-flops for the read/write registers and CLB look-up tables for the read-only registers results in optimized packing density and layout.

Table 1: 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
Base Address Register 4 (BAR5)				20h
Base Address Register 5 (BAR5)				24h
Cardbus CIS Pointer				28h
Subsystem ID		Subsystem Vendor ID		2Ch
Expansion ROM Base Address				30h
Reserved				34h
Reserved				38h
Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	3Ch

Note:
Italicized address areas are not implemented in the LogiCORE PCI Interface default configuration. These locations return zero during configuration read accesses.

User Application with Optional Burst FIFOs

The LogiCORE PCI Interface provides a simple, general-purpose interface with a 32-bit data path and latched address for de-multiplexing the PCI address/data bus. The general-purpose user interface allows the rest of the device to be used in a wide range of applications.

Typically, the user application contains burst FIFOs to increase PCI system performance (An Application Note is available, please see the *Xilinx Documents* section). An on-chip read/write FIFO, built from the on-chip synchronous

dual-port RAM (SelectRAM™) available in XC4000E devices, supports data transfers in excess of 33 MHz.

Core Configuration

The LogiCORE PCI Interface can easily be configured to fit unique system requirements using Xilinx web-based CORE Generator or changing the Viewlogic schematics. Following customization is supported by the LogiCORE product and described in accompanying documentation.

- Initiator or target functionality (PCI Master only)
- Base Address Register configuration (1 - 2 Registers, size and mode)
- Configuration Space Header ROM
- Initiator and target state machine (e.g., termination conditions, transaction types and request/transaction arbitration)
- Burst functionality
- User Application including FIFO (back-end design)

Table 2: PCI Bus Commands

CBE [3:0]	Command	PCI Master	PCI Slave
0000	Interrupt Acknowledge	No[1]	Ignore
0001	Special Cycle	No[1]	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[1]	Ignore
1110	Memory Read Line	Yes	Yes
1111	Memory Write Invalidate	No[1]	Yes

Note:
1. The Initiator can present these commands, however, they either require additional user-application logic to support them or have not been thoroughly tested.

Supported PCI Commands

Table 2 illustrates the PCI bus commands supported by the LogiCORE PCI Interface. The *LogiCORE™ PCI Interface Protocol Checklist (2.1)* has more details on supported and unsupported commands (see the *Xilinx Documents* section)

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. A FIFO to support PCI burst transfer can efficiently be implemented using the XC4000E on-chip RAM feature, SelectRAM™. Each XC4000E CLB supports two 16x1 RAM blocks. This corresponds to 32 bits of single-ported RAM or 16 bits of dual-ported RAM, with simultaneous read/write capability. Table provides a summary of different FIFO sizes and performance in an XC4000E-2.

To reliably perform a burst transfer in a generic PCI system the LogiCORE Interface automatically inserts a wait state when it is supplying data to the PCI bus. Consequently, the LogiCORE Interface can accept data at 100% burst transfer rate and supply data at 50%. See Table 4 for a PCI bus transfer rates for various operations.

The resulting PCI bus bandwidth is shown in Figure 2.

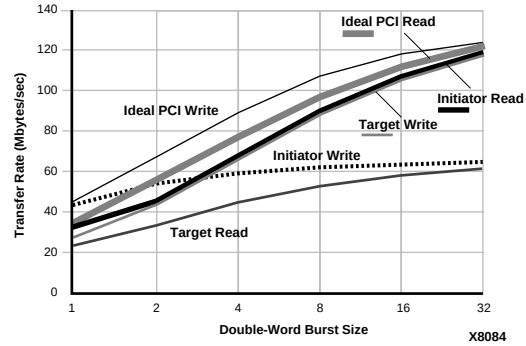


Figure 2: PCI Bus Bandwidth

Pinout

The LogiCORE™ PCI Master and Slave Interfaces support the PCI-SIG recommended pin-out for add-in cards. Tables 5 to 7 describe the signals and pin-out for the LogiCORE PCI Master and Slave Interfaces. See the *LogiCORE PCI Master and Slave Interface User's Guide* for a detailed signal description.

Table 3: XC4000E-2 Synchronous FIFO Modules

Depth x Width	# CLBs	Single Port Performance	Equivalent Dual-port Performance
16x16	23	65 MHz	130 MHz
32x8	28	50 MHz	100 MHz
64x8	48	50 MHz	100 MHz
16x32	48	50 MHz	100 MHz

Table 4: LogiCORE PCI Transfer Rates

Operation	Transfer Rate
Initiator Write (PCI ← LogiCORE)	3-2-2-2
Initiator Read (PCI → LogiCORE)	4-1-1-2
Target Write (PC I→ LogiCORE)	5-1-1-1
Target Read (PCI ← LogiCORE)	6-2-2-2

Note:
Initiator Read and Target Write operations have effectively the same bandwidth for burst transfer.

Table 5: Signal Description (internal signals)

Signal Name	Signal Description	Target	Initiator
Address/Data			
AD[31:0]	PCI Address/Data Bus	I/O	I/O
CBE[3:0]	Command/Byte Enable	In	I/O
PAR	PCI Parity Signal	I/O	I/O
Interface			
FRAME	Frame	In	I/O
TRDY	Target Ready	Out	I/O
IRDY	Initiator Ready	In	I/O
STOP	Stop	Out	I/O
DEVSEL	Device Select	Out	I/O
IDSEL	Initialization Device Select	In	In
LOCK	Lock	In	I/O
Interrupts			
INTD	Interrupt D	N/A	N/A
INTC	Interrupt C	N/A	N/A
INTB	Interrupt B	N/A	N/A
INTA	Interrupt A	OD	OD
Cache (not supported)			
SDONE	PCI SDONE	N/A	N/A
SBO	PCI SBO	N/A	N/A
Error Signals			
PERR	Parity Error	Out	I/O
SERR	System Error	OD	OD
Arbitration			
REQ	Request PCI Bus	N/A	Out
GNT	Grant PCI Bus	N/A	In
Boundary Scan			
TDI	Test Data Input	In	In
TMS	Test Mode Select	In	In
TCK	Test Clock	In	In
TDO	Test Data Output	Out	Out
Miscellaneous			
RST	Global Reset	In	In
CLK	PCI Clock	In	In

Note:
Signals marked N/A are not applicable or not supported. Signals marked OD are Open Drain output.

Table 6: Signal Description (internal signal)

Signal Name	Signal Description	Target	Initiator
Cycle Control			
FRAME-	Frame	Out	Out
IRDY-	Initiator Ready	Out	Out
TRDY-	Target Ready	Out	Out
STOP-	Stop Transaction	Out	Out
DEVSEL-	Device Selected	Out	Out
Bus Control			
BASE_HIT	Base Address Hit	Out	Out
ADDR_VLD	Address Valid	Out	Out
DATA_VLD	Data Valid	Out	Out
CNFG_VLD	Configuration Valid	Out	Out
S_WRDN	Slave Write/Read Dir.	Out	Out
PCI_CMD[15:0]	PCI Bus Command	Out	Out
S_CBE[3:0]	Slave Comm/Byte Enable	Out	Out
Address/Data			
ADDR[31:0]	Latched Address Bus	Out	Out
ADIO[31:0]	Address/Data Bus	I/O	I/O
PERR-	Data Parity Error	In	In
User Control			
READY	Ready	In	In
TERM	Terminate Transaction	In	In
T_ABORT	Target Abort	In	In
SRC_EN	Source Data Enable	Out	Out
INTR-	Interrupt	In	In
KEEPOUT	Keep Out	In	In
Initiator Only Functions			
REQUEST	Request Transaction	N/A	In
M_CBE[3:0]	Master Comm/Byte Enable	N/A	In
M_WRDN	Master Write/Read Dir.	N/A	In
COMPLETE	Complete Transaction	N/A	In
TIME_OUT	Latency Timer Timeout	N/A	Out
State Bits (Initiator)			
M_DATA	Data Transfer State	N/A	Out
DR_BUS	Bus Parked	N/A	Out
M_ADDR-	Addr. State/Bus Parking	N/A	Out
I_IDLE	Initiator Idle State	N/A	Out
State Bits (Target)			
IDLE	Target Idle State	Out	Out
B_BUSY	PCI Bus Busy	Out	Out
S_DATA	Data Transfer State	Out	Out
BACKOFF	TERM Asserted	Out	Out
FREE	Free State	Out	Out
LOCKED	Locked State	Out	Out
Status Output			
CSR[39:0]	Extended Comm/Status	Out	Out

Note:
Signals marked I/O are using internal tri-state.

Table 7: LogiCORE PCI Pinout

PCI Signal	PQ160	PQ/HQ208	HQ240
AD31	147	192	225
AD30	148	193	226
AD29	149	195	228
AD28	150	196	229
AD27	152	197	230
AD26	153	198	231
AD25	154	199	232
AD24	155	200	233
AD23	158	6	4
AD22	3	7	5
AD21	4	10	8
AD20	5	11	9
AD19	8	12	10
AD18	9	13	11
AD17	11	15	15
AD16	12	16	16
AD15	30	33	41
AD14	31	34	42
AD13	32	35	43
AD12	33	36	44
AD11	34	38	48
AD10	35	39	49
AD9	36	40	50
AD8	45	41	51
AD7	47	60	66
AD6	49	61	67
AD5	50	63	69
AD4	52	64	70
AD3	53	65	71
AD2	54	66	72
AD1	55	68	76
AD0	56	69	77
CBE3	156	201	236
CBE2	14	18	18
CBE1	28	32	36
CBE0	46	59	65
PAR	26	30	34
FRAME-	16	20	24
TRDY-	18	22	26
IRDY-	17	21	25

PCI Signal	PQ160	PQ/HQ208	HQ240
STOP-	22	24	28
DEVSEL-	21	23	27
IDSEL	157	203	238
LOCK-	23	27	31
INTA			
PERR-	24	28	32
SERR-	25	29	33
REQ	N/A	31	35
GNT	N/A	19	23
TDI	6	8	6
TMS	13	17	17
TCK	7	9	7
TDO	121	159	181
RST-	83	109	123
CLK	2	4	2

Timing Specification

The XC4000E family, together with the LogiCORE PCI products enables design of fully compliant PCI systems. The choice of FPGA speed grade for your PCI application is determined by the PCI configuration and your back-end design as illustrated in Figure 3. Factors affecting your back-end designs include loading of hot signals coming directly from the PCI bus, gate count and floor planning. Table 8 shows the key timing parameters for the LogiCORE PCI Interfaces that must be met for full PCI compliance.

Figure 3: Choice of Speed Grade

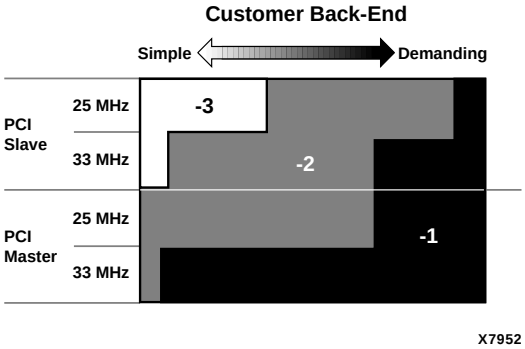


Table 8: Timing Parameters [ns]

Parameter	Ref.	PCI Spec.		LogiCORE PCI, XC4000E-2/1	
		Min	Max	Min	Max
CLK Cycle Time		30	∞	30[1]	∞
CLK High Time		11		11	
CLK Low Time		11		11	
CLK to Bus Signals Valid[4]	T_{ICK-OF}	2	11	2[2]	9.4[3]
CLK to REQ# and GNT# Valid[4]	T_{ICK-OF}	2	12	2[2]	9.4[3]
Tri-state to Active		2		2[2]	
CLK to Tri-state			28		28[1]
Bus Signal Setup to CLK (IOB)	T_{PSU}		7		6[3]
Bus Signal Setup to CLK (CLB)			7		7[1]
GNT# Setup to CLK	T_{PSU}		10		6[3]
Input Hold Time After CLK (IOB)	T_{PH}		0		0[3]
Input Hold Time After CLK (CLB)			0		0[2]
RST# to Tri-state			40		40[2]

Notes:
1. Controlled by TimeSpecs, included in product
2. Verified by analysis and bench-testing
3. Advanced speed grade data
4. IOB configured for Fast slew rate

Verification Methods

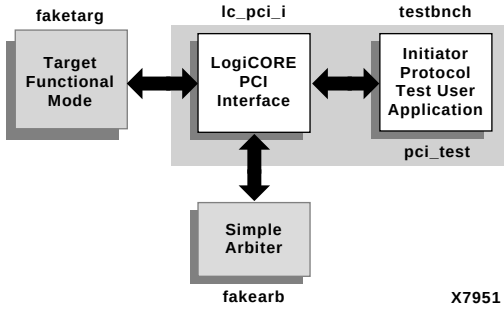
The LogiCORE PCI Interfaces have been extensively simulated using the VirtualChips VHDL PCI test bench from Phoenix Technologies, Ltd. (not included with the LogiCORE™ PCI products). The Interface has also been verified in hardware in the XC4013E-2 PQ208C FPGA.

Included with the LogiCORE™ PCI Master and Slave Interface is an example design and a VIEW*logic* based PCI protocol test bench that verifies the PCI interface functions according to the test scenarios specified in the *PCI Local Bus Specification* see Figure 4. This test bench consists of 28 test scenarios, each designed to test compliance to a specific PCI bus operation. Refer to the *LogiCORE PCI Interface Protocol Checklist* for a complete list of supported test scenarios (see the *Xilinx Documents* section).

Ping Reference Design

The Xilinx LogiCORE PCI “PING” Application Example, delivered in VHDL and Verilog, has been developed to provide an easy-to-understand example which demonstrates many of the principles and techniques required to successfully use the LogiCORE™ PCI macro in a System On A Chip solution.

Figure 4: PCI Protocol Testbench



Recommended Design Experience

The LogiCORE PCI interface is pre-implemented allowing engineering focus at the unique back-end functions of a PCI design. Regardless, PCI is a high-performance system that is challenging to implement in any technology, ASIC or FPGA. Therefore, we recommend previous experience with building high-performance, pipelined FPGA designs using Xilinx implementation software, Floorplanner, TIMESPECs, and guide files. The challenge to implement a complete PCI design including back-end functions varies depending on configuration and functionality of your application. Contact your local Xilinx representative for a closer review and estimation for your specific requirements.

Table 9: Part Numbers

Product	Part Number	Supplier
LogiCORE™ PCI Master (Intiator/Target)	LC-DI-PCIM-C	Xilinx, Inc.
LogiCORE™ PCI Slave (Target Only)	LC-DI-PCIS-C	Xilinx, Inc.

Ordering Information

Table 9 shows the part numbers for the LogiCORE™ products. Before placing an order, please read and sign the attached LogiCORE™ license agreement and fax it to Xilinx at +1 408-879-4780. For pricing and availability please contact your local Xilinx sales office.

Related Information

Recommended Design Centers

Listed below are design centers and design consultants that have experience with the LogiCORE PCI products.

HighGate Design
12380 Saratoga-Sunnyvale Road, Suite 8
Saratoga, CA 95070-3090, USA
Phone: +1 408-255-7160
Fax: +1 408-255-7162
E-mail: highgate@highgatedesign.com
URL: www.highgatedesign.com

Memec Design Services
1819 S. Dobson Rd., Ste. 203
Mesa, Arizona 85202, USA
Phone: +1 602-491-4311
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Comit Systems
1250 Oakmead Pkwy, Suite 210
Sunnyvale, CA 94088, USA
Phone: +1 408-988-2988
Fax: +1 408-988-2133
E-mail: preeth@comit.com
URL: www.comit.com

PCI Special Interest Group (PCI-SIG) Publications

The PCI-SIG publishes various PCI specifications and related documents such as *PCI Local Bus Specification*, *PCI Compliance Checklist* and *PCI System Design Guide*.

PCI Special Interest Group
2575 NE Kathryn St #17
Hillsboro, OR 97124
Phone: +1 800-433-5177 (inside the US)
+1 503-693-6232 (outside the US)
Fax: +1 503-693-8344
Office hours: 8:30am - 4:00pm PST
E-mail: info@pcisig.com
URL: www.pcisig.com

Xilinx Documents

More PCI related information is available on Xilinx Web: www.xilinx.com/products/logicore/logicore.htm.

XILINX LOGICORE™ "PCI MASTER INTERFACE" AND"PCI SLAVE INTERFACE" LICENSE AGREEMENT

PLEASE READ THIS DOCUMENT CAREFULLY BEFORE USING THE **XILINX LOGICORE PCI MASTER INTERFACE** OR **PCI SLAVE INTERFACE DESIGN**, BY USING THE XLINX LOGICORE DESIGN, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE.

1. **License.** XLINX, INC. ("XILINX") hereby grants you a nonexclusive, nontransferable license to use the LOGICORE PCI MASTER/SLAVE INTERFACEdesign (the "Design"), solely for your use in developing designs for XLINX programmable logic devices or XLINX HardWire™ devices. Use of the Design in non-XILINX devices or technologies is prohibited unless you have entered into a separate written agreement with XILINX for such use. XILINX retains title to the Design and to any patents, copyrights, trade secrets and other intellectual property rights therein. To protect such intellectual property rights, you may not decompile, reverse engineer, disassemble, or otherwise reduce the Design to a human-perceivable form. You may not modify or prepare derivative works of the Design in whole or in part. This License allows you to make an unlimited number of copies of the Design for internal use only, provided you reproduce on each copy of the Design the copyright and any other proprietary legends that were on the original copy of the Design.
2. **Termination.** This License is effective until terminated. You may terminate this License at any time by destroying the Design and all copies thereof. This License will terminate immediately without notice from XILINX if you fail to comply with any provision of this License. Upon termination you must destroy the Design and all copies thereof.
3. **Governmental Use.** The Design is commercial computer software developed exclusively at Xilinx's expense. Accordingly, pursuant to the Federal Acquisition Regulations (FAR) Section 12.212 and Defense FAR Supplement Section 227.2702, use, duplication and disclosure of the Design by or for the Government is subject to the restrictions set forth in this License Agreement. Manufacturer is XILINX, INC., 2100 Logic Drive, San Jose, California 95124.
4. **Limited Warranty and Disclaimer.** THE DESIGN IS PROVIDED TO YOU "AS IS". XILINX AND ITS LICENSORS MAKE AND YOU RECEIVE NO WARRANTIES OR CONDITIONS, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, AND XILINX SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. XILINX does not warrant that the functions contained in the Design will meet your requirements, or that the operation of the Design will be uninterrupted or error free, or that defects in the Design will be corrected. Furthermore, XILINX does not warrant or make any representations regarding use or the results Design in terms of correctness, accuracy, reliability or otherwise.
5. **Limitation of Liability.** IN NO EVENT WILL XILINX OR ITS LICENSORS BE LIABLE FOR ANY LOSS OF DATA, LOST PROFITS, COST OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL OR INDIRECT DAMAGES ARISING FROM THE USE OR OPERATION OF THE DESIGN OR ACCOMPANYING DOCUMENTATION, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY. THIS LIMITATIN WILL APPLY EVEN IF XILINX HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THIS LIMITATION SHALL APPLY NOTWITHSTANDING THE FAILURE OF THE ESSENTIAL PURPOSE OF ANY LIMITED REMEDIES HEREIN.
6. **Export Restriction.** You agree that you will not export or re-export the Design, reference images or accompanying documentation in any form without the appropriate United States and foreign government licenses. Your failure to comply with this provision is a material breach of this Agreement.
7. **Third Party Beneficiary.** You understand that portions of the Design and related documentation may have been licensed to XILINX from third parties and that such third parties are intended third party beneficiaries of the provisions of this Agreement.
8. **Non-Transferable.** You may not provide design source information including, but not limited to, schematics, hardware description language source code, or netlist files, to a third party without prior written approval from XILINX. You may provide device programming files— XILINX bit-stream files or PROM files—of the resulting HardWire gate array to third-parties without prior approval.
9. **General.** This License shall be governed by the laws of the State of California, without reference to conflict of laws principles. If for any reason a court of competent jurisdiction finds any provision of this License, or portion thereof, to be unenforceable, that provision of this License shall be replaced to the maximum extent permissible so as to effectuate the intent of the parties, and the remainder of this License shall continue in full force and effect. This License constitutes the entire agreement between the parties with respect to the use of this Design and related documentation, and supersedes all prior or contemporaneous understandings or agreement, written or oral, regarding such subject matter.

I have read the Xilinx LogiCORE License Agreement, and I agreement to comply with the terms and conditions therein.

Name: _____ _____

(Signed) (Date)

Printed: _____

Title: _____ Company: _____

Address: _____ _____

E-mail: _____ PO#: _____

Fax to: +1 408-879-4780 (USA)

LogiCORE™

PCI32 Spartan Master & Slave Interfaces

January 12, 1998 Advanced Data Sheet

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-377-3259
E-mail: Techsupport: hotline@xilinx.com
Feedback: logicore@xilinx.com
URL: www.xilinx.com

Introduction

With Xilinx LogiCORE PCI32 Spartan Master and Slave interfaces, a designer can build a customizable, low-cost 32-bit, 33MHz fully PCI compliant system in a Spartan-family FPGA.

Features

- Fully 2.1 PCI compliant 32 bit, 33MHz PCI Interface
  - Master (Initiator/Target)
  - Slave (Target-only)
- Pre-defined implementation for predictable timing in Xilinx Spartan FPGAs (see *LogiCORE Facts* for listing of supported devices)
- 5 V Operation with Spartan devices
- Fully verified design
  - Tested with the Xilinx internal testbench
  - Tested in hardware (silicon proven)
- Configurable on-chip FIFO can be added for maximum burst speed (see *Xilinx Documents* section)
- Programmable single-chip solution with customizable back-end functionality
- Supported Initiator functions (PCI Spartan Master only)
  - Initiate Memory Read, Memory Write, Memory Read Multiple (MRM), Memory Read Line (MRL) commands
  - Initiate I/O Read, I/O Write commands
  - Initiate Configuration Read, Configuration Write commands
  - Bus Parking

LogiCORE™ Facts

Core Specifics	
Device Family	XCS Spartan
CLBs Used	TBD
IOBs Used[1]	53/51
System Clock f_{max}	0 – 33MHz
Device Features Used	Bi-directional data buses SelectRAM™ (optional user FIFO) Boundary scan (optional)

Supported Devices/Resources Remaining		
	I/O[1]	**CLB[2]**
XCS30 PQ208	107/109	TBD
XCS30 PQ240	141/143	TBD
XCS40 PQ208	107/109	TBD
XCS40 PQ240	141/143	TBD

Provided with Core	
Documentation (on-line only)	*PCI Spartan User's Guide* *PCI Data Book*
Design File Formats	EDIF Netlist[3]
Constraint Files	M1 User Constraint File (UCF) M1 Guide files
Verification Tools	VHDL Testbench Verilog Testbench
Schematic Symbols	VHDL, Verilog
Simulation Model	VHDL, Verilog Simulation Model[3]
Reference designs & application notes	TBD
Additional Items	TBD

Design Tool Requirements	
Xilinx Core Tools	M1.4
Entry/Verification Tools	For CORE generator: VHDL, Verilog, Schematic

Notes: See next page.

LogiCORE™ Facts (cont.)

Support

Xilinx provides technical support for this LogiCORE product when used as described in the User's Guide or supporting Application Notes. Xilinx cannot guarantee timing, functionality, or support of the product if implemented in devices not listed above, or customized beyond that referenced in the product documentation, or if any changes are done in sections of the design marked as "DO NOT MODIFY".

Notes:
1. Master/Slave
2. The exact number of CLBs depends on user configuration of the core and level of resource sharing with adjacent logic. Factors that can affect the size of the design are number and size of the BARs, zero vs. one wait state, and medium vs. slow decode. These numbers include a 16 x 32 FIFO.
3. Available on Xilinx Home Page, in the LogiCORE PCI VIP Lounge: www.xilinx.com/products/logicore/pci/pci_sol.htm

Features (cont.)

- Supported Target functions (PCI Master and Slave)
  - Type 0 Configuration Space Header
  - Up to 2 Base Address Registers (memory or I/O with adjustable block size from 16 bytes to 2 GBytes, slow decode speed)
  - Parity Generation (PAR), Parity Error Detection (PERR# and SERR#
  - ACPI Configuration Registers (backend module)
  - Memory Read, Memory Write, Memory Read Multiple (MRM), Memory Real Line (MRL), Memory Write, Invalidate (MWI) commands
  - I/O Read, I/O Write commands
  - Configuration Read, Configuration Write commands
  - 32-bit data transfers, burst transfers with linear address ordering
  - Target Abort, Target Retry, Target Disconnect
  - Full Command/Status Register
  - Supported by Xilinx CORE Generator
  - Web-based configuration
  - Generation of proven design files

Applications

- PCI add-in boards such as graphic cards, video adapters, LAN adapters and data acquisition boards.
- Embedded applications within telecommunication and industrial systems.
- CompactPCI boards
- Other applications that need PCI

General Description

The LogiCORE™ PCI Spartan Master and Slave Interfaces are pre-implemented and fully tested modules for Xilinx Spartan FPGAs (see *LogiCORE Facts* for listing of supported devices). The pin-out and the relative placement of the internal Configurable Logic Blocks (CLBs) are predefined. Critical paths are controlled by TimeSpec's to ensure that timing is always met. This significantly reduces engineering time required to implement the PCI portion of your design. Resources can instead be focused on the unique back-end logic in the FPGA and the system level design. As a result, the LogiCORE PCI products can cut your development time by several months.

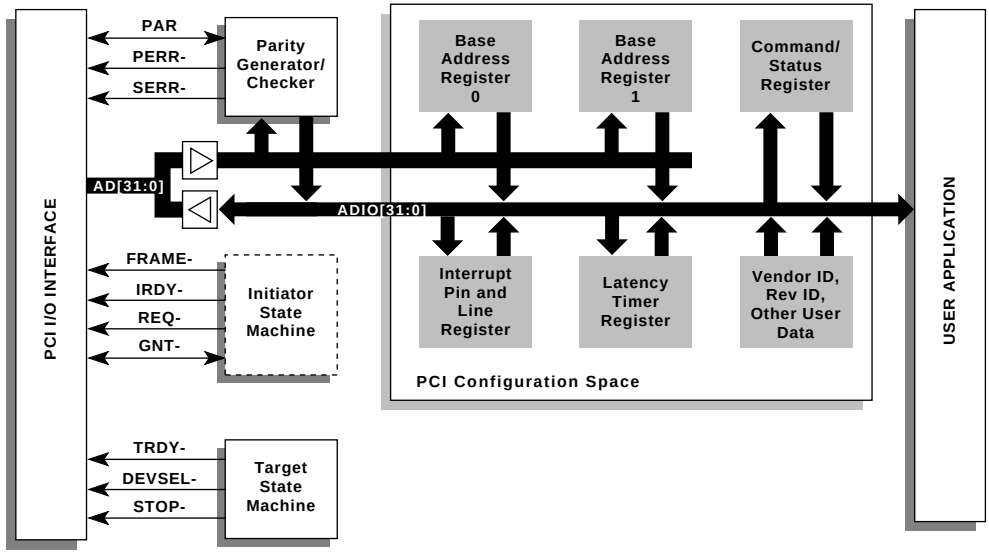


Figure 1. LogiCORE PCI Interface Block Diagram

Xilinx Spartan Series FPGAs enables designs of fully PCI-compliant systems. The devices meet all required electrical and timing parameters including AC output drive characteristics, input capacitance specifications (10pF), 7 ns setup and 0 ns hold to system clock, and 11 ns system clock to output. These devices meet all specifications for 5 V PCI.

The *PCI Compliance Checklist* has additional details (see the *PCI Data Book*). Other features that enable efficient implementation of a complete PCI system in the Spartan family includes:

- Select-RAM™ memory: on-chip ultra-fast RAM with synchronous write option and dual-port RAM option. Used in the PCI Interfaces to implement the FIFO.
- Individual output enable for each I/O
- Internal 3-state bus capability
- 8 global low-skew clock or signal distribution networks
- IEEE 1149.1-compatible boundary scan logic support

See *Xilinx 1996 Data Book* for more details.

The module is carefully optimized for best possible performance and utilization in the Spartan FPGA architecture. When implemented in the XC30, more than 50% of the FPGA's resources remain for integrating a unique back-end interface and other system functions into a fully programmable one-chip solution.

Functional Description

The LogiCORE PCI Master Interface is partitioned into five major blocks, plus the user application, shown in Figure 1. Each block is described below.

PCI I/O Interface Block

The I/O interface 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.

Parity Generator/Checker

Generates/checks even parity across the AD bus, the CBE lines, and the PAR signal. Reports data parity errors via PERR- and address parity errors via SERR-.

Target State Machine

This block manages control over the PCI interface for Target functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The controller is a high-performance state machine using state-per-bit (one-hot) encoding for maximum performance. State-per-bit encoding has narrower and shallower next-state logic functions that closely match the Xilinx FPGA architecture.

Initiator State Machine (PCI Master only)

This block manages control over the PCI interface for Initiator functions. The states implemented are a subset of equations defined in "Appendix B" of the *PCI Local Bus Specification*. The Initiator Control Logic also uses state-per-bit encoding for maximum performance.

PCI Configuration Space

This block provides the first 64 bytes of Type 0, version 2.1, Configuration Space Header (CSH) (see Table 1) to support software-driven "Plug-and Play" initialization and configuration. This includes Command, Status, and two Base Address Registers (BARs). These BARs illustrate how to implement memory- or I/O-mapped address spaces. Each BAR sets the base address for the interface and allows the system software to determine the addressable range required by the interface. Using a combination of Configurable Logic Block (CLB) flip-flops for the read/write registers and CLB look-up tables for the read-only registers results in optimized packing density and layout.

With this release, the hooks for extending configuration space has been built into the backend interface. Setting the CapPtr and bit 15 of the Status Register allows the user to implement functions such as Advanced Configuration and Power Interface (ACPI) in the backend design. The ACPI module, whcih attaches to the backend, is included in this release.

User Application with Optional Burst FIFOs

The LogiCORE PCI Interface provides a simple, general-purpose interface with a 32-bit data path and latched address for de-multiplexing the PCI address/data bus. The general-purpose user interface allows the rest of the device to be used in a wide range of applications.

Typically, the user application contains burst FIFOs to increase PCI system performance (An Application Note is available, please see the *Xilinx Documents* section). An on-chip read/write FIFO, built from the on-chip synchronous dual-port RAM (SelectRAM™) available in Spartan devices, supports data transfers in excess of 33 MHz.

Table 1. 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
Base Address Register 4 (BAR5)				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	Interrupt Pin	Interrupt Line	3Ch
Reserved				40h-FFh

Note:
Italicized address areas are not implemented in the LogiCORE PCI Interface default configuration. These locations return zero during configuration read accesses.

Core Configuration

The LogiCORE PCI Interface can easily be configured to fit unique system requirements using Xilinx web-based PCI CORE Generator. The following customization is supported by the LogiCORE product and described in accompanying documentation.

- Initiator or target functionality (PCI Master only)
- Base Address Register configuration (1 - 3 Registers, size and mode)
- Configuration Space Header ROM
- Initiator and target state machine (e.g., termination conditions, transaction types and request/transaction arbitration)
- Burst functionality
- User Application including FIFO (back-end design)

Table 2. PCI Bus Commands

CBE [3:0]	Command	PCI Master	PCI Slave
0000	Interrupt Acknowledge	No[1]	Ignore
0001	Special Cycle	No[1]	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[1]	Ignore
1110	Memory Read Line	Yes	Yes
1111	Memory Write and Invalidate	No[1]	Yes

Note:
1. The Initiator can present these commands, however, they either require additional user-application logic to support them or have not been thoroughly tested.

Supported PCI Commands

Table 2 illustrates the PCI bus commands supported by the LogiCORE PCI Interface. The *LogiCORE™ PCI Interface Protocol Checklist (2.1)* has more details on supported and unsupported commands (see the *PCI Data Book v2.0*)

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. A FIFO to support PCI burst transfer can efficiently be implemented using the Spartan on-chip RAM feature, SelectRAM™. Each Spartan CLB supports two 16x1 RAM blocks. This corresponds to 32 bits of single-ported RAM or 16 bits of dual-ported RAM, with simultaneous read/write capability.

Wait States

In this version of the core, only one wait state is supported. The XCS30-4 and XC40-4 are able to run with one wait state. For zero wait state applications, use the LogiCORE PCI v2.0, implemented in XC4000XLT devices.

With one wait state, the LogiCORE Interface automatically inserts a wait state when sourcing data (Initiator Write, Target Read) during a burst transfer. In this mode, the LogiCORE Interface can accept data at 100% burst transfer rate and supply data at 50%. See Table 3 for a PCI bus transfer rates for various operations.

Table 3. LogiCORE PCI Spartan Transfer Rates

Operation	Transfer Rate
Initiator Write (PCI ← LogiCORE)	3-2-2-2
Initiator Read (PCI → LogiCORE)	4-1-1-2
Target Write (PC I→ LogiCORE)	5-1-1-1
Target Read (PCI ← LogiCORE)	6-2-2-2

Note:
Initiator Read and Target Write operations have effectively the same bandwidth for burst transfer.

Timing Specification

The XCS family, together with the LogiCORE PCI products enables design of fully compliant PCI systems. Backend design can affect the maximum speed your design is capable of. Factors in your back-end designs that can affect timing include loading of hot signals coming directly from the PCI bus, gate count and floor planning. Table 4 shows the key timing parameters for the LogiCORE PCI Interfaces that must be met for full PCI compliance.

Verification Methods

Xilinx has developed an internal testbench with over 50,000 vectors to test the Xilinx PCI design. The LogiCORE PCI Interfaces have also been extensively simulated using the VirtualChips VHDL PCI test bench from Phoenix Technologies, Ltd. (not included with the LogiCORE™ PCI products). The Interface has also been verified in hardware in the XSC30-4 PQ208C FPGA.

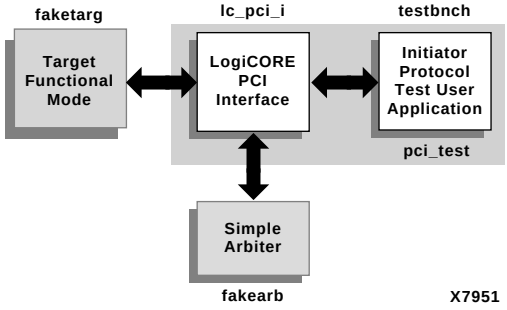
Included with the LogiCORE™ PCI Master and Slave Interface is an example design and a VIEW*logic* based PCI protocol test bench that verifies the PCI interface functions according to the test scenarios specified in the *PCI Local Bus Specification* see Figure 2. This test bench consists of 28 test scenarios, each designed to test compliance to a specific PCI bus operation. Refer to the *LogiCORE PCI Interface Protocol Checklist* for a complete list of supported test scenarios (see the *PCI Spartan Data Book*).

Table 4. Advanced Timing Parameters [ns]

Parameter	Ref.	PCI Spec.		LogiCORE PCI, XCS-4	
		Min	Max	Min	Max
CLK Cycle Time		30	∞	30[1]	∞
CLK High Time		11		11	
CLK Low Time		11		11	
CLK to Bus Signals Valid[3]	T_{ICKOF}	2	11	2[2]	9.6
CLK to REQ# and GNT# Valid[3]	T_{ICKOF}	2	12	2[2]	9.6
Tri-state to Active		2		2[2]	
CLK to Tri-state			28		28[1]
Bus Signal Setup to CLK (IOB)	T_{PSU}		7		7
Bus Signal Setup to CLK (CLB)			7		7[1]
GNT# Setup to CLK	T_{PSU}		10		5.2
Input Hold Time After CLK (IOB)	T_{PH}		0		0
Input Hold Time After CLK (CLB)			0		0[2]
RST# to Tri-state			40		40[2]

Notes:
1. Controlled by TIMESPECs, included in product
2. Verified by analysis and bench-testing
3. IOB configured for Fast slew rate

Figure 2. PCI Protocol Testbench



Recommended Design Experience

The LogiCORE PCI interface is pre-implemented allowing engineering focus at the unique back-end functions of a PCI design. Regardless, PCI is a high-performance system that is challenging to implement in any technology, ASIC or FPGA. Therefore, we recommend previous experience with building high-performance, pipelined FPGA designs using Xilinx implementation software, TIMESPECs, and guide files. The challenge to implement a complete PCI design including back-end functions varies depending on configuration and functionality of your application. Contact your local Xilinx representative for a closer review and estimation for your specific requirements.

Table 5. Part Numbers

Product	Part Number
LogiCORE™ PCI Spartan Master (Initiator/Target)	
LogiCORE™ PCI Spartan Slave (Target Only)	
LogiCORE™ PCI Spartan Slave to Master Upgrade	
LogiCORE™ PCI Spartan Master (Initiator/Target) Software Renewal	
LogiCORE™ PCI Spartan Slave (Target Only) Software Renewal	

Ordering Information

Table 5 shows the part numbers for the LogiCORE™ products. Before placing an order, please read and sign the attached LogiCORE™ license agreement and fax it to Xilinx at +1 408-377-3259. For pricing and availability please contact your local Xilinx sales office.

Related Information

Recommended Design Centers

Listed below are design centers and design consultants that have experience with the LogiCORE PCI products.

HighGate Design
12380 Saratoga-Sunnyvale Road, Suite 8
Saratoga, CA 95070-3090, USA
Phone: +1 408-255-7160
Fax: +1 408-255-7162
E-mail: info@highgatedesign.com
URL: www.highgatedesign.com

Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, Arizona 85202, USA
Phone: +1 602-491-4311
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Comit Systems
1250 Oakmead Pkwy, Suite 210
Sunnyvale, CA 94088, USA
Phone: +1 408-988-2987
Fax: +1 408-988-2133
E-mail: preeth@comit.com
URL: www.comit.com

Austin Franklin
Dark Room Technologies, Inc.
126 Poor Farm Road
Harvard MA, 01451
Phone: +1 508-772-9928
Fax: +1 508-
E-mail: darkroom@ix.netcom.com

PCI Special Interest Group (PCI-SIG) Publications

The PCI-SIG publishes various PCI specifications and related documents such as *PCI Local Bus Specification*, *PCI Compliance Checklist* and *PCI System Design Guide*.

PCI Special Interest Group
2575 NE Kathryn St. #17
Hillsboro, OR 97124
Phone: +1 800-433-5177 (inside the US)
+1 503-693-6232 (outside the US)
Fax: +1 503-693-8344
Office hours: 8:30am - 4:00pm PST
E-mail: info@pcisig.com
URL: www.pcisig.com

Xilinx Documents

More PCI related information is available on Xilinx Web:

www.xilinx.com/products/logicore/logicore.htm.

LogiCORE™

Synthesizable PCI Bridge Design Example

November 11, 1997 | Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-377-3259
E-mail: Techsupport: hotline@xilinx.com
Feedback: logicore@xilinx.com
URL: http://www.xilinx.com

Introduction

This synthesizable PCI bridge design is an example application bridge for use with the LogiCORE V2.0 PCI core. It is delivered in Verilog and VHDL. This example demonstrates how to interface to the PCI core and provides a modular foundation upon which to base other designs.

Features

- Initiator Functions
  - Separate read and write FIFOs (unidirectional)
  - Simple block data transfer engine (DMA)
  - Burst sizes fixed by a transfer counter
  - Auto data delivery (handles terminations)
  - Discard counter to prevent deadlock
  - Initiator address counter
- Target Functions
  - First base address register in MEM space, subdivided into four regions
    - Supports single data phase transfers only
    - First region demonstrates doorbells
    - Second region demonstrates mailboxes
    - Third region demonstrates single transfer, long latency accesses
    - Fourth region contains control registers for initiator data transfer engine and other functions (in target only designs, initiator control registers are non-functional)
  - Second base address register in MEM space
    - Separate read and write FIFOs (unidirectional) with posted writes and prefetch reads
    - Delayed completion discard after time-out
    - Generates target abort on address wrap
  - Target address counter
    - Target functions independent of initiator

LogiCORE™ Facts

Design Specifics		
Device Family	XC4000XLT	
CLBs Used	Up to 800[1]	
IOBs Used	Up to 320[2]	
System Clock f_{max}	0 – 33MHz	
Device Features Used	Bi-directional data buses SelectRAM™	
Supported Devices / Resources Available		
	I/O[3]	**CLB[4]**
XC4013XLT PQ208	99/101	268 - 398
XC4013XLT HQ240	133/135	268 - 398
XC4028XLT HQ240	133/135	716 - 846
XC4062XLT HQ240	133/135	1996 - 2126
XC4062XLT BG432	293/295	1996 - 2126
Provided with Design		
Documentation	User's Guide	
Design File Formats	Verilog Source Code VHDL Source Code	
Constraint Files	Supplemental .UCF and .PCF files for use with LogiCORE V2.0 Sample synthesis script	
Verification Tool	Verilog Testbench VHDL Testbench	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
LogiCORE Product	PCI Master or Slave V2.0.0	
Implementation / Verification Tools	Verilog or VHDL simulation and synthesis tools supported by the LogiCORE V2.0 design flow.	

Notes:
1. Full design, including PCI interface, is under 800 CLBs. Actual count depends on the implemented feature set.
2. Full design, including PCI interface, is under 320 IOBs assuming all signals are routed off chip. The actual count depends on the implemented feature set.
3. Target / Master.
4. The exact number of CLBs depends on user configuration of the core and level of resource sharing with adjacent logic.

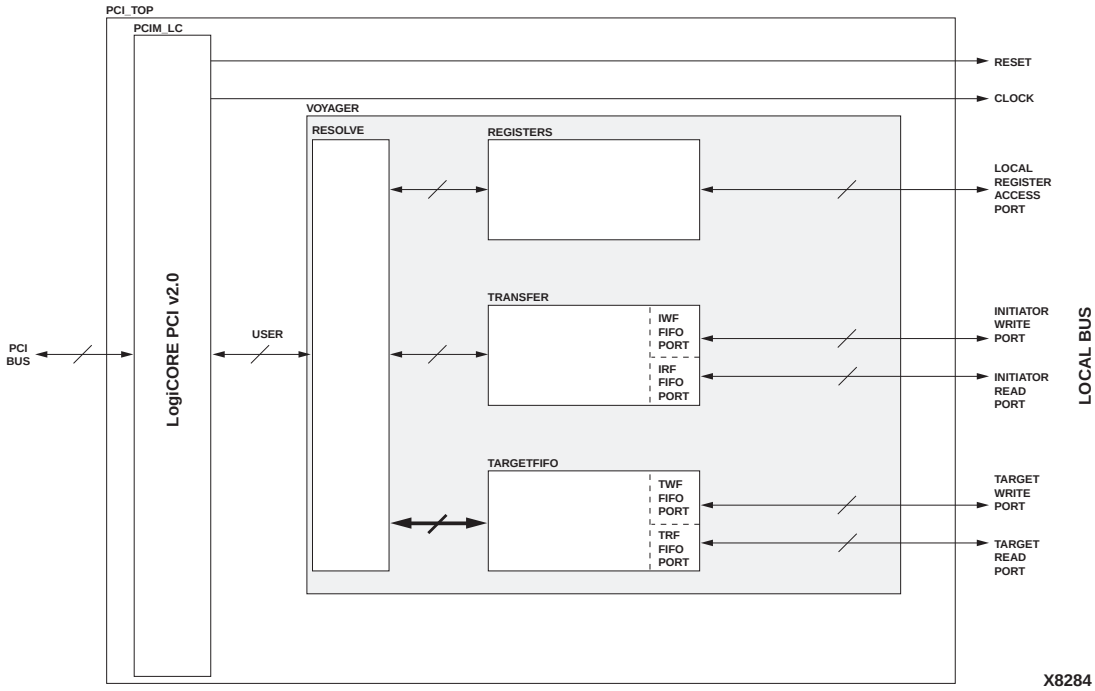


Figure 1: LogiCORE PCI 2.0 Block Diagram

General Description

This design is a general purpose data transfer engine to be used with the LogiCORE V2.0 PCI core. Figure 1 presents a block diagram of the design. Typically, the user will customize the local interface to conform to a particular peripheral bus (ISA, VME, i960). The design is modular so that unused portions may be removed.

Functional Description

This design example supports target functionality in two memory spaces. Initiator functionality is controlled by writing into registers. The local bus interface signals are distinct for each block in the design, allowing blocks to be added or removed. Data transfer is pipelined for high clock rate.

BAR0 Configuration

BAR0 is configured as a 4 kilobyte MEM space which maps to a number of registers. This space does not support multiple data phase transfers. All accesses to this space terminate with target disconnect with data.

This space is logically divided into four regions based on functionality. The four regions, and the functions of the registers, are discussed below.

Region One: Doorbells

Register DBELL_P1 is a PCI-to-local doorbell. A PCI agent may create an interrupt on the local side by setting any bit of the register. A PCI agent is permitted to read back the status of this register with no side-effects.

When the local side services the interrupt, it reads this register to determine the cause of the interrupt, then clears the interrupt by writing a one to that bit. The local side may read this register without side-effects.

Similarly, DBELL_L1 is a local-to-PCI doorbell. To prevent spurious interrupts, an interrupt may not be cleared by the agent that requested it. The recipient of the interrupt must clear the interrupt. To enforce this, doorbell register bits may not be cleared from the requesting side. Before doorbell interrupts may occur, the doorbell interrupt enable bits in the CONTROL register must be set.

Region Two: Mailboxes

Register MBOX_P1 is a PCI-to-local mailbox. A PCI agent may deliver mail to an empty mailbox for a local agent to pick up. When a PCI agent writes to this register, the data is registered and a “full” flag is set. Subsequent writes to a full mailbox have no effect. The PCI agent may not read back delivered mail. Reads of the mailbox from the PCI bus

side return the state of the full flag (replicated in all bits).

When the local side reads the mailbox, the “full” flag is cleared. Subsequent reads of an empty mailbox return the last valid data present in the mailbox.

Similarly, MBOX_L1 is a local-to-PCI mailbox. The “full” flag may be monitored in two ways. Mailbox “full” flags are always observable in the CONTROL register, so both PCI agents and local agents may poll the CONTROL register to watch for new messages. Optionally, full mailboxes may create interrupts. Interrupts are created on the recipient’s side, and are cleared by reading the mailbox. Before mailbox interrupts may occur, the mailbox interrupt enable bits in the CONTROL register must be set.

Region Three: Bounded Latency Accesses

The two registers in this region are used for demonstrating bounded latency non-burst accesses. This type of access may be used in situations where the user application has a short latency with a known upper bound of 16 PCI clocks from the time the initiator asserts FRAME#. This is done by inserting wait states until the target is capable of completing the transaction.

Register BL_CTRL controls the initial latency of read and write operations for itself and BL_DATA. Only the least-significant four bits of the register are implemented, and the register is only accessible from the PCI bus. The local side has no access to this register, so local reads will return all zeroes and writes have no effect.

The second register, BL_DATA, is a general purpose, read/write register that responds according to the settings in BL_CTRL. This data register is only accessible from the PCI bus. The local side has no access to this register, so local reads will return all zeroes and writes have no effect.

Region Four: Control Registers

The first three registers in this region control the initiator transfer engine.

Register XFER_LEN is used to indicate the length of the data block to be transferred. The low half of the register is not implemented. The high half is implemented as a loadable 16-bit counter.

This register is accessible from both the PCI bus and local bus, and may be read at any time. Status of transfers in progress may be obtained by reading this register. With each successful transfer, the counter decrements.

Register XFER_PADR contains the current PCI bus address for transfers performed by the transfer engine. Depending on the direction of the transfer, this address may be a source or destination.

This register is accessible from both the PCI bus and local bus, and may be read at any time. Status of transfers in progress may be obtained by reading this register. With each successful transfer, the address increments.

Register XFER_LADR contains the current local address for transfers performed by the transfer engine. Depending on the direction of the transfer, this address may be a source or destination. Only the low half of this register is implemented.

This register is accessible from both the PCI bus and local bus, and may be read at any time. Status of transfers in progress may be obtained by reading this register. With each successful transfer, the address increments.

These registers must not be written to while the initiator is active. To ensure this does not occur, writes to these register are disabled while the initiator is active.

BAR1 Configuration

BAR1 is configured as a 64 kilobyte MEM space which maps to the target FIFOs. This space supports multiple data phase transfers. Transfers beyond the end of the address space result in target abort. For all other accesses, this region will respond according to how it is accessed. Consult the PCI specification regarding delayed transactions. Data transfer between the local side and PCI bus is achieved using retries and delayed transactions as needed.

Posted Writes

The target performs posted writes. On writes to an idle target, the FIFO accepts incoming data until it is full or the write transaction has ended, whichever occurs first. In the event of a full FIFO, the target issues a disconnect. After the PCI transaction is complete, the target empties the FIFO by writing the data out to the local side until the FIFO is empty. To achieve this, the target latches the destination address for use during write out.

On writes to a busy target (the FIFO is still busy from the previous transaction) the target responds with retry, without putting the request in a retry queue.

Prefetched Reads

For reads, the target may not anticipate the length of the transaction or have the data available in time. For this reason, the target puts the transaction in a retry queue and responds with a retry termination. Then the target prefetches data to fill the FIFO. When the initiator returns to retry the transaction, the data will be available.

If the initiator returns to retry the transaction, and does not completely empty the FIFO, the FIFO is flushed after the transaction is complete. If the initiator does empty the FIFO, and attempts to read more, the target issues a disconnect.

If the initiator never retries the original transaction, deadlock may occur. For this reason, there exists a discard timer that signals a waiting delayed completion should be discarded. This timer times out after 32,768 PCI clocks. This period may be shortened to allow simulation of this event in a reasonable amount of time.

Register File Interface

The operation of this block is synchronous to the PCI clock. This block contains all the control and status registers discussed in the functional description. The local bus access port is defined in Table 1.

Table 1: Local Bus Register Interface

Name	Direction	Function
LWE	Input	Write enable for registers
LRE	Input	Read enable for registers
LADDR	Input	Address input
LDIN	Input	Data input
LDOUT	Output	Data output
LINT_N	Output	Active low local interrupt

Target FIFO Interface

The operation of this block is synchronous to the PCI clock. This block is interfaced to two FIFOs; one is the target read (TRF) FIFO, and the other is the target write (TWF) FIFO.

The FIFOs are identical, but data flows in opposite directions. Table 2 lists the signals used in the interface.

Table 2: Local Bus Target FIFO Interface

Name	Direction	Function
TRF_LD TWF_ST	Output Output	Data requested or available
TRF_ADDR TWF_ADDR	Output Output	Transfer starting address
TRF_AF, TWF_AE	Output	Transfer almost done flag
TRF_WR, TWF_RD	Input	FIFO write and read enable
TRF_DIN TWF_DOUT	Input Output	Data transfer ports

Initiator FIFO Interface

The operation of this block is controlled by the contents of registers in the register block. This block is interfaced to two FIFOs, similar to the memory interface block. One is the initiator read (IRF) FIFO, and the other is the initiator write (IWF) FIFO.

The FIFOs are identical, but data flows in opposite directions. A description of a FIFO follows in the FIFO section. Table 3 lists the signals used in the interface.

Table 3: Local Bus Initiator FIFO Interface

Name	Direction	Function
IWF_LD IRF_ST	Output Output	Data requested or available
IF_ADDR	Output	Transfer starting address

Table 3: Local Bus Initiator FIFO Interface

Name	Direction	Function
IWF_AF, IRF_AE	Output	Transfer almost done flag
IWF_WR, IRF_RD	Input	FIFO write and read enable
IWF_DIN IRF_DOUT	Input Output	Data transfer ports

Pinout

The register file and FIFO interface pinouts are not fixed to specific FPGA I/O pads, allowing flexibility in customization. The PCI bus specific signals are constrained as part of the LogiCORE PCI implementation.

As shipped, all of the register file and FIFO interface signals are brought off-chip, but it is not necessary that any interface signals be brought off chip at all in single FPGA designs.

Core Modifications

Modifications can be done to remove the initiator functionality or selected portions of the target functionality. The full design may be expanded as needed or reduced to a very small subset of the original design. The PCI interface itself is also configurable by the designer.

Verification Methods

This design example includes a system level testbench that simulates a four-slot PCI system. This simulation testbench includes a behavioral host bridge (with programmable arbiter) capable of generating burst transactions and a programmable behavioral target.

Recommended Design Experience

The challenge to implement a complete PCI design varies depending on configuration and functionality of your application. We recommend previous experience with building high-performance, pipelined FPGA designs using the Xilinx implementation software and familiarity with either VHDL or Verilog.

Ordering Information

This design is available at no cost to all registered LogiCORE V2.0 PCI core customers. It may be downloaded from the Xilinx home page. Simply log on to the CORE Solutions section of WebLINX (http://www.xilinx.com/products/logicore/logicore.htm) and select the appropriate link.

LogiCORE™

One Dimensional ROM-Based Correlator

February 8, 1998 | Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Data can be loaded in either serial or parallel (broadside) fashion
- Supports data words from 4 to 128 bits in steps of 4
- Mask register that selects which bits within the data pattern to compare
- Cascadable to any length
- Easily extendible to 2 dimensions
- High performance implementation utilizing efficient look-up table (LUT) design
- Requires relatively few CLBs
- Single level of logic (1 LUT) per pipeline stage
- Drop-in modules for the XC4000E, EX, and XL families
- Density and performance guaranteed through Relationally Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This parameterized core accepts data in a serial or parallel fashion and looks for a predetermined bit pattern that is stored in ROM look-up tables. The output is a binary representation of the number of bits that match the desired bit pattern after a bit mask has been applied to remove all of the “don’t care” bits.

The match register, the mask register, and all of the logic to implement the compare is implemented in look-up table logic so the function can be realized as a series of small LUTs (one for each 4 bits of data) followed by a small adder tree.

Latency is equal to one input buffer plus one pipeline register for each level of the 4-bit adder tree. For example, if the match register is 24 bits, the adder tree is 3 levels deep with a 5-bit wide output. The total latency is two (registers) + three (adder trees) = five.

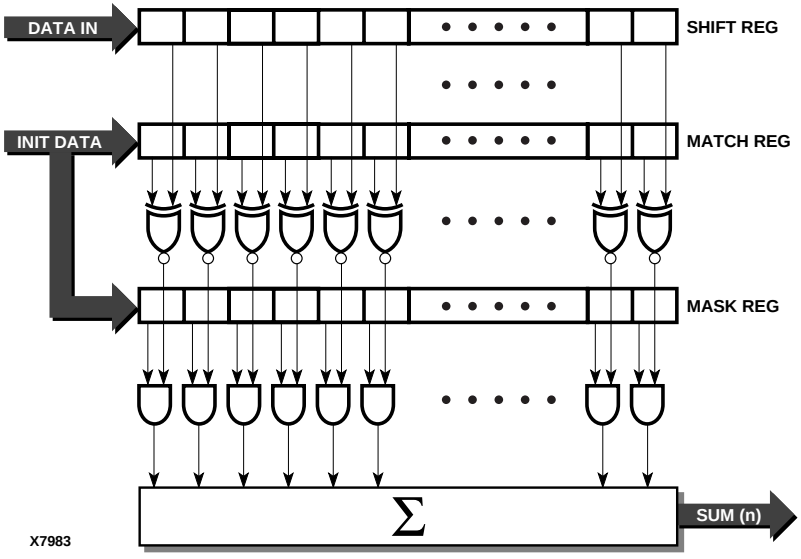


Figure 1: Serial In Correlator Block Diagram

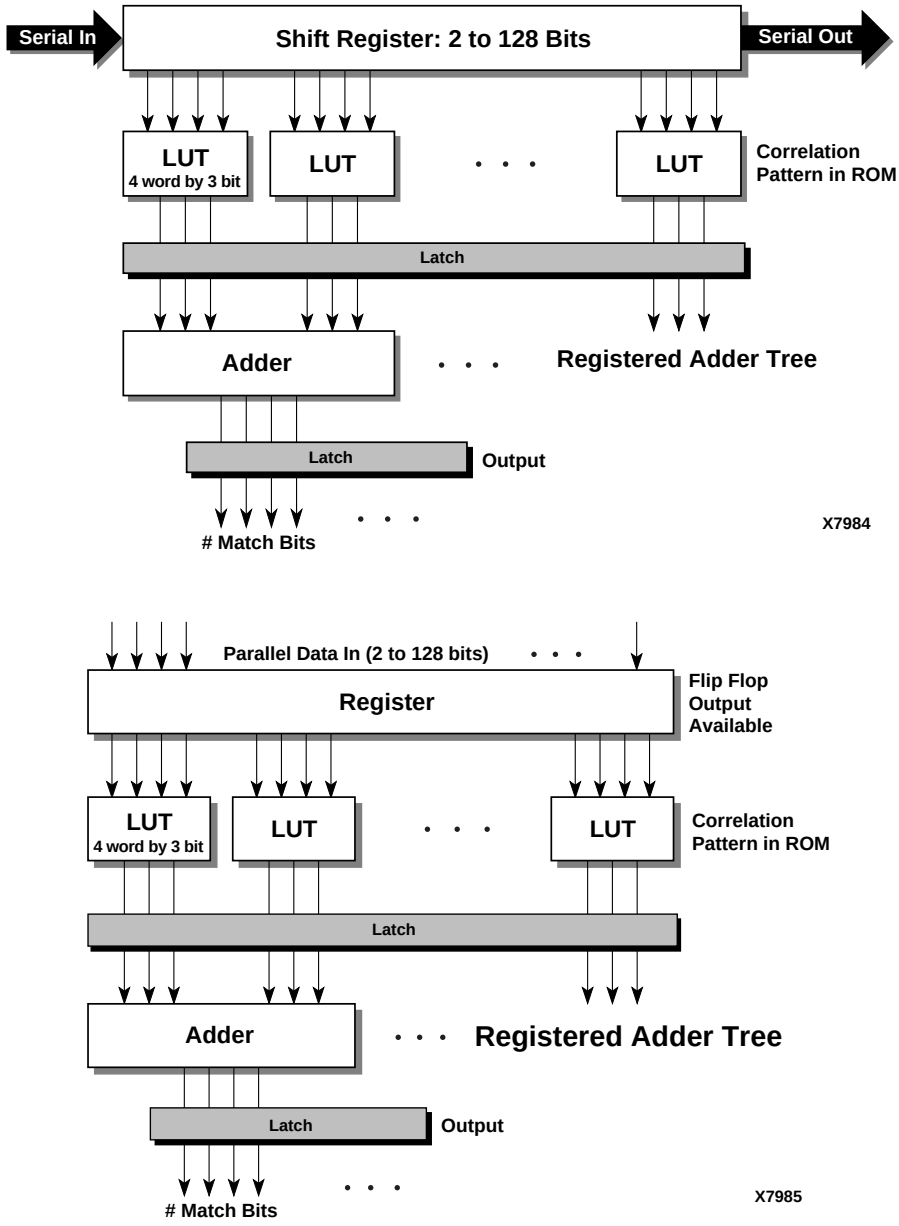


Figure 2: CORE Generator Implementation for Serial Data (top) and Parallel Data (bottom)

This function is used in data communications to establish synchronization in a serial bit stream, and in any application that requires pattern recognition.

Using Look-Up Tables for Implementing Correlators

The following example illustrates what the CORE Generator calculates to produce the contents of the distributed look-up tables. For a 4-bit correlator with an input search pattern of 1101 and a mask pattern of 1111 (include all bits), the LUT contains the following data.

Any n-stage correlator can be decomposed into (n/4) 4-stage correlators. The LUTs contain all potential outputs for each 4-stage correlation. For example, the correlation pattern 1101 stores 4 at address D in the LUT (all four bits match) and 3 at addresses 5, 9, C, and F in the LUT (three bits match).

Table 1: LUT Contents

Address	Data
0000	001 (1 bit matches)
0001	010 (2 bits match)
0010	000 (0 bits match)
0011	001 (1 bit matches)
0100	010 (2 bits match)
0101	011 (3 bits match)
0110	001 (1 bit matches)
0111	010 (2 bits match)
1000	010 (2 bits match)
1001	011 (3 bits match)
1010	001 (1 bit matches)
1011	010 (2 bits match)
1100	011 (3 bits match)
1101	100 (4 bits match)
1110	010 (2 bits match)
1111	011 (3 bits match)

Combining 4-Input Sections

Each four-input correlator section can be combined by summing the outputs with an adder tree. The look-up tables are 16 words (4 address lines) by 3 bits wide each. The adder tree grows by one bit for each level and the resulting output is a binary number representing the number of matches in the input data word after the mask register has been applied.

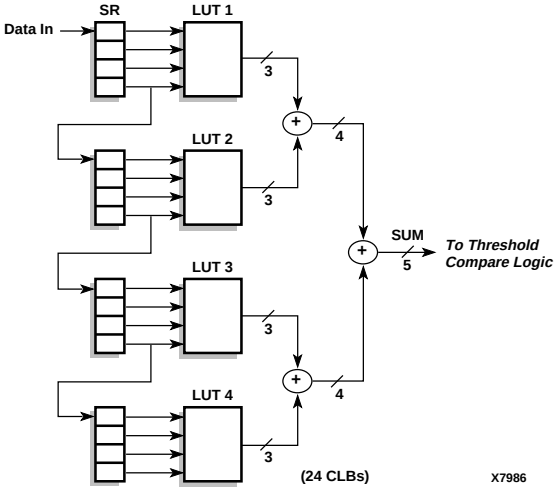


Figure 3: Combining 4-Input Sections

Pinout

Signal names for the schematic symbols are shown in Figures 4 and 5, and described in Tables 2 and 3.

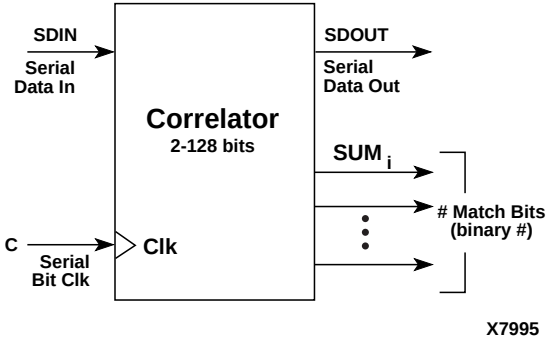


Figure 4: Core Schematic Symbol – Serial Data In

Table 2: Core Signal Pinout – Serial Data In

Signal	Signal Direction	Description
SDIN	Input	Serial Data In – Serial data stream for correlation.
C	Input	Serial Data Clock
SDOUT	Output	Serial Data Out – SDIN delayed N clocks.
SUM_i	Output	Binary representation of the number of bits that match the correlation pattern excluding any bits defined in the mask register.

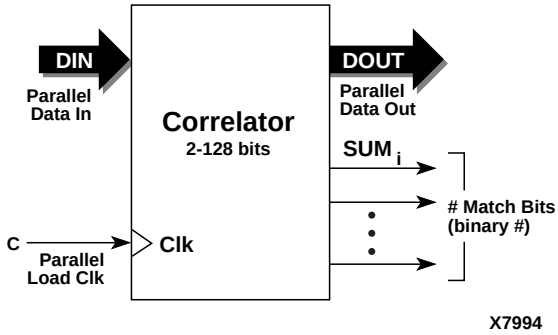


Figure 5: Core Schematic Symbol – Parallel Data In

Table 3: Core Signal Pinout – Parallel Data In

Signal	Signal Direction	Description
DIN	Input	Parallel Data In – 2 to 128 bits
C	Input	Clock – Parallel load clock
DOUT	Output	Parallel Data Out – DIN delayed 1 clock.
SUM_i	Output	Binary representation of the number of bits that match the correlation pattern excluding any bits defined in the mask register.

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

CORE Generator Parameters

This macro has two CORE Generator dialog boxes. The one for the parallel correlator is shown in Figure 6. The serial correlator dialog box has a different title, but the parameters are the same. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Data Width**: Select an input width from the pull-down menu for the data string to be compared. The valid range is 4-128 in multiples of 4.
- **Match:** Hexadecimal string specifying the pattern to look for.
- **Mask:** A zero in this hexadecimal string specifies that the data bit in the same position should be ignored when calculating the number of match bits.
- **Radix**: Hexadecimal or Binary representation of the Match and Mask values.
- **Read From File**: Read the Match and Mask values from a text file.

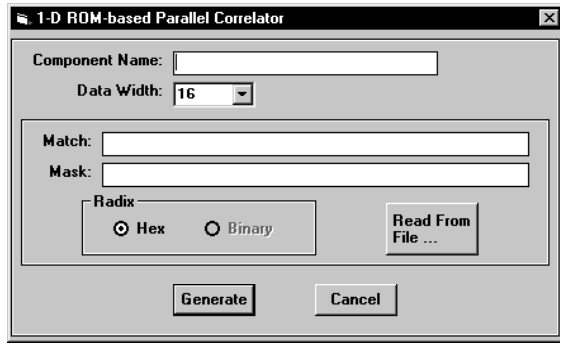


Figure 6: CORE Generator Dialog Box

Bit Width and CLB Count

Table 4 lists the number of CLBs required for example bit widths. The maximum speed is for XC4000E-1 devices.

Table 4: Bit Width versus CLB Count

Bit Width	CLB Count
4	4
7	11
8	14
9	16
10	17
11	19
12	20
13	22
14	24
15	26
16	25
17	29
20	36
32	54

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

One Dimensional RAM-Based Correlator

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Data can be loaded in either serial or parallel (broadside) fashion
- Supports data words from 4 to 128 bits in steps of 4
- Cascadable to any length
- Easily extendible to 2 dimensions
- High performance implementation utilizing efficient RAM design
- Requires relatively few CLBs
- Single level of logic per pipeline stage
- Drop-in modules for the XC4000E, EX, and XL families
- Density and performance guaranteed through Relationally Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This parameterized core accepts data in a serial or parallel fashion and looks for a predetermined bit pattern that is stored in RAM. The output is a binary representation of the number of bits that match the desired bit pattern.

The match register, the mask register, and all of the logic to implement the compare is implemented in look-up table (LUT) logic so the function can be realized as a series of small LUTs (one for each 4 bits of data) followed by a small adder tree.

Unlike the ROM-based correlator, the match pattern is not specified in the GUI. The RAMs must be explicity loaded before the correlator can be used. The WE signal controls when data is written into the RAM: when WE is high, the DIN input (for parallel) or the last **W** input bits of SDIN (for serial, where **W** is the data width) act as the address to the RAM. Each 4 bits of input address a 16x3 RAM. The data to be stored in the RAM must come in at the LDATA bus. The width of LDATA is 3/4 the width of the correlator -- 4 bit slices of DIN are used as an address to store the corresponding 3 bit slices of LDATA into individual 16x3 RAMs. It takes 16 clock cycles to fully load the RAM after which WE must go low. After that, the correlator functions like a ROM correlator. The RAM can be reloaded with a new pattern at any time by making WE high and going through 16 clock cycles with the appropriate inputs at DIN and LDATA.

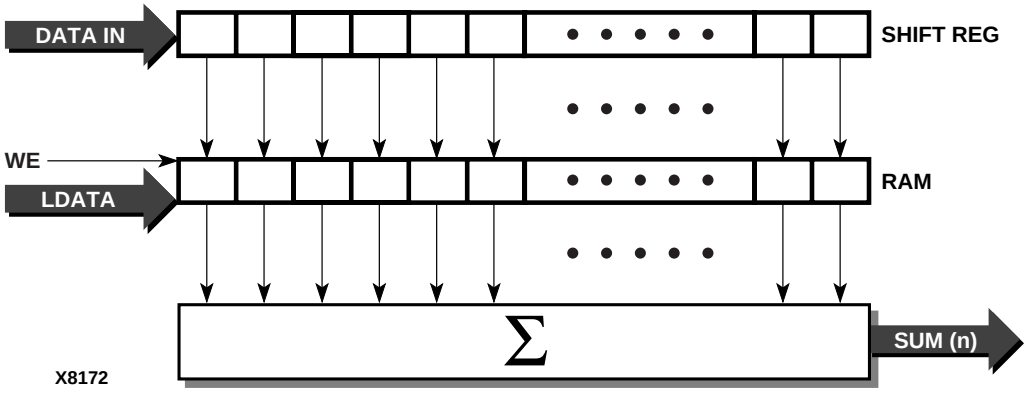


Figure 1: Serial In Correlator Block Diagram

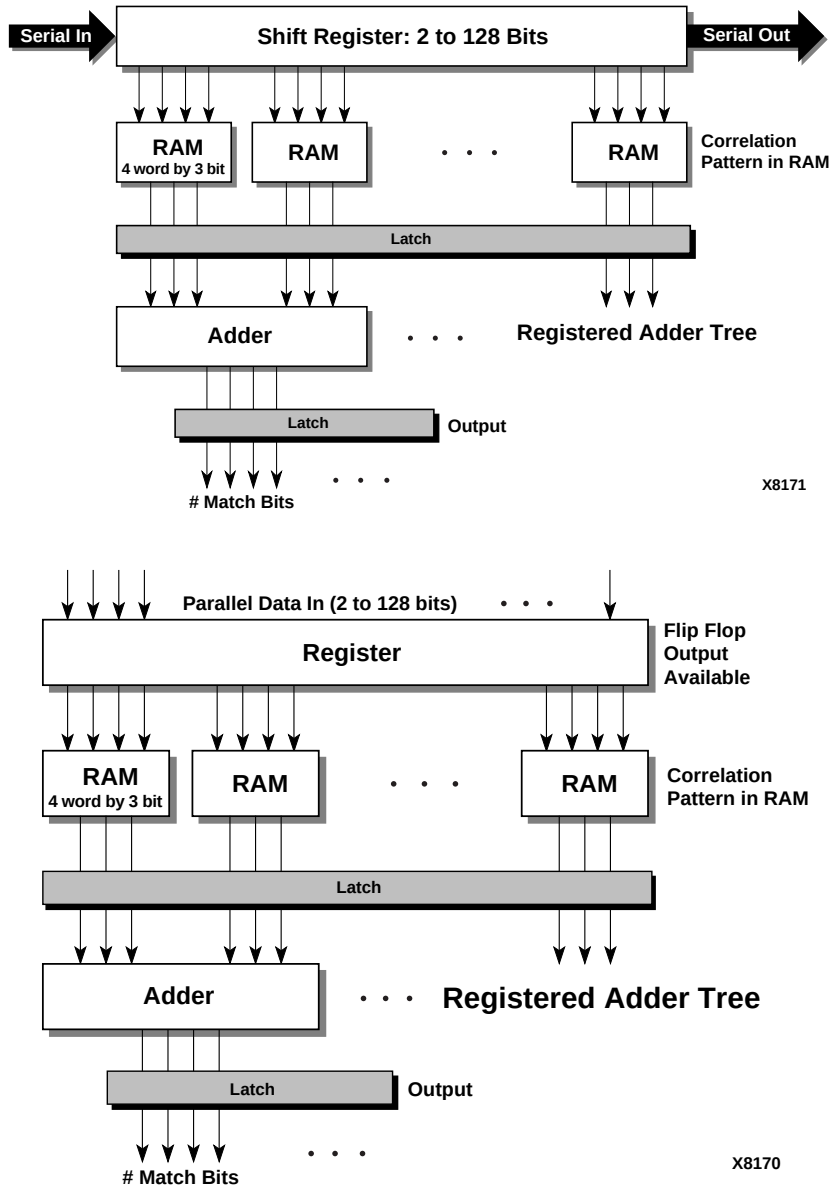


Figure 2: CORE Generator Implementation for Serial Data (top) and Parallel Data (bottom)

Latency is equal to one input buffer plus one pipeline register for each level of the 4-bit adder tree. For example, if the match register is 24 bits, the adder tree is 3 levels deep with a 5-bit wide output. The total latency is two (registers) + three (adder trees) = five.

This function is used in data communications to establish synchronization in a serial bit stream, and in any application that requires pattern recognition.

Using RAM for Implementing Correlators

Table 1 illustrates what the user must calculate to produce the contents of the distributed RAM. For a 4-bit correlator with an input search pattern of 1101 and a mask pattern of 1111 (include all bits), the RAM contains the following data.

Any n-stage correlator can be decomposed into (n/4) 4-stage correlators. The RAMs contain all potential outputs for each 4-stage correlation. For example, the correlation pattern 1101 stores 4 at address D in the RAM (all four bits match) and 3 at addresses 5, 9, C, and F in the RAM (three bits match).

The user must compute the contents of the RAMs for the desired “match” pattern and load the RAMs before doing any comparisons.

Table 1: RAM Contents

Address	Data
0000	001 (1 bit matches)
0001	010 (2 bits match)
0010	000 (0 bits match)
0011	001 (1 bit matches)
0100	010 (2 bits match)
0101	011 (3 bits match)
0110	001 (1 bit matches)
0111	010 (2 bits match)
1000	010 (2 bits match)
1001	011 (3 bits match)
1010	001 (1 bit matches)
1011	010 (2 bits match)
1100	011 (3 bits match)
1101	100 (4 bits match)
1110	010 (2 bits match)
1111	011 (3 bits match)

Combining 4-Input Sections

Each four-input correlator section can be combined by summing the outputs with an adder tree. The RAMs are 16 words (4 address lines) by 3 bits wide each. The adder tree grows by one bit for each level and the resulting output is a binary number representing the number of matches in the input data word after the mask register has been applied.

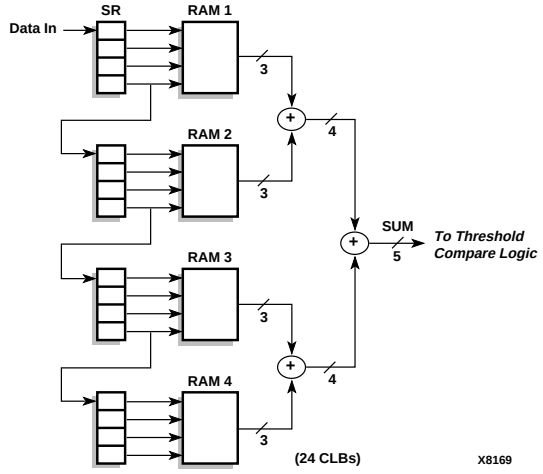


Figure 3: Combining 4-Input Sections

Pinout

Signal names for the schematic symbols are shown in Figures 4 and 5, and described in Tables 2 and 3.

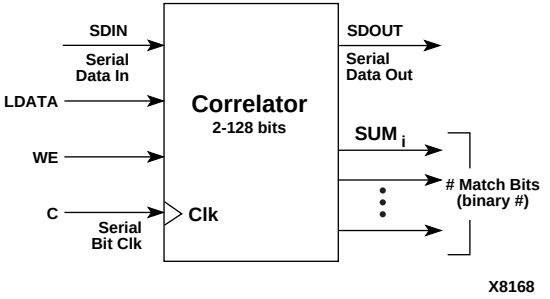


Figure 4: Core Schematic Symbol – Serial Data In

Table 2: Core Signal Pinout – Serial Data In

Signal	Signal Direction	Description
SDIN	Input	Serial Data In – Serial data stream for correlation.
C	Input	Serial Data Clock
WE	Input	Write Enable – to load the RAM.
LDATA	Input	The data that gets written into all the RAMs at the address locations in the input shift register.

Table 2: Core Signal Pinout – Serial Data In (cont.)

Signal	Signal Direction	Description
SDOUT	Output	Serial Data Out – SDIN delayed N clocks.
SUM_i	Output	Binary representation of the number of bits that match the correlation pattern excluding any bits defined in the mask register.

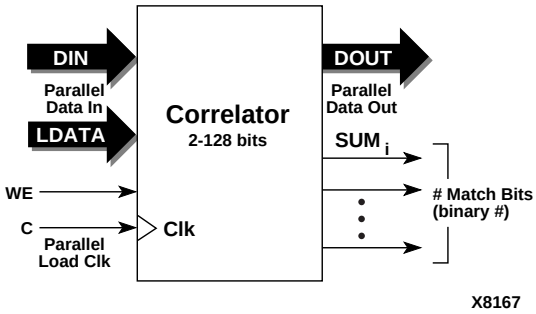


Figure 5: Core Schematic Symbol – Parallel Data In

Table 3: Core Signal Pinout – Parallel Data In

Signal	Signal Direction	Description
DIN	Input	Parallel Data In – 2 to 128 bits
C	Input	Clock – Parallel load clock
WE	Input	Write Enable – to load the RAM.
LDATA	Input	The data that gets written into all the RAMs at the address locations in the input shift register.
DOUT	Output	Parallel Data Out – DIN delayed 1 clock.
SUM_i	Output	Binary representation of the number of bits that match the correlation pattern excluding any bits defined in the mask register.

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

CORE Generator Parameters

This macro has two CORE Generator dialog boxes. The one for the parallel correlator is shown in Figure 6. The serial correlator dialog box has a different title, but the parameters are the same. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Data Width**: Select an input width from the pull-down menu for the data string to be compared. The valid range is 4-128 in multiples of 4.

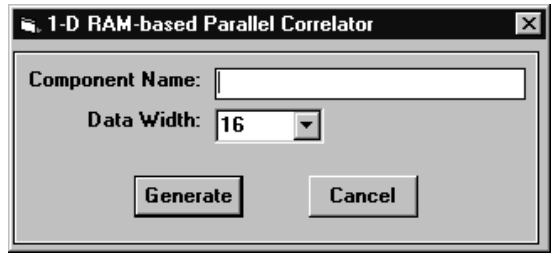


Figure 6: CORE Generator Dialog Box

Bit Width and CLB Count

Table 4 lists the number of CLBs required for example bit widths. The maximum speed is for XC4000E-1 devices.

Table 4: Bit Width versus CLB Count

Bit Width	CLB Count
4	4
8	14
12	20
16	25
20	36
32	54

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

LogiCORE™ Comb Filter

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Multiplier-free filter yields efficient implementation
- All zeros, no poles
- Input data widths from 2 to 32 bits
- Variable length feed-forward path selection (configurable delay) from 1 to 17 samples
- Registered outputs
- Sections can be cascaded
- 2's complement input data
- Uses fast carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Can be combined with an Integrator core to build integrating and decimating filters as described by Hogenauer
- Available in Xilinx CORE Generator

Functional Description

The Comb filter provides a standard differentiator with variable length feed-forward delays from 1 to 17 incoming sample periods. The subtractor takes the feed-forward delayed input data and subtracts it from the data input to give a registered output that is one bit wider than the input.

Multiple Comb Filter stages can be cascaded by connecting the output of the first stage to the input of the second stage and parameterizing the proper bit widths to allow for bit growth through the multiple adder stages. The data output from the last stage can be truncated (the least significant bits not connected) to carry the desired amount of precision to the next signal processing block.

Latency for this block is equal to the number of feed-forward delays plus one for the output register.

The Comb filter transfer function, $H_c(z) = 1\text{-}z^{-M}$, has M roots which map as zeros (nulls) on the unit circle of the complex Z plane. The corresponding frequency response exhibits nulls at $f_k = (f_s/M)k$, where:

f_s = sample frequency
M = feed-forward path delay samples
k = 0, 1, 2, 3, …, M

The amplitude-frequency response resembles the teeth of a comb, and the cascading of Comb filters produces sharper teeth. Because a one-bit word growth may occur in each filter stage, the succeeding stages must grow in word length, or a scaling and truncation regimen is required to maintain constant word size in all stages.

This all-zero Comb filter, in tandem with single pole recursive filters, can, through pole-zero cancellation, produce selective frequency responses such as low pass, band pass, etc. This is a viable filter design technique and is known as frequency sampling [1]. An outstanding example of this technique is to cascade several Comb filter stages in series with an equal number of single pole integrators (poles are at f=0). The result is a multiplier-free, high order decimating (or interpolating) filter [2].

1. L. R. Rabiner and B. Gold, *"Theory and Application of Digital Signal Processing."* Prentice-Hall, 1975, pp. 48-50.
2. E. B. Hogenauer, *"An Economical Class of Digital Filters for Decimation and Interpolation."* IEEE Trans. On Acoustics, Speech, and Signal Processing, Vol. ASSP-29, April 1981, pp. 155-162.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

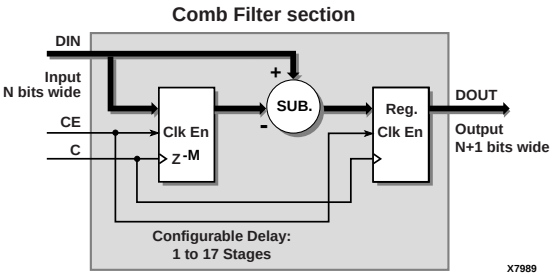


Figure 1: Block Symbol and Schematic Diagram

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
DIN	Input	Serial Data In – for correlation
CE	Input	Clock Enable – active high
C	Input	Clock – data is clocked into the delay element on the rising edge
DOUT	Output	Comb filter output

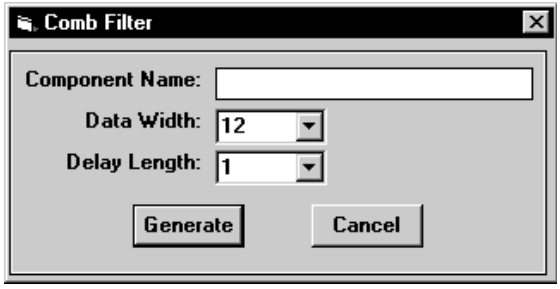
CORE Generator Parameters

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name:** Enter a name for the component.
- **Data Width:** Select an input filter width from the pull-down menu. The valid range is 2-31.
- **Delay Length:** Select the number of delay stages from the pull-down menu. The valid range is 1-17.

Figure 2: CORE Generator Dialog Box



Core Resource Utilization

The following equations show the number of CLBs used in the XC4000 family based on the input data width.

Number of CLBs where N = Input Data Width:

Even N:

CLBs = N + 1 If delay = 1 or 2
CLBs = N + 2 If delay = 3, 4, or 5
CLBs = N + 4 If delay = 6 or more

Odd N:

CLBs = N + 2 If delay = 1 or 2
CLBs = N + 4 If delay = 3, 4, or 5
CLBs = N + 6 If delay = 6 or more

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Serial Distributed Arithmetic FIR Filter

December 22, 1997 | Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Input data words from 4 to 32 bits
- Coefficient width from 4 to 24 bits
- Output data words from 2 to 31 bits
- Taps from 6 to 80 for symmetric filters
- Taps from 6 to 40 for non-symmetric filters
- Support for cascading multiple filters together to create a larger filter
- Support for symmetric, non-symmetric, and negative symmetric filters
- Full precision
- Scaleable output
- Input and output synchronization signals
- Registered output
- 2's complement input data
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

General Description

FIR filters are one of the most basic building blocks used in digital signal processing, taxing the performance that DSP hardware can deliver. Multiply accumulates must be performed at an ever-increasing rate and demands in the billions of MACs per second range are now common.

Field programmable gate arrays (FPGAs) using distributed arithmetic algorithms can implement large numbers of taps at MHz data sample rates, outperforming DSP processors by one or two orders of magnitude. Sample rates can be efficiently handled over a wide range by applying the proper structure that just meets the required performance with the least number of FPGA configurable logic blocks (CLBs).

Serial distributed arithmetic operates at relatively slow sample rates (3 to 15 million samples per second) but consumes few CLBs. Bits are processed serially, but all taps are processed in parallel. If higher sample rates are required, structures are available which process 2, 3, 4, or all of the bits in parallel. Multi-rate FIR filters can also be supported, resulting in higher sample rates with the same number of resources.

Functional Description

The FIR filter coefficients are formatted and stored in distributed ROM memory (look-up tables) when the module is generated and full precision is maintained throughout all stages of arithmetic computation. Hand-shake signals are provided to control the flow of data with a bit level clock.

The filter maintains full precision throughout processing. For example, a symmetrical 12-bit input data word and 14-bit coefficient would yield a 30-bit result (16-bit LUT + 2 bits adder tree growth + 12-bits data + 1-bit subtracter minus one bit for saturation). The maximum available resolution is automatically calculated by the SDA FIR dialog box after the balance of the other parameters have been entered and can be displayed by looking at the pull-down menu for Output Data Width. The user can adjust the output to fewer bits if desired. In the above example, the 30-bit number could be scaled to a 12-bit number if desired. Note that this does not significantly reduce the amount of logic required due to the nature of the module design. The important factor is that no noise is introduced into the system as a result of arithmetic operations.

An SDA FIR filter processes data in a bit serial fashion. This means that a data word of length N will require N clock cycles for a non-symmetric filter. This module processes ALL taps of the filter in parallel and therefore requires only N clock cycles to process ALL of the filter's taps. For a symmetrical filter an additional clock cycle is required. For a symmetric SDA FIR the number of clock cycles is N+1 per filter output. This will influence the frequency of the clock, **C**. The user must supply a clock which is at least N times the data sample rate for non-symmetric filters and N+1 times the data rate for symmetric.

Hand-shake Control

Data must be presented to the DATA IN bus when the RFD (Ready For Data) output control line is high and before the rising edge of CK (Clock). The ND (New Data) input control line must go high prior to the rising edge of CK, signifying that a new data word is available. ND must not be taken high again until RFD is high or data corruption within the filter may occur.

Valid output data appears on the RSLT output bus on a rising clock edge and remains stable until the next rising clock edge. This interval corresponds with the Result Ready output RDY going high.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

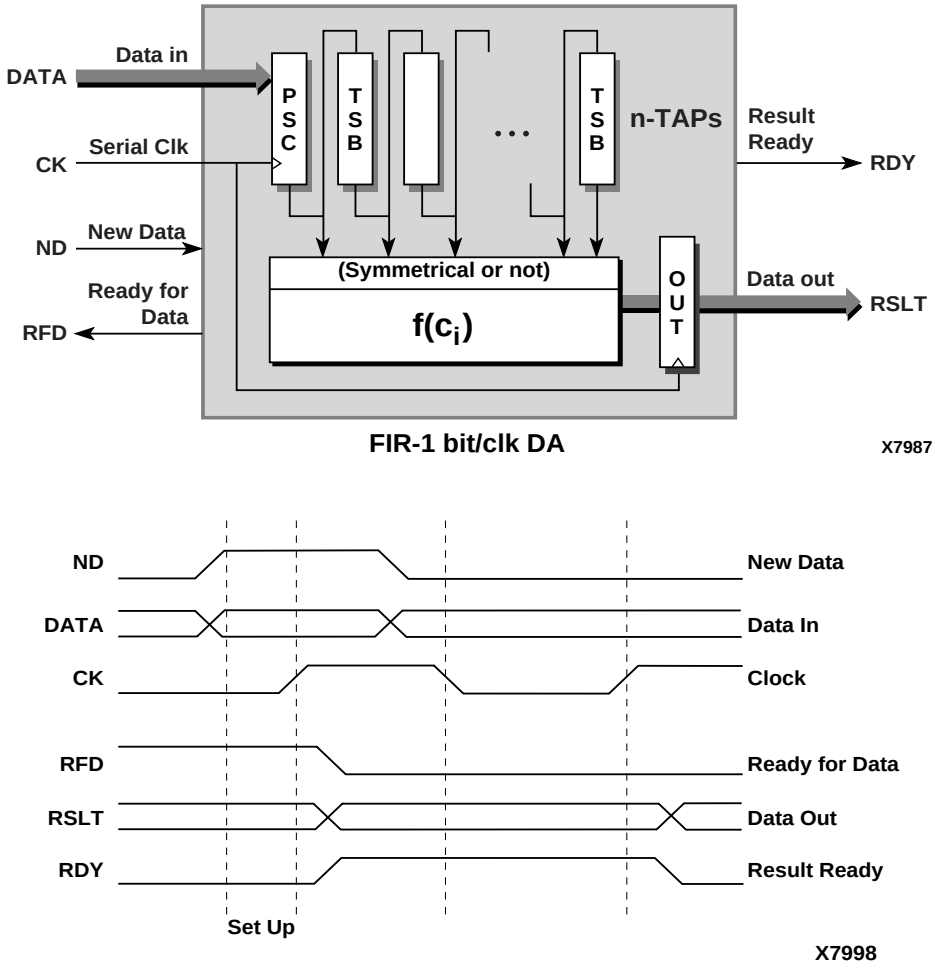


Figure 1: Serial Distributed Arithmetic FIR Filter Block and Timing Diagrams

Table 1: CORE Signal Pinout

Signal	Signal Direction	Description
DATA	Input	Parallel Data Input – N-bits wide
CK	Input	Bit rate clock
ND	Input	New Data, active high to indicate that the next rising edge of the clock will cause data to be loaded into the parallel-to-serial converter (PSC).
RFD	Output	Ready For Data – active high when the last data bit is about to leave the PSC and a new data word may be applied and loaded into the filter on the DATA input.
RSLT	Output	Parallel Data Out – N+x bits wide
RDY	Output	Result Ready – active high when the RSLT data is available

Core Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Input Data Width**: Select an input bit width from the pull-down menu. The valid range is 4-32. The data can be signed or unsigned.
- **Output Data Width**: Full arithmetic precision is carried through the adder trees and scaling accumulator. This parameter selects the number of bits to bring out to the next stage of processing. The valid range is 2-31.
- **Generate Cascadeable Section**: Allows this module to be combined with other filter modules to form a larger filter. See the online User's Guide for more information.
- **Symmetry:** Select Symmetric or Non-symmetric coefficients. The Negative Symmetry option is only enabled if you choose Symmetric.
- **Taps:** For symmetrical coefficients the range is 6 to 80. For non-symmetrical it is 6 to 40.
- **Shape:** For 27 to 40 taps two different shaped implementations are available. The rectangle of CLBs can be taller and thinner or shorter and wider. The default is Tall & Thin. The Shape parameter is only enabled if Symmetric coefficients are selected.
- **Coefficient Width:** This determines the width of the tables of sums of coefficients. The tables automatically grow from the width specified to accommodate the bit growth as the coefficient sums are calculated. The valid range is 4-24.
- **Trim Empty ROMs:** This option will eliminate the hardware necessary to implement ROM memory if the contents turn out to be all zeros or all ones. This reduces the fanout required, but has the disadvantage of requiring a new filter structure to be generated if the coefficients change.
- **Load Coefficients:** Get coefficients from a specified file.
- **Show Coefficients**: Display the coefficients after they have been loaded.
- **.coe file**: Displays the name of the coefficient file. This field is read-only.

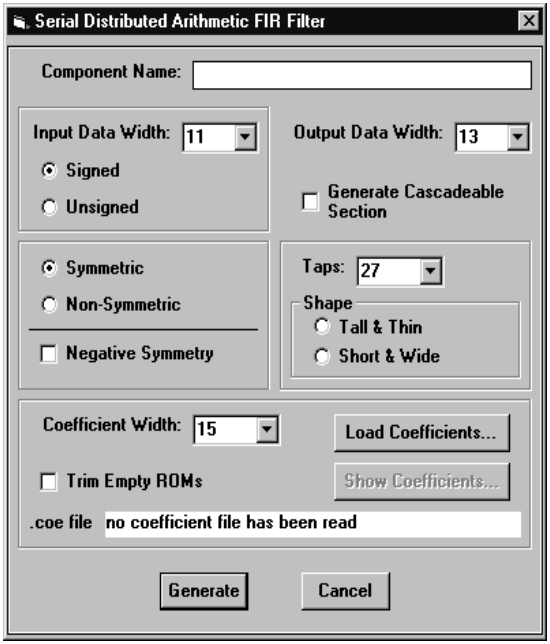


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the approximate number of CLBs required for various example bit widths. Table 3 shows the maximum speed for data sample rates using XC4000E-1 devices. The sample rate is calculated by dividing the system clock by the number of bits in the data word (N) for non-symmetrical FIR filters and by N+1 for symmetric filters. Sample rates are independent of the number of taps, but a small increase in delay per clock is a function of the width of the coefficients due to the increase in the width for the adders.

Table 2: CLB Utilization for Example SDA FIR Filter Implementations

Taps	Symmetry	CLB Counts for Example Data Word Widths[1]							
		5-bit	8-bit	10-bit	12-bit	14-bit	16-bit	18-bit	20-bit
8	Symmetrical		33	36	39	42	45	52	55
	Non-symmetrical		46	54	59	64	69	77	85
16	Symmetrical	53	61	69	71	76	81	96	102
	Non-symmetrical		80	95	104	112	123	138	
24	Symmetrical	80	89	101	108	116	127	146	154
	Non-symmetrical		101	114	127	140	153	174	187
32	Symmetrical	93	107	118	126	137	148	175	182
40	Symmetrical	116	138	154	165	179	191	226	239
48	Symmetrical		158	173	187	202	217	246	261
64	Symmetrical		197	215	233	250	268	305	323
80	Symmetrical		236	257	278	299	320	364	385

Note:

1. Coefficient width is equal to the word width.

Table 3: SDA FIR Filter Data Sample Rates in MHz Using XC4000E-1

Symmetry	5-bit	8-bit	10-bit	12-bit	14-bit	16-bit	18-bit	20-bit
Symmetrical	13.3	8.9	7.3	6.2	5.3	4.7	4.2	3.8
Non-symmetrical	16.0	10.0	8.0	6.7	5.7	5.0	4.4	4.0

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Dual-Channel Serial Distributed Arithmetic FIR Filter

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Input data words from 4 to 16 bits
- Output data words from 2 to 31 bits
- Coefficient width from 4 to 24 bits
- Taps from 3 to 40 for non-symmetric filters
- Support for cascading multiple filters together to create a larger filter
- Full precision
- Scaleable output
- Input and output synchronization signals
- Registered output
- 2's complement input data
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

General Description

FIR filters are one of the most basic building blocks used in digital signal processing, taxing the performance that DSP hardware can deliver. Multiply accumulates must be performed at an ever-increasing rate and demands in the billions of MACs per second range are now common.

Field programmable gate arrays (FPGAs) using distributed arithmetic algorithms can implement large numbers of taps at MHz data sample rates, outperforming DSP processors by one or two orders of magnitude. Sample rates can be efficiently handled over a wide range by applying the proper structure that just meets the required performance with the least number of FPGA configurable logic blocks (CLBs).

Serial distributed arithmetic operates at relatively slow sample rates (3 to 15 million samples per second) but consumes few CLBs. Bits are processed serially, but all taps are processed in parallel. If higher sample rates are required, structures are available which process 2, 3, 4, or all of the bits in parallel. Multi-rate FIR filters can also be supported, resulting in higher sample rates with the same number of resources.

The dual-channel SDA FIR filter can process two independent channels of data, where both channels have the same set of filter coefficients. The implementation requires no increase in the number of CLBs used as compared to the single SDA FIR filter. The dual-channel SDA FIR filter can only implement non-symmetric filters up to 40 taps.

Functional Description

The FIR filter coefficients are formatted and stored in distributed ROM memory (look-up tables) when the module is generated and full precision is maintained throughout all stages of arithmetic computation. Hand-shake signals are provided to control the flow of data with a bit level clock.

The filter maintains full precision throughout processing. For example, a non-symmetrical 12-bit input data word and 14-bit coefficient would yield a 30-bit result (16-bit LUT + 2 bits adder tree growth + 12-bits data + 1-bit subtracter minus one bit for saturation). The maximum available resolution is automatically calculated by the dual-channel SDA FIR filter dialog box after the balance of the other parameters have been entered and can be displayed by looking at the pull-down menu for Output Data Width. The user can adjust the output to fewer bits if desired. In the above example, the 30-bit number could be scaled to a 12-bit number if desired. Note that this does not significantly reduce the amount of logic required due to the nature of the module design. The important factor is that no noise is introduced into the system as a result of arithmetic operations.

A dual-channel SDA FIR filter processes data in a bit serial fashion, alternately taking data from each channel. This means that to process a data word of length N from a single channel will require N clock cycles. Because two data channels are processed, the module requires 2N clock cycles to process one data word from each channel. This module processes ALL taps of the filter in parallel and therefore requires only 2N clock cycles to process ALL of the filter's taps for each channel. This will influence the frequency of the clock, **C**. The user must supply a clock which is at least 2N times the data sample rate of each channel.

Multiple dual-channel SDA FIR filters can be cascaded together to form a larger filter. See the online User's Guide for more information.

Hand-shake Control

Data must be presented to the two DATA IN buses when the RFD (Ready For Data) output control line is high and before the rising edge of CK (Clock). The ND (New Data) input control line must go high prior to the rising edge of CK, signifying that a new data word is available. ND must not be taken high again until RFD is high or data corruption within the filter may occur.

Valid output data appears on the two RSLT output buses on the same rising clock edge and remains stable until the next rising clock edge. This interval corresponds with the Result Ready output RDY going high.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

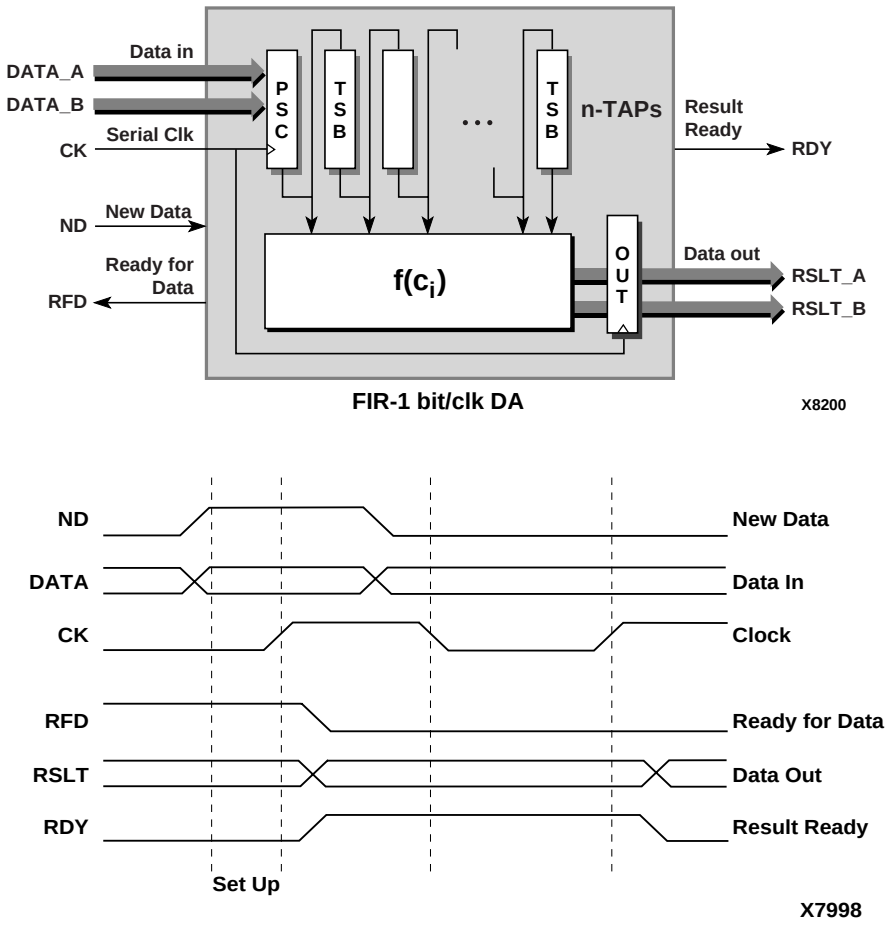


Figure 1: Dual-Channel Serial Distributed Arithmetic FIR Filter Block and Timing Diagrams

Table 1: CORE Signal Pinout

Signal	Signal Direction	Description
DATA_A	Input	Channel A Data Input – N-bits wide
DATA_B	Input	Channel B Data Input – N-bits wide
CK	Input	Bit rate clock
ND	Input	New Data, active high to indicate that the next rising edge of the clock will cause data to be loaded into the parallel-to-serial converter (PSC).
RFD	Output	Ready For Data – active high when the last data bit is about to leave the PSC and a new data word may be applied and loaded into the filter on the DATA input.
RSLT_A	Output	Channel A Data Out – N+x bits wide
RSLT_B	Output	Channel B Data Out – N+x bits wide
RDY	Output	Result Ready – active high when the RSLT data is available

Core Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Input Data Width**: Select an input bit width from the pull-down menu. The valid range is 4-16. If **Reverse Channel B Coefficients** is selected, the valid range is 4-8.
- **Output Data Width**: Full arithmetic precision is carried through the adder trees and scaling accumulator. This parameter selects the number of bits to bring out to the next stage of processing. The valid range is 2-31.
- **Taps:** Select the number of taps from the pull-down menu. The valid range is 3-40.
- **Coefficient Width:** This determines the width of the tables of sums of coefficients. The tables automatically grow from the width specified to accommodate the bit growth as the coefficient sums are calculated. The valid range is 4-24.
- **Trim Empty ROMs:** This option will eliminate the hardware necessary to implement ROM memory if the contents turn out to be all zeros or all ones. This reduces the fanout required, but has the disadvantage of requiring a new filter structure to be generated if the coefficients change.
- **Load Coefficients**: Get the coefficients from a specified file.
- **Show Coefficients**: Display the coefficients after they have been loaded.
- **.coe file**: Displays the name of the coefficient file. This field is read-only.
- **Reverse Channel B Coefficients**: Make Channel B's filter coefficients the reverse of Channel A's. Selecting this option reduces the valid range of the input data width to 4-8.
- **Channel A/B Input Data**: Select Serial or Parallel for both the Channel A and Channel B inputs. When **Reverse Channel B Coefficients** is selected, this field changes to **Channel A Input Data** only. It is useful when cascading multiple filters together to form a larger filter.
- **Channel B Input Data**: Select Serial or Parallel for the input. This field is enabled only when **Reverse Channel B Coefficients** is selected. It is useful when cascading multiple filters together to form a larger filter.

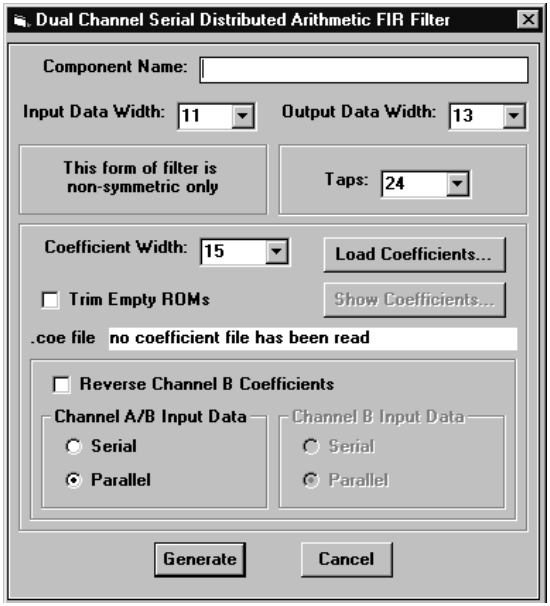


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the approximate number of CLBs required for various example bit widths. Table 3 shows the maximum speed for data sample rates using XC4000E-1 devices. The sample rate is calculated by dividing the system clock by the number of bits in the data word (N). Sample rates are independent of the number of taps, but a small increase in delay per clock is a function of the width of the coefficients due to the increase in the width for the adders.

Table 2: CLB Utilization for Example Dual-Channel SDA FIR Filter Implementations

Taps	Symmetry	CLB Counts for Example Data Word Widths[1]						
		8-bit	10-bit	12-bit	14-bit	16-bit	18-bit	20-bit
8	Non-symmetrical	46	54	59	64	69	77	85
16	Non-symmetrical	80	95	104	112	123	138	
24	Non-symmetrical	101	114	127	140	153	174	187

Notes:
1. Coefficient width is equal to the word width.

Table 3: Dual-Channel SDA FIR Filter Data Sample Rates in MHz Using XC4000E-1

Symmetry	5-bit	8-bit	10-bit	12-bit	14-bit	16-bit	18-bit	20-bit
Non-symmetrical	16.0	10.0	8.0	6.7	5.7	5.0	4.4	4.0

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Parallel Distributed Arithmetic FIR Filter

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Input data words from 4 to 32 bits
- Coefficient width from 4 to 24 bits
- Output data words from 2 to 31 bits
- Taps from 2 to 20 for symmetric filters
- Taps from 2 to 10 for non-symmetric filters
- Support for cascading multiple filters together to create a larger filter
- Support for symmetric, non-symmetric, and negative symmetric filters
- Full precision
- Scaleable output
- Input and output synchronization signals
- Registered output
- 2's complement input data
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

General Description

FIR filters are one of the most basic building blocks used in digital signal processing, taxing the performance that DSP hardware can deliver. Multiply accumulates must be performed at an ever-increasing rate and demands in the billions of MACs per second range are now common.

Field programmable gate arrays (FPGAs) using distributed arithmetic algorithms can implement large numbers of taps at MHz data sample rates, outperforming DSP processors by one or two orders of magnitude. Sample rates can be efficiently handled over a wide range by applying the proper structure that just meets the required performance with the least number of FPGA configurable logic blocks (CLBs).

Functional Description

The FIR filter coefficients are formatted and stored in distributed ROM memory (look-up tables) when the module is generated and full precision is maintained throughout all stages of arithmetic computation.

The filter maintains full precision throughout processing. For example, a symmetrical 8-tap filter processing 12-bit input data words and 8-bit coefficients would yield a 24-bit result. The maximum available resolution is automatically calculated by the PDA FIR dialog box after the balance of the other parameters have been entered and can be displayed by looking at the pull-down menu for Output Data Width. The user can adjust the output to fewer bits if desired. In the above example, the 24-bit number could be scaled to a 12-bit number if desired. Note that this does not significantly reduce the amount of logic required due to the nature of the module design. The important factor is that no noise is introduced into the system as a result of arithmetic operations.

A PDA FIR filter processes data in parallel. This means that it can process a new data word every clock cycle.

Valid output data appears on the DATA_OUT output bus on a rising clock edge and remains stable until the next rising clock edge.

Pinout

Signal names for the schematic symbol are shown in Figures 1 and 2 and described in Table 1.

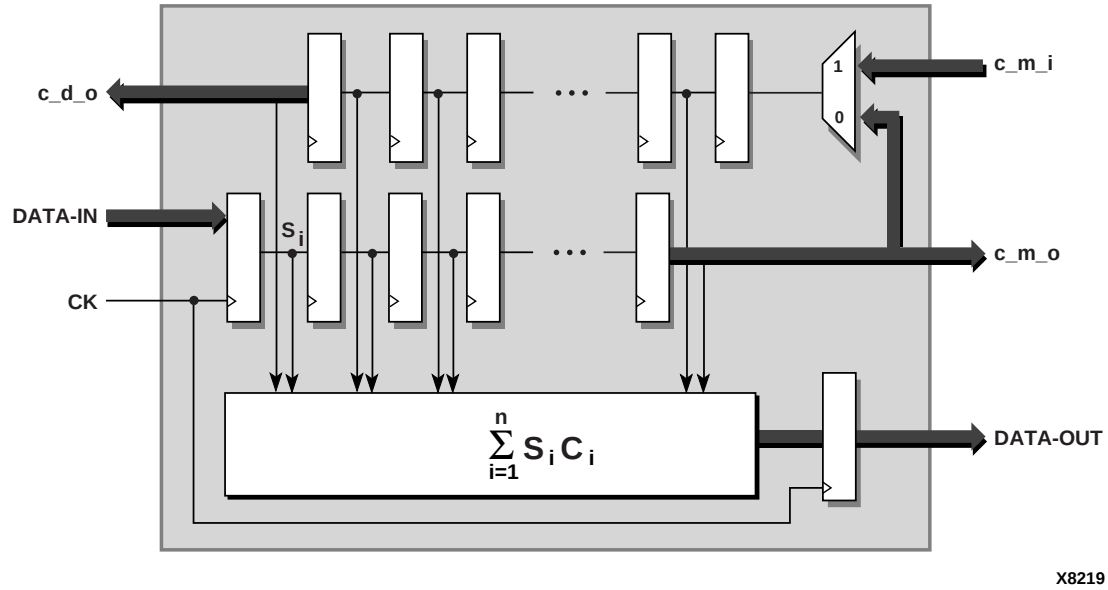


Figure 1: Symmetrical Parallel Distributed Arithmetic FIR Filter Block Diagram

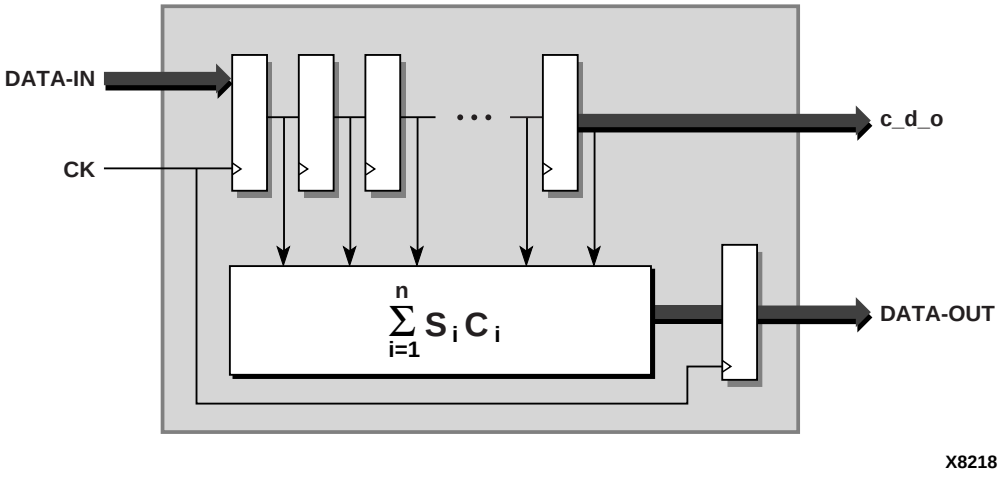


Figure 2: Non-symmetrical Parallel Distributed Arithmetic FIR Filter Block Diagram

Table 1: CORE Signal Pinout

Signal	Signal Direction	Description
DATA_IN	Input	Parallel Data Input
CK	Input	Clock
c_m_i	Input	Cascade Mid Input – Parallel data from successive cascaded filter
c_m_o	Output	Cascade Mid Output – Parallel DATA_IN shifted out from a Symmetric filter if cascading. Connect to DATA_IN of successive filter
c_d_o	Output	Cascade Data Output – Parallel DATA_IN shifted out from filter
DATA_OUT	Output	Filter result – Parallel Data Out

Cascading Multiple PDA FIR Filters

Multiple PDA FIR filters may be cascaded together to build filters with larger numbers of taps. The CASCADE option must be selected in the GUI to enable filters to be cascaded. Every filter in the cascade chain must have this option enabled, except for the last filter in the chain. For the last filter, the CASCADE option must be deselected.

The c_d_o output is always provided whether CASCADE has been selected or not.

To maintain similar latency through each filter, cascade filters must have the same number of taps. Note that the latency through each filter in the chain will vary if the number of taps varies with each filter.

Cascading Multiple Symmetric Filters

Multiple symmetric filters may be cascaded as shown in Figure 3. Input data is presented to the first filter (filter A) at the DATA_IN pin. Connect c_m_o of the first filter to DATA_IN of the next filter, and c_m_i of the first filter to c_d_o of the next filter. The c_m_o and c_m_i signals of the last filter in the chain are internally connected. These pins may therefore be left unconnected.

Only the last filter in the chain is permitted to have an odd number of taps. All other filters must have an even number of taps.

Cascading Multiple Non-Symmetric Filters

Multiple non-symmetric filters may be cascaded as shown in Figure 4. Input data is presented to the first filter (filter A) at the DATA_IN pin. Then, for each pair of adjacent filters, connect c_d_o of the first filter to DATA_IN of the next filter.

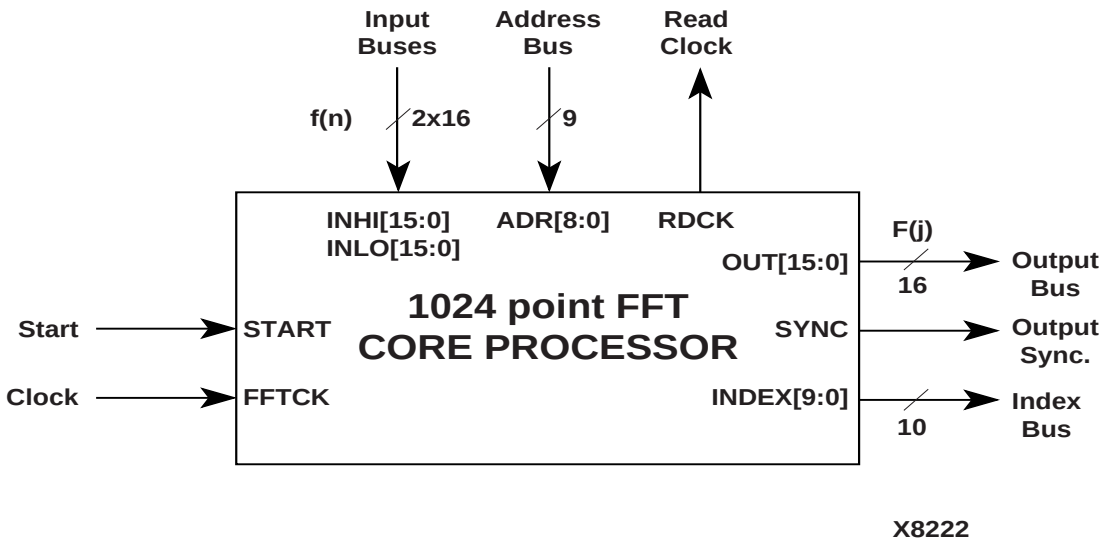


Figure 3: Cascading Symmetric Filters

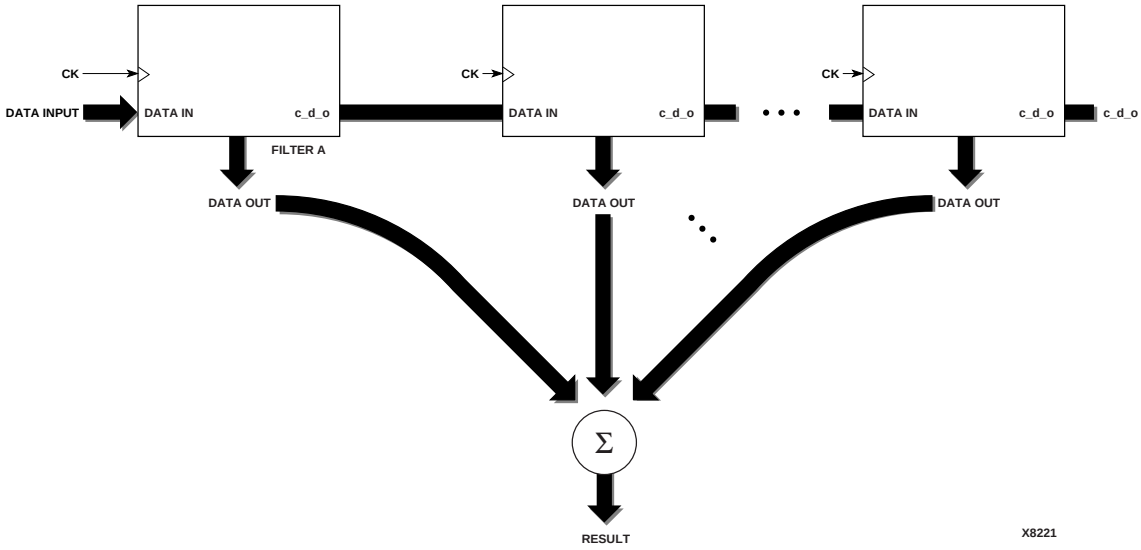


Figure 4: Cascading Non-symmetric Filters

Core Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 5. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Input Data Width**: Select an input bit width from the pull-down menu. The valid range is 4-16.
- **Input Data Sign:** Signed or Unsigned.
- **Output Data Width**: Full arithmetic precision is carried through the adder trees and scaling accumulator. This parameter selects the number of bits to bring out to the next stage of processing. The valid range is 4-20.
- **Symmetry:** Select Symmetric or Non-symmetric coefficients. The Negative Symmetry option is only enabled if you choose Symmetric and the input data is Signed.
- **Generate Cascadeable Section**: Allows this module to be combined with other filter modules to form a larger filter. See the online User's Guide for more information.
- **Taps:** For symmetrical coefficients the range is 2 to 20. For non-symmetrical it is 2 to 10.
- **Coefficient Width:** This determines the width of the tables of sums of coefficients. The tables automatically grow from the width specified to accommodate the bit growth as the coefficient sums are calculated. The valid range is 4-16.
- **Trim Empty ROMs:** This option will eliminate the hardware necessary to implement ROM memory if the contents turn out to be all zeros or all ones. This reduces the fanout required, but has the disadvantage of requiring a new filter structure to be generated if the coefficients change.
- **Load Coefficients:** Get coefficients from a specified file.
- **Show Coefficients**: Display the coefficients after they have been loaded.
- **.coe file**: Displays the name of the coefficient file. This field is read-only.

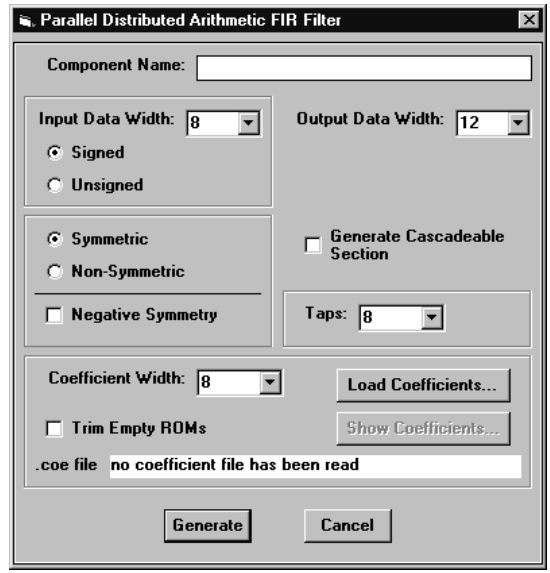


Figure 5: CORE Generator Dialog Box

Core Resource Utilization

The PDA FIR filter processes one data word every clock cycle. The sample rate is therefore equal to the clock rate.

Tables 2 and 3 show the approximate number of CLBs required for various example bit widths. Tables 4 and 5 show the maximum speed (in MHz) for data sample rates using XC4036XL-1 devices.

Table 2: CLB Utilization for Symmetrical PDA FIR filters

# Taps	# Bits (Data and Coefficients)		
	8	10	12
4	142	191	252
8	185	241	309
10	252	333	430

Table 3: CLB Utilization for Non-symmetrical PDA FIR filters

# Taps	# Bits (Data and Coefficients)		
	8	10	12
2	95	156	204
3	110	173	224
4	115	325	231
5	165	252	331

Table 4: Performance for Symmetrical PDA FIR filters

# Taps	# Bits (Data and Coefficients)		
	8	10	12
4	88	78	77
8	80	74	65
10	70	74	60

Table 5: Performance for Non-symmetrical PDA FIR filters

# Taps	# Bits (Data and Coefficients)		
	8	10	12
2	78	81	68
3	74	77	64
4	74	73	64
5	75	66	70

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

LogiCORE™

DFT Cores
(Real Data In, Complex Data Out)

February 8, 1998 — Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

DFT technology developed by Rice Electronics.

Features

- 2's complement, fixed point arithmetic
- Real-valued input data (15 bit)
- Complex output data (16 bit ~86 dB available output SNR)
- Integrated Input Buffer (no external memory requirements)
- Transform size (N) = 32, 64, or 128
- Process real-time sampling rates >1Mhz (>12Mhz for N=32)
- Simplified interface (nominal support logic required)
- Synchronous design, optimized for XC4000E, EX, and XL families of FPGAs
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Applications

- Communications (high speed modems, transmultiplexers)
- Instrumentation (medical, scientific, test)
- Multi-media (signal compression/decompression)
- Military (radar, EW, ELINT, ESM)

General Description

The 16-bit Discrete Fourier Transform (DFT) Cores are functionally complete elements. The designs present a simplified interface and require no external memory. The Cores target the Xilinx XC4000E, EX, and XL FPGA product series.

The Cores render the following transform for a real-valued input vector, f(n):

$$F(j) = \sum_{n=0}^{N-1} f(n) e^{-2\pi i n \frac{j}{N}}$$

for n=0 to N-1, j=0 to (N/2-1)

where:

F(j) = output (frequency domain) coefficients
f(n) = input (time domain) sequence
N = length of input sequence (transform size)

The Cores accept a real-valued input sequence f(n) and produce a complex output sequence F(j). Due to the real-valued nature of f(n), only the lower half of the set of F(j) is generated. The upper half of the F(j) is the complex conjugate of the lower half, and therefore represents redundant information.

The Cores are optimized for small to medium size transforms (32 - 128 points). A 32-point transform can be executed in less than 5 microseconds (XC4000-3). General specifications of the Cores are listed in Table 1.

Table 1: 16-bit DFT Family Parameters

Core Name	N = Size of Transform	P = Clock Periods[1]	Clock Speed[2]	Execution Time[3]	Core Size[4]
16b32pt	32 points	144	59 Mhz	2.5 usecs	274 CLBs
16b64pt	64 points	1088	59 Mhz	18.5 usecs	302 CLBs
16b128pt	128 points	4224	59 Mhz	71.8 usecs	394 CLBs

1. P = number of clock periods for transform execution
2. Preliminary maximum clock speeds based on XC4000E-1 series. Higher clock speeds are expected pending fixed layout.
3. Execution Time = P/(Clock Speed in Mhz)
4. XC4008 = 324 CLBs
   XC4010 = 400 CLBs
   XC4085 = 3,136 CLBs

Functional Description

The 16-bit DFT Cores are functionally complete processors requiring minimal external control.

The Cores incorporate an Input Buffer, which is loaded with the vector f(n) to be processed. This constitutes the **Initialization** state of the Core.

After initialization, the Core can perform the DFT function with no external control. During this time, output coefficients F(j) are produced by the Core at a constant rate. This is termed the **Execution** state of the Core.

The DFT Cores possess physically separate input and output interfaces.

The input interface is used only during the **Initialization** state. Separate data and address busses are provided, allowing access to individual locations of the Input Buffer memory.

The output interface is active only during the **Execution** state. This interface provides physically separate busses for output coefficients, and index information. The index identifies the specific output coefficient F(j).

Pinout

DATA INPUT

Prior to processing, an input vector f(n) is loaded to the Core's Input Buffer memory. Data is written to the Input Buffer using the 15 bit Data Input bus (IN[14:0]), the Address bus (ADR[x:0]), and a Write Enable (WREN).

DATA OUTPUT

Output consists of the 16-bit Output bus (FREQ[15:0]), the Index bus (INDEX[x:0]), and an output Synchronization signal (SYNC). Output coefficients F(j) are presented on the Output bus. The corresponding value of j appears on the Index bus.

The Index bus identifies the component (imaginary or real) and the coefficient number (j), associated with the data on the Output bus.

TIMING INPUTS

Timing inputs consist of a START signal, and a continuous clock (MULCK). A pre-defined number of clock pulses is required for execution of the transform (see Table 1).

Figure 1 illustrates the 16-bit DFT Core interface signals.

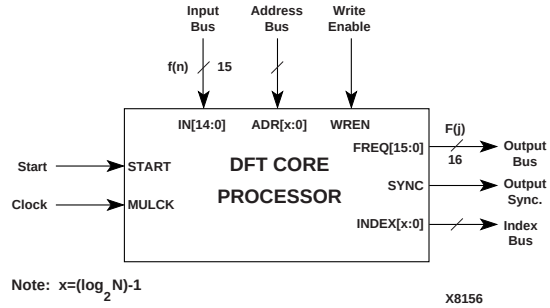


Figure 1:16-bit DFT Core Interfaces

The format of the interface signals is summarized in Table 2.

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
START	Input	Logic level dictates operational state: 0=Initialization, 1=Execution
MULCK	Input	Continuous clock (see Table 1 for max.frequency)
WREN	Input	Logic level controls writing to Core Input Buffer memory: 1=Write, 0=No action. Active only when START is low
IN[14:0]	Input	One sign bit + 14 magnitude bits; 2's complement notation. Uni-directional input bus for f(n)
ADR[x:0]	Input	Consists of $\log_2 N$ bits, where N is transform size (e.g., 5 address lines for N=32). Specifies write address to Input Buffer memory
SYNC	Output	Positive pulse indicates presence of new output component on FREQ output bus (pulse width = 1 clock period)
FREQ[15:0]	Output	One sign bit + 15 magnitude bits; 2's complement notation. Uni-directional output bus for F(j)
INDEX[x:0]	Output	Consists of $\log_2 N$ bits, where N is transform size (e.g., 5 bits for N=32). Identifies F(j) value on FREQ bus. Index MSB indicates imaginary or real component: 0=Imaginary, 1=Real. Remainder of Index indicates j

Timing and Control

The 16-bit DFT cores possesses two operational states as follows:

1) **Initialization** (input vector f(n) is loaded into Core)

2) **Execution** (output vector F(j) is produced by Core)

The Start signal determines whether the Core is in the Initialization or Execution state. The Core requires a continuous clock input (MULCK) in both states. (Table 1 lists maximum clock frequency.)

Initialization (START=0)

(Input State)

When Start is low, DFT processing is disabled and the Input Buffer can be initialized (loaded). Individual address locations are loaded by means of the Address Bus, Input Bus, and Write Enable (WREN).

The Core latches these inputs on the rising edge of MULCK. All inputs should be stable at least 20ns prior to the rising edge of MULCK. When WREN is high and Start is low, data is written to the specified address.

The input, f(n), must be loaded into the following Input Buffer address locations:

f(0)--->location 0
f(1)--->location 1
f(2)--->location 2
•
•
•
f(N-1)--->location (N-1)

When Start is high, WREN has no effect on the Input Buffer.

Execution (START=1)

(Output State)

Setting Start high begins DFT processing. The number of clock periods (P) required to execute the DFT is given in Table 1.

Access to the Core internal memory is disabled during the execution phase.

During execution, the frequency coefficients F(j) are produced at the Output bus. Separate 16 bit values are generated for the real and imaginary components of F(j). The components are produced in a specific order, as defined in Table 3.

Table 3: DFT Output Order

Output Bus	Index Bus LSBs	Index Bus MSB
$F_I(0)$ [First out]	0	0
$F_I(0)$	0	0
$F_R(0)$	0	1
$F_R(0)$	0	1
$F_I(1)$	1	0
$F_I(N/2 - 1)$	N/2 - 1	0
$F_R(1)$	1	1
$F_R(N/2 - 1)$	N/2 - 1	1
$F_I(2)$	2	0
$F_I(N/2 - 2)$	N/2 - 2	0
$F_R(2)$	2	1
$F_R(N/2 - 2)$	N/2 - 2	1
$F_I(3)$	3	0
$F_I(N/2 - 3)$	N/2 - 3	0
$F_R(3)$	3	1
$F_R(N/2 - 3)$	N/2 - 3	1
:	:	:
$F_I(N/4)$	N/4	0
$F_I(N/2 - N/4)$	N/2 - N/4	0
$F_R(N/4)$	N/4	1
$F_R(N/2 - N/4)$ [Last out]	N/2 - N/4	1

Note: The 16-bit DFT core generates redundant information at the beginning and end of the transform, resulting in (N+4) total output values.

The output components are produced at a constant rate, with P/(N+4) clock pulses between outputs (P=number of clocks for transformation, N=transform size).

The Output Sync line produces a positive pulse for each new output component. The pulse is one clock cycle wide, and occurs one clock cycle after the appearance of a new component. Figure 2 illustrates output timing.

The Index bus identifies the component on the Output Bus. The Index MSB is low (0) for imaginary output and high (1) for real output. The remainder of the Index bus represents j, which ranges from 0 (DC) to (N/2-1).

An Index value of all zeros indicates no meaningful data is on the Output bus. This corresponds to the component $F_I(0)$, which is defined by the DFT equation as zero, for a real-valued input sequence. Simulation output for this component may be either undefined (xxxx) or zero.

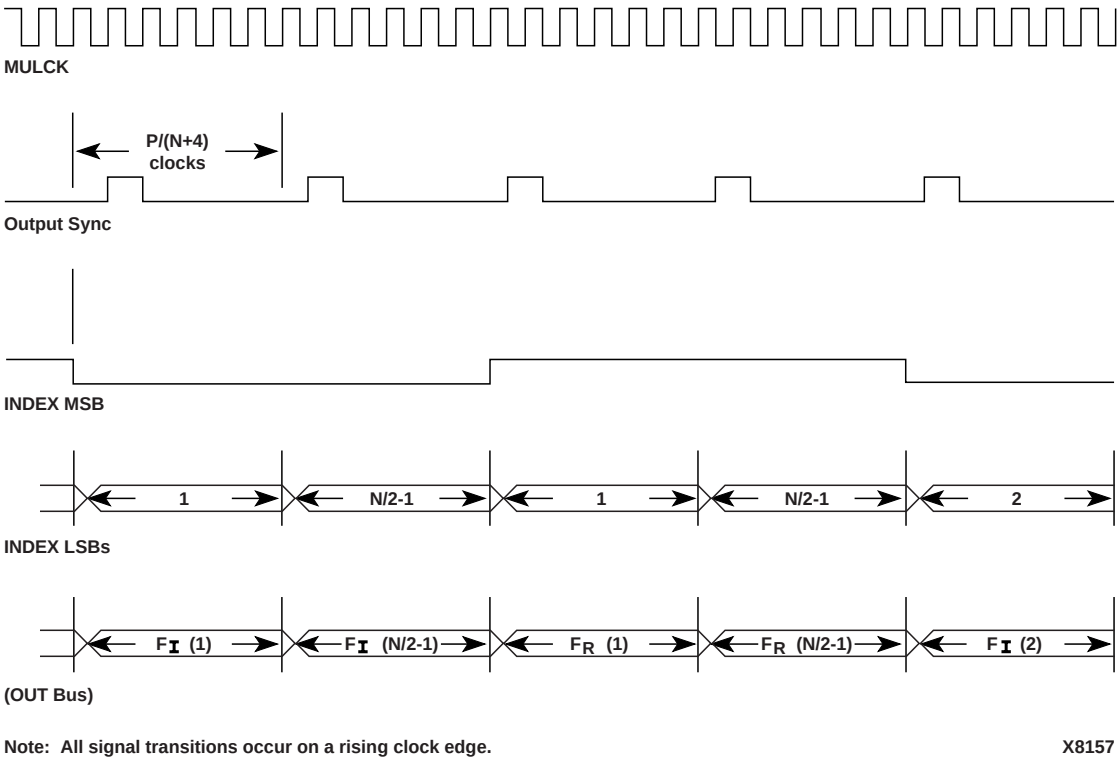


Figure 2: Execution State – Output Timing

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

For information on Rice Electronics, contact:

Rice Electronics
PO Box 741
Florissant, MO 63032
Phone: +1 314-838-2942
Fax: +1 314-838-2942
E-mail: ricedsp@aol.com

LogiCORE™ FFT Core (1024 Points)

February 8, 1998 | Product Specification – PRELIMINARY

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

FFT technology developed by Rice Electronics.

Features

- 2's complement, fixed-point arithmetic
- Real-valued input data (15 bit)
- Complex output data (16 bit ~86 dB available output SNR)
- Transform size (N) = 1024
- No programming required
- No "twiddle factor" memory required (internal to Core)
- Process real-time sampling rates ~2Mhz
- Simplified interface (nominal support logic required)
- Synchronous design, optimized for XC4000E, EX, and XL families of FPGAs
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Applications

- Communications (high speed modems, transmulti-plexers)
- Instrumentation (medical, scientific, test)
- Multi-media (signal compression/decompression)
- Military (radar, EW, ELINT, ESM)

General Description

The 1024-point Fast Fourier Transform (FFT) Core is a functionally complete processor. The design requires a 1024-point external data memory and nominal interface support. The Core targets the Xilinx XC4000E, EX, and XL FPGA product series.

The Core renders the following transform for a real-valued input vector, f(n):

$$F(j) = \sum_{n=1}^{N-1} f(n) e^{\frac{-2\pi i j n}{N}}$$

for j=0 to (N/2-1)

where:

F(j) = output (frequency domain) coefficients
f(n) = input (time domain) sequence
N = length of input sequence (transform size)

The Core accepts a real-valued input sequence f(n) and produces a complex output sequence F(j). Due to the real-valued nature of f(n), only the lower half of the set of F(j) is generated. The upper half of the F(j) is the complex conjugate of the lower half, and therefore represents redundant information.

General specifications of the FFT Core are listed in Table 1.

Functional Description

The 1024-point FFT Core is a functionally complete processor requiring minimal external control. Only an External Memory (1k word) is required for Core operation. The External Memory holds the input vector f(n) to be transformed.

The Core itself requires no initialization, and may be activated whenever valid data is present in External Memory. When in operation, the Core must have exclusive access to External Memory. The Core performs "read-only" accesses to the Memory (no write operations).

Table 1: 1024-Point FFT Parameters

Core Name	N = Size of Transform	P = Clock Periods[1]	Clock Speed[2]	Execution Time[3]	Core Size[4]
1024 FFT	1024 points	17408			532

1. P = number of clock periods for transform execution
2. Maximum clock speed based on XC4000E-3 series
3. Execution Time = P/(Clock Speed in Mhz)
4. Approximately 70% utilization of F/G function generators for XC4013 device

The 1024-point FFT Core requires no external storage of constants. "Twiddle-factors" are generated internally to the Core and require no user programming.

The Core possesses physically separate input and output interfaces.

The input interface has separate data and address buses for accessing External Memory. While External Memory consists of 1024 words, only 9 address bits are required from the Core. This is due to the simultaneous access of two memory words on every read cycle, as explained below.

The output interface presents output coefficients F(j) at a constant rate. This interface provides physically separate buses for output coefficients and index information. The index identifies the specific output coefficient F(j).

Pinout

DATA INPUT

The input interface includes dual 16-bit unidirectional data buses to the Core (INHI[15:0], INLO[15:0]) and a 9-bit address bus (ADR[8:0]) from the Core. These buses support read-only operations from External Memory during FFT processing.

DATA OUTPUT

Output consists of the 16-bit unidirectional Output bus (OUT[15:0]), the Index bus (INDEX[9:0]), and an output Synchronization signal (SYNC). Output coefficients F(j) are presented on the Output bus. The corresponding value of j appears on the Index bus.

The Index bus identifies the component (imaginary or real) and the coefficient number (j), associated with the data on the Output bus.

TIMING INPUTS

Timing inputs consist of a START signal and a continuous clock (FFTCK). A pre-defined number of clock pulses is required for execution of the transform (see Table 1).

Figure 1 illustrates the 1024-point FFT Core interface signals. The format of the interface signals is summarized in Table 2.

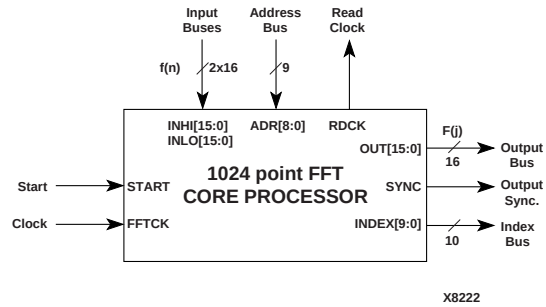


Figure 1: 1024-point FFT Core Interface

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
START	Input	Logic level dictates operational state: 0=Reset State (Dormant), 1=Execution State
FFTCK	Input	Continuous clock (see Table 1 for max.frequency)
INHI[14:0] INLO[14:0]	Input	One sign bit + 14 magnitude bits; 2's complement notation. Dual unidirectional input buses for f(n)
ADR[8:0]	Input	Unsigned 9-bit bus. Specifies read address to External Memory
RDCK	Output	Continuous clock from Core. Synchronizes access to External Memory. Derived from FFTCK (+2)
OUT[15:0]	Output	One sign bit + 15 magnitude bits; 2's complement notation. Unidirectional output bus for F(j)
SYNC	Output	Positive pulse indicates presence of new output component (pulse width = one clock period)
INDEX[9:0]	Output	Unsigned 10-bit bus. Identifies F(j) value on OUT bus. Index MSB indicates imaginary or real component: 0=Imaginary, 1=Real. Remainder of Index indicates j

Timing and Control

When the START signal is low, the Core is "reset". This prepares the Core for execution of a new transform. START must go low for a minimum of one FFTCK period for reset to occur.

While START is low, the Core interfaces are inactive. FFT processing begins when START goes high.

Input Interface

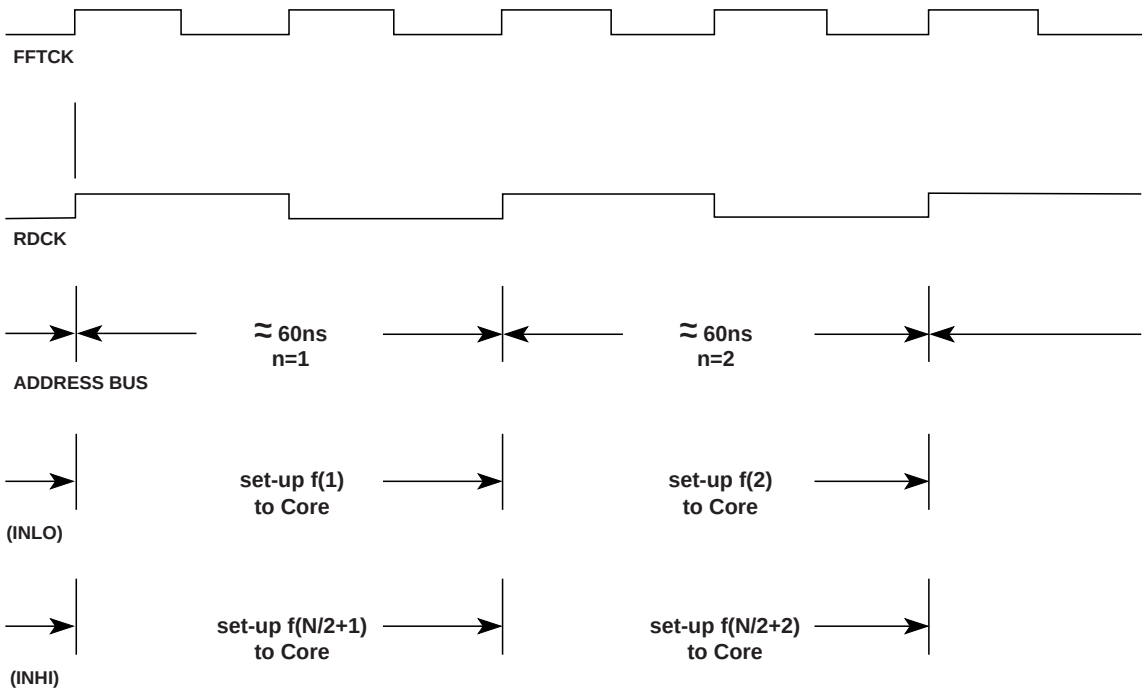
Input Interface Timing

The input interface requires exclusive (uninterrupted) access to the External Memory during FFT processing. This interface consists of an address bus, dual input data buses, and a continuous clock (RDCK).

RDCK is produced by the Core and can be used to synchronize External Memory to the Core. RDCK is derived from the Core input clock (FFTCK) and is half the FFTCK frequency.

The address bus from the Core changes on the rising edge of RDCK. The data buses to the Core must be stable by the next rising edge of RDCK. As seen in Figure 2, the time allocated for memory access is one RDCK (2 FFTCKs). This is equivalent to ~60ns at maximum clock speeds.

At the Core interface, the address and data buses terminate (respectively) at the output and input of FD type registers (XC4000 library primitives). Consequently, the Core contributes minimal logic delay in the memory access path. Accordingly, most of the RDCK period is available for delay through External Memory and associated I/O buffers.



Address bus represents index on input sequence f(n). INLO and INHI (input data buses) must respond with associated data samples within 1 RDCK period.

Continuous timing sequence is maintained for duration of FFT process.

Note: All signal transitions occur on rising clock edge.

X8224

Figure 2: Input Interface Timing

External Memory Organization

The input buffer, f(n), must be accessible in two separate halves from External Memory. The two halves must be available simultaneously on the INLO and INHI data buses. This imposes the following organizational requirements on External Memory:

The lower half of f(n) must be available at the INLO bus from the following External Memory address locations:

f(0)--->location 0
f(1)--->location 1
f(2)--->location 2
•
•
•
f(N/2 - 1)--->location (N/2 - 1)

The upper half of f(n) must be available at the INHI bus from the following External Memory address locations:

f(N/2 + 0)--->location 0
f(N/2 + 1)--->location 1
f(N/2 + 2)--->location 2
•
•
•
f(N -1)--->location (N/2 - 1)

Note: the common 9-bit ADR bus is used to simultaneously address both halves of External Memory.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

For information on Rice Electronics, contact:

Rice Electronics
PO Box 741
Florissant, MO 63032
Phone: +1 314-838-2942
Fax: +1 314-838-2942
E-mail: ricedsp@aol.com

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

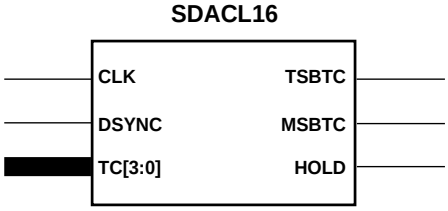


Figure 1: Core Schematic Symbol

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts a bit serial processing clock input, a data-sync (DSYNC), and a data-width signal (TC) on the input pins to control a Serial Distributed Arithmetic (SDA) process. The DSYNC input is used to start the bit serial process by signaling that a new data sample has arrived for processing. The internal control logic will start counting from 0 to a TC equal to N-2 for N-bit data to produce the outputs TSBTC, MSBTC, and HOLD. The DSYNC input is used to trigger the LSB of the SDA process. TSBTC resets the time-skew buffer and MSBTC disables the macros on the MSB of the SDA process until a new data sample is registered. The process is then repeated.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
CLK	Input	BIT-PROCESSING CLOCK – clocks the SDA bit serial process. Must have a frequency of N+1 times or greater than the data clock for N-bit data.
DSYNC	Input	DATA SAMPLE READY INPUT – DSYNC is used to reset the Control Logic to begin processing the next data sample. When DSYNC is asserted (HIGH), the input data is registered on the next rising edge of the C (CLOCK). This data is fed into the parallel-to-serial converter and the DSYNC is propagated through the data flow to trigger the LSB input of each cascaded stage of the SDA process. The active duration of DSYNC must be equal to one period, from rising-edge to rising-edge, of the bit processing clock and remain inactive for at least N clock cycles of the bit processing clock for N-bit data.

Table 1: Core Signal Pinout (cont.)

Signal	Signal Direction	Description
HOLD	Output	WAIT UNTIL NEXT SAMPLE – Used to terminate the SDA process. This signal is propagated through the data flow to disable each cascaded stage of the SDA process, as required to maintain data integrity for each clock cycle greater than N+1 for N-bit data.
TC[3:0]	Input	TERMINAL COUNT RESET CONSTANT – Used to terminate the incrementing of the internal counter at TC equal to N-2 for N-bit data.
TSBTC	Output	TIME-SKEW-BUFFER TERMINAL COUNT RESET – Used to reset the TSB Counter at N-1 for N-bit data.
MSBTC	Output	MOST SIGNIFICANT BIT TERMINAL COUNT – Used to indicate the MSB of the SDA process. This signal is propagated through the data flow to trigger the MSB Output of each cascaded stage of the SDA process.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Control Width**: Select a control width from the pull-down menu. The valid range is 2-5. The data width is 2 to this power.

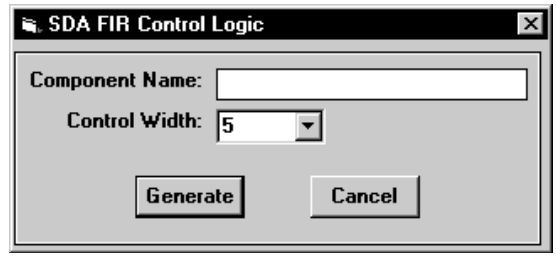


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

An RPM is not generated for this module, so the number of CLBs will vary with the Placer tool.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Fully parameterizable sin/cos look-up table
- Input address widths from 3 to 10 bits
- Output data widths from 4 to 16 bits
- User can determine the accuracy of the result by selecting the desired output width
- Utilizes fast internal XC4000 distributed ROM
- Easy to use: any table size and output resolution can be specified
- Efficient: calculates full 360 degree table from 90 degree segment
- Useful in high-speed modulation/demodulation applications
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Sine/Cosine module returns the value **sin(theta)** when the control input is High and the value **cos(theta)** when the control input is Low.

The module contains an internal ROM look-up table that stores only the values for the first quarter of the sine wave. Values of sin(theta) or cos(theta) are derived from this stored data by appropriate negations of an input theta and the value fetched from the internal table. The decisions to negate are based on the state of the two most significant bits of theta and the Ctrl input.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

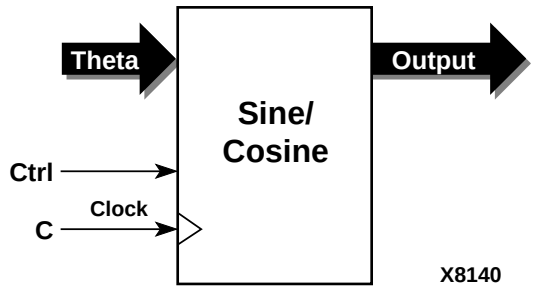


Figure 1: Block Diagram

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Theta	Input	Input value
Ctrl	Input	Control – when high, returns the sine of theta; when low, returns the cosine
C	Input	Clock – active on rising edge
Output	Output	Output value – the sine or cosine of theta

CORE Generator Parameters

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Address Width**: Select the width of the theta input bus. The valid range is 3 to 10. This value is also used to specify the depth of the required look-up table. The number of sample data points in the look-up table is 2 to the Address Width power.

- **Output Data Width**: Select the width of the look-up table from the pull-down menu. The valid range is 4 to 16. This value determines the precision of the output sin(theta) or cos(theta).

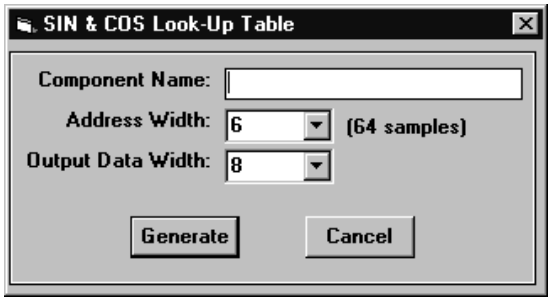


Figure 2: CORE Generator Dialog Box

Latency

The module has a pipeline latency that depends on the size of the address width:

Table 2: Latency

Address Width	Latency (# Clocks)
3 to 4 bits	1
5 bits	2
6 to 10 bits	3

Core Resource Utilization

The number of CLBs required for the look-up table depends on the size of the address and output data widths selected in the CORE Generator dialog box.

Table 3 shows the equations to calculate the maximum number of CLBs required for each available address width. In these equations **m** is the output bit width. (When using these equations, round down to the nearest integer.)

For example, if the address width is 10 (generating 1024 locations) and the output width is 8 bits, the look-up table requires 74 CLBs.

Table 3: Bit Width versus CLB Count

Address Width	CLB Count
3	(m+1)/2
4	m
5	m + (m+1)/2
6	m + 4
7	2m + 4
8	2m + 5 + (m+1)/2
9	4m + 6 + (m+1)/2
10	8m + 6 + (m+1)/2

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Non-Symmetric, 16-Deep Time Skew Buffer

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts a serial data input on the SDI pin. This data is stored in a series of Serially Cascaded, 16-bit Synchronous RAM-based shift registers. The RAM's output data is registered and presented on the Q pins of the Macro. Each register's Q pin is internally cascaded to the following RAM's Data Input. The 16-bit RAM-based shift register supports 1- to 16-bit wide data storage per stage. The RESET pin is used to control the Bit-Width of the serial-data to be stored via the control logic.

The RAM address is controlled via an internal, free-running, 16-bit counter (CNT04RE). The counter is controlled by the CE (active high) and RESET inputs. The CNT04RE generates an output, 0 to TC, where TC is controlled via the RESET signal from the Control Logic, SDACL16. The addressing sequence of the cascaded registered RAM is structured to increment the address of each data-bit by one. That is, the bit-n is stored at an address ADDR+1 for each consecutive RAM. This is because each data sample is registered prior to writing it into the following cascaded RAM. The registration of the RAM data is important to maintain a synchronous system, as the RAM is synchronous write and asynchronous read. Hence, the RAM data output will change, to represent the data stored at a given address, as the address changes. The Truth Table for this function is shown in Table 1.

Table 1: Truth Table

WE (mode)	Data	CE	TC	C	CNT	Q[N-1,0]
X	X	0	X	X	No Change	No Change
0 (read)	X	1	CNT<TC+1	1/4	CNT+1	Data@CNT
1 (write)	Data	1	CNT<TC+1	1/4	CNT+1	Data@CNT
0 (read)	X	1	CNT=TC+1	1/4	0	Data@CNT
1 (write)	Data	1	CNT=TC+1	1/4	0	Data@CNT

Pinout

Signal names for the schematic symbol are shown in Figure 1 and Table 2.

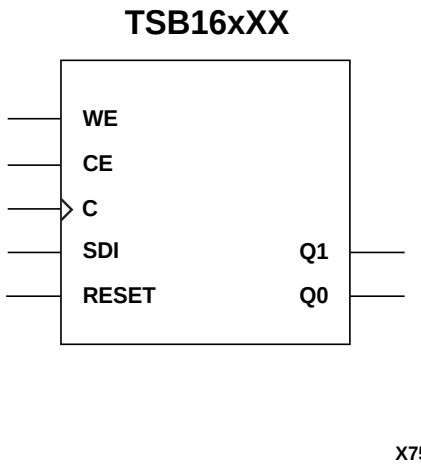


Figure 1: Core Schematic Diagram

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
WE	Input	WRITE ENABLE – Active high signal, Enables writing to the RAM in association with the internal address counter. This signal is required to hold a block a data for consecutive processing. (The WE pin can be externally tied to the CE if not used.)
CE	Input	CLOCK ENABLE – active high signal used to allow the transfer of data from the internal RAM to the registered output and the incrementing of the Address used to control the TSB RAM data transfer sequence. (Tie to VCC if not used.)
C	Input	CLOCK – clocks the output registers, TSB Counter, and internal synchronous RAM. CE must be active (high) to enable the RAM, Counter, and registers.
SDI	Input	SERIAL DATA INPUT – data presented on this pin is stored into the TSB when the CE (CLOCK ENABLE) is asserted (HIGH) and the rising edge of the C (CLOCK). SDI0 feeds the Input to the first RAM going up.
RESET	Input	TERMINAL COUNT RESET – Used to reset the internal RAM's address counter to zero. RESET should be set high at a count of N-2 for an N-bit sample during normal operation of cascaded data-sample.
Q[N-1, 0]	Output	REGISTERED OUTPUT DATA – the registered output of the RAM.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.

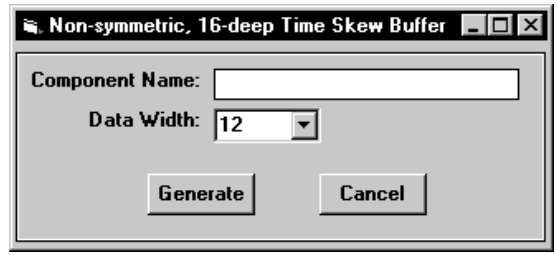


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 3 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 3: Bit Width versus CLB Count

Bit Width	CLB Count
2	3
3	4
4	4
5	5
6	5
7	6
8	6
9	7
10	7
11	8
12	8
13	9
14	9
15	10
16	10
17	11
18	11
19	12
20	12
21	13
22	13
23	14
24	14
25	15
26	15
27	16
28	16
29	17
30	17
31	18
32	18

LogiCORE™

Non-Symmetric, 32-Deep Time Skew Buffer

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts a serial data input on the SDI pin. This data is stored in a series of Serially Cascaded, 32-bit Synchronous RAM-based shift registers. The RAM's output data is registered and presented on the Q pins of the Macro. Each register's Q pin is internally cascaded to the following RAM's Data Input. The 32-bit RAM-based shift register supports 1- to 32-bit wide data storage per stage. The RESET pin is used to control the Bit-Width of the serial-data to be stored via the control logic.

The RAM address is controlled via an internal, free-running, 32-bit counter (CNT05RE). The counter is controlled by the CE (active high) and RESET inputs. The CNT05RE generates an output, 0 to TC, where TC is controlled via the RESET signal from the Control Logic, SDACL32. The addressing sequence of the cascaded registered RAM is structured to increment the address of each data-bit by one. That is, the bit-n is stored at an address ADDR+1 for each consecutive RAM. This is because each data sample is registered prior to writing it into the following cascaded RAM. The registration of the RAM data is important to maintain a synchronous system, as the RAM is synchronous write and asynchronous read. Hence, the RAM data output will change, to represent the data stored at a given address, as the address changes. The Truth Table for this function is shown in Table 1.

Table 1: Truth Table

WE (mode)	Data	CE	TC	C	CNT	Q[N-1,0]
X	X	0	X	X	No Change	No Change
0 (read)	X	1	CNT<TC+1	1/4	CNT+1	Data@CNT
1 (write)	Data	1	CNT<TC+1	1/4	CNT+1	Data@CNT
0 (read)	X	1	CNT=TC+1	1/4	0	Data@CNT
1 (write)	Data	1	CNT=TC+1	1/4	0	Data@CNT

Pinout

Signal names for the schematic symbol are shown in Figure 1 and Table 2.

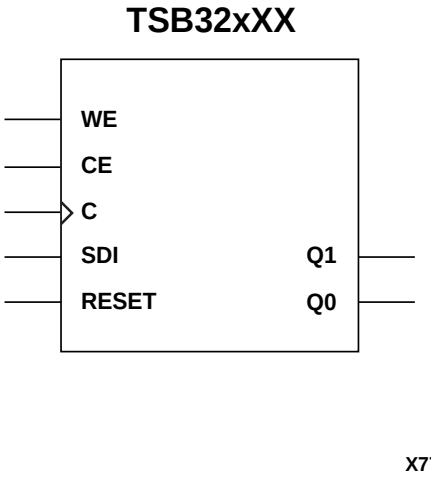


Figure 1: Core Schematic Symbol

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
WE	Input	WRITE ENABLE – Active high signal, Enables writing to the RAM in association with the internal address counter. This signal is required to hold a block a data for consecutive processing. (The WE pin can be externally tied to the CE if not used.)
CE	Input	CLOCK ENABLE – active high signal used to allow the transfer of data from the internal RAM to the registered output and the incrementing of the Address used to control the TSB RAM data transfer sequence. (Tie to VCC if not used.)
C	Input	CLOCK – clocks the output registers, TSB Counter, and internal synchronous RAM. CE must be active (high) to enable the RAM, Counter, and registers.
SDI	Input	SERIAL DATA INPUT – data presented on this pin is stored into the TSB when the CE (CLOCK ENABLE) is asserted (HIGH) and the rising edge of the C (CLOCK). SDI0 feeds the Input to the first RAM going up.
RESET	Input	TERMINAL COUNT RESET – Used to reset the internal RAM's address counter to zero. RESET should be set high at a count of N-2 for an N-bit sample during normal operation of cascaded data-sample.
Q[N-1, 0]	Output	REGISTERED OUTPUT DATA – the registered output of the RAM.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.

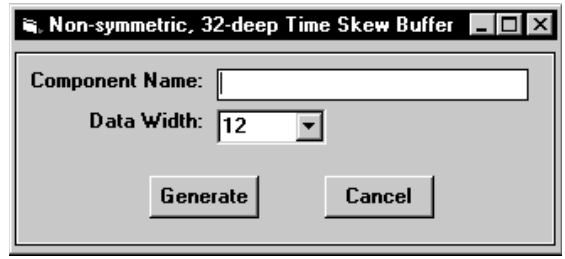


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 3 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 3: Bit Width versus CLB Count

Bit Width	CLB Count
2	5
3	6
4	7
5	8
6	9
7	10
8	11
9	12
10	13
11	14
12	15
13	16
14	17
15	18
16	19
17	20
18	21
19	22
20	23
21	24
22	25
23	26
24	27
25	28
26	29
27	30
28	31
29	32
30	33
31	34
32	35

LogiCORE™

Symmetric, 16-Deep Time Skew Buffer

February 8, 1998

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts a serial data input on the SDI pin. This data is stored in a series of Serially Cascaded, 16-bit Synchronous RAM-based shift registers. The RAM's output data is registered and presented on the Q pins of the Macro. Each register's Q pin is internally cascaded to the following RAM's Data Input. The 16-bit RAM-based shift register supports 1- to 16-bit wide data storage per stage. The RESET pin is used to control the Bit-Width of the serial-data to be stored via the control logic.

The RAM address is controlled via an internal, free-running, 16-bit counter (CNT04RE). The counter is controlled by the CE (active high) and RESET inputs. The CNT04RE generates an output, 0 to TC, where TC is controlled via the RESET signal from the Control Logic, SDACL16. The addressing sequence of the cascaded registered RAM is structured to increment the address of each data-bit by one. That is, the bit-n is stored at an address ADDR+1 for each consecutive RAM. This is because each data sample is registered prior to writing it into the following cascaded RAM. The registration of the RAM data is important to maintain a synchronous system, as the RAM is synchronous write and asynchronous read. Hence, the RAM data output will change, to represent the data stored at a given address, as the address changes. The Truth Table for this function is shown in Table 1.

Table 1: Truth Table

WE (mode)	Data	CE	TC	C	CNT	Q[N-1,0]
X	X	0	X	X	No Change	No Change
0 (read)	X	1	CNT<TC+1	≠	CNT+1	Data@CNT
1 (write)	Data	1	CNT<TC+1	≠	CNT+1	Data@CNT
0 (read)	X	1	CNT=TC+1	≠	0	Data@CNT
1 (write)	Data	1	CNT=TC+1	≠	0	Data@CNT

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 2.

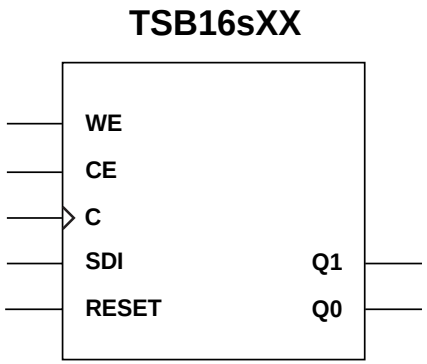


Figure 1: Core Schematic Symbol

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
WE	Input	WRITE ENABLE – Active high signal, Enables writing to the RAM in association with the internal address counter. This signal is required to hold a block a data for consecutive processing. (The WE pin can be externally tied to the CE if not used.)
CE	Input	CLOCK ENABLE – active high signal used to allow the transfer of data from the internal RAM to the registered output and the incrementing of the Address used to control the TSB RAM data transfer sequence. (Tie to VCC if not used.)
C	Input	CLOCK – clocks the output registers, TSB Counter, and internal synchronous RAM. CE must be active (high) to enable the RAM, Counter, and registers.
SDI	Input	SERIAL DATA INPUT – data presented on these pins is stored into the TSB when the CE (CLOCK ENABLE) is asserted (HIGH) and the rising edge of the C (CLOCK). SDI0 feeds the Input to the first RAM going up and SDIn feeds the Input to the first RAM going down to create an up-down data flow.
RESET	Input	TERMINAL COUNT RESET – Used to reset the internal RAM's address counter to zero. RESET should be set high at a count of N-2 for an N-bit sample during normal operation of cascaded data-sample.
Q[N-1, 0]	Output	REGISTERED OUTPUT DATA – the registered output of the RAM.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows.

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.

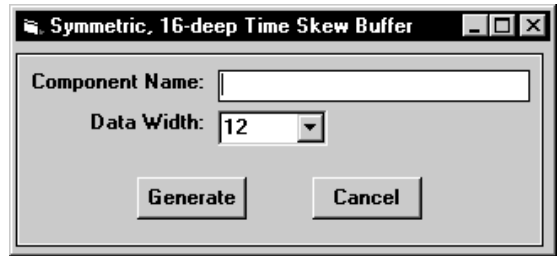


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 3 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 3: Bit Width versus CLB Count

Bit Width	CLB Count
2	3
3	4
4	4
5	5
6	5
7	6
8	6
9	7
10	7
11	8
12	8
13	9
14	9
15	10
16	10
17	11
18	11
19	12
20	12
21	13
22	13
23	14
24	14
25	15
26	15
27	16
28	16
29	17
30	17
31	18
32	18

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This module generates an object that forces a bus to a fixed value.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

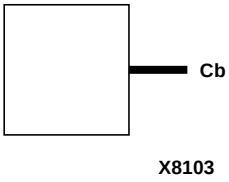


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Cb	Output	CONSTANT – the fixed value applied to a bus

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-31.
- **Constant Value**: Enter a number.
- **Radix**: Select the radix of the constant: Hex or Decimal.
- **Sign**: If the radix is decimal, select the sign of the constant value: Signed or Unsigned. If the radix is hex, the constant is always signed.

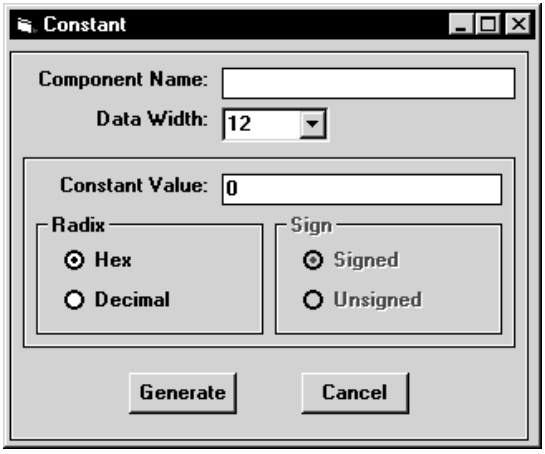


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

The constant is represented by VCC GND, so it does not use any resources.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro is a two-input multiplexer. One input is chosen by the Select line and directed to the output data bus.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

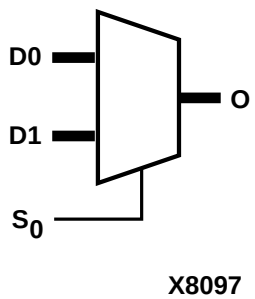


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D0	Input	First INPUT DATA BUS
D1	Input	Second INPUT DATA BUS
S_0	Input	SELECT INPUT - chooses which input data bus is directed to the output bus
O	Output	OUTPUT DATA BUS

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.
- **Create RPM**: When checked, a Relational Placed Macro is created.

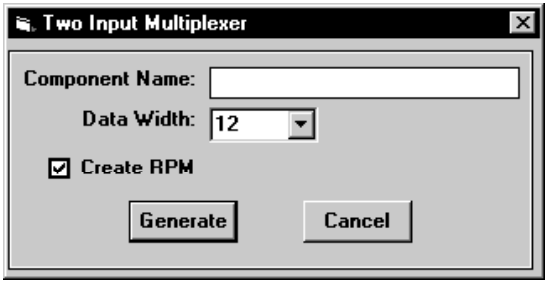


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	1
3	2
4	2
5	3
6	3
7	4
8	4
9	5
10	5
11	6
12	6
13	7
14	7
15	8
16	8
17	9
18	9
19	10
20	10
21	11
22	11
23	12
24	12
25	13
26	13
27	14
28	14
29	15
30	15
31	16
32	16

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Three Input MUX

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro is a three-input multiplexer. One input is chosen by the Select lines and directed to the output data bus.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

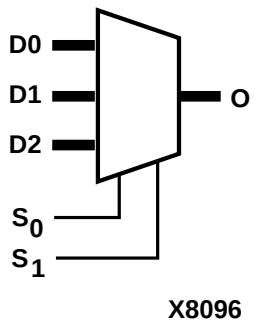


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D0	Input	First INPUT DATA BUS
D1	Input	Second INPUT DATA BUS
D2	Input	Third INPUT DATA BUS
S_0, S_1	Input	SELECT INPUT - chooses which input data bus is directed to the output bus
O	Output	OUTPUT DATA BUS

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.
- **Create RPM**: When checked, a Relational Placed Macro is created.

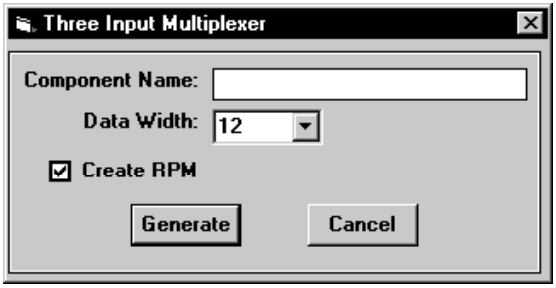


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro is a four-input multiplexer.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

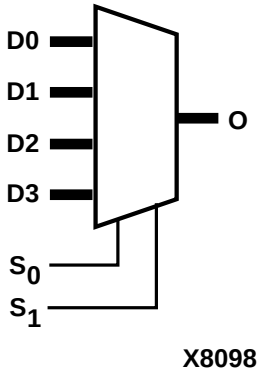


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D0	Input	First INPUT DATA BUS
D1	Input	Second INPUT DATA BUS
D2	Input	Third INPUT DATA BUS
D3	Input	Fourth INPUT DATA BUS
S_0, S_1	Input	SELECT INPUT - chooses which input data bus is directed to the output bus
O	Output	OUTPUT DATA BUS

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.
- **Create RPM**: When checked, a Relational Placed Macro is created.

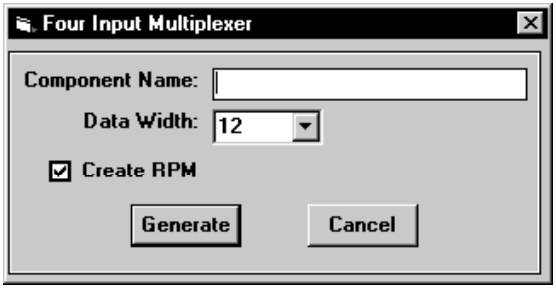


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts an N-bit value and loads it into a shift register when the L (LOAD) and CE (CLOCK ENABLE) signals are asserted. With the L signal not asserted and the CE signal asserted, SDI data is shifted into the MSB register while the registered data is shifted out serially, LSB to MSB, on each rising edge of the C (CLOCK) signal. The Truth Table for this function is shown in Table 1.

Table 1: Truth Table

LOAD	CE	C	SDO	PO[n:0]
X	0	≠	No-Change	No-Change
0	1	≠	PO[0]	SDI,PO[n:1]
1	1	≠	PI[0]	PI[n:0]

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 2.

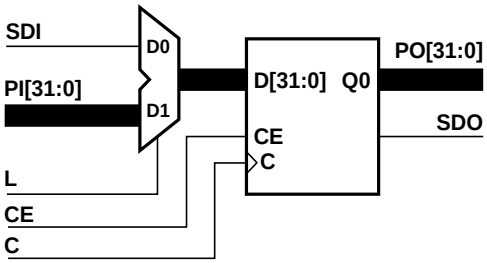


Figure 1: Core Schematic Symbol

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
PI[N-1, 0]	Input	PARALLEL DATA INPUT – Value is registered when the L (LOAD) and CE (CLOCK ENABLE) signals are asserted. It is logically shifted to the next cascaded register, from LSB to MSB, when L is not asserted and CE is asserted.
SDI	Input	SERIAL DATA IN – Serial Data Input. Connects the MSB of the Input data, PI(N-1) to SDI to Sign Extend, signed Input Data.
L	Input	LOAD – When this signal is asserted, the data on PI[N-1, 0] is loaded directly to the output register. The CE signal must be asserted for the load operation.
CE	Input	CLOCK ENABLE – active high signal used to enable the registration of data for both shift and load modes of operation.
C	Input	CLOCK – clocks the output register. CE must be active (high) to enable the register.

Signal	Signal Direction	Description
SDO	Output	SERIAL DATA OUTPUT – Serial, LSB to MSB output bit of shift parallel data.
PO[N-1, 0]	Output	PARALLEL DATA OUTPUT – Parallel Data output bits of shifted parallel data. This output can be used for serial to parallel conversion in association with the SDI.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.
- **Create RPM**: When checked, a Relational Placed Macro is created.

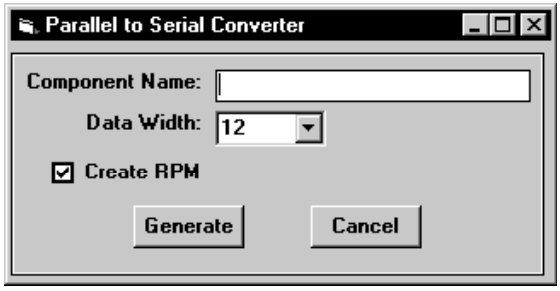


Figure 2: CORE Generator Dialog Box

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Core Resource Utilization

Table 3 shows the number of CLBs required for each available bit width.

Table 3: Bit Width versus CLB Count

Bit Width	CLB Count
2	1
3	2
4	2
5	3
6	3
7	4
8	4
9	5
10	5
11	6
12	6
13	7
14	7
15	8
16	8
17	9
18	9
19	10
20	10
21	11
22	11
23	12
24	12
25	13
26	13
27	14
28	14
29	15
30	15
31	16
32	16

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts an N-bit value and stores it in the register.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and Table 1.

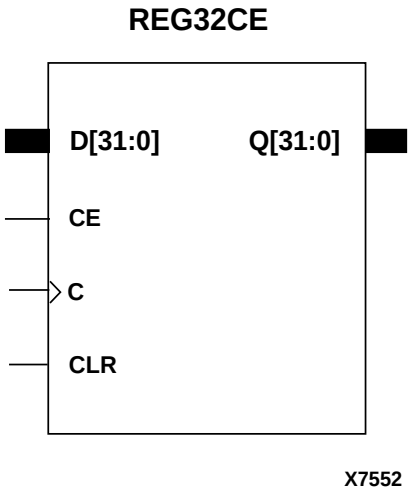


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D[N-1, 0]	Input	DATA INPUT – input that is stored into the register when CE is asserted.
CE	Input	CLOCK ENABLE – active high signal used to allow the transfer of data from the internal registers input to output.
C	Input	CLOCK – clocks the output register. CE must be active (high) to enable the register.
CLR	Input	ASYNCHRONOUS CLEAR – clears the register. Does not require the Clock or CE signal.
Q[N-1, 0]	Output	REGISTERED OUTPUT DATA – the registered output.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.

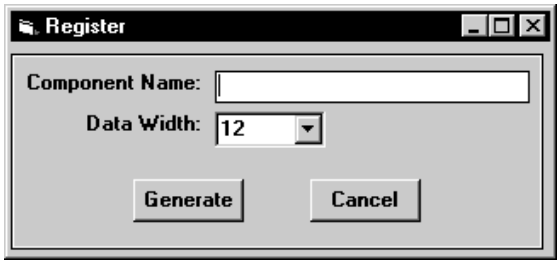


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	1
3	2
4	2
5	3
6	3
7	4
8	4
9	5
10	5
11	6
12	6
13	7
14	7
15	8
16	8
17	9
18	9
19	10
20	10
21	11
22	11
23	12
24	12
25	13
26	13
27	14
28	14
29	15
30	15
31	16
32	16

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

In 1's complement mode this macro accepts an N-bit value and inverts the data on a bit-by-bit basis when the INV signal is asserted (HIGH). In 2's complement mode this macro accepts an N-bit value and calculates the corresponding 2's complement value when the INV signal is asserted (HIGH). The data is passed through unchanged when INV is not asserted. The inverted or buffered data is then stored in the register.

The Truth Table for this function is shown in Table 1.

Table 1: Truth Table

INV	CE	C	Q[n]
X	X	X	0
X	0	↑	No Change
0	1	↑	A[n]
1	1	≠	INV(A[n])

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 2.

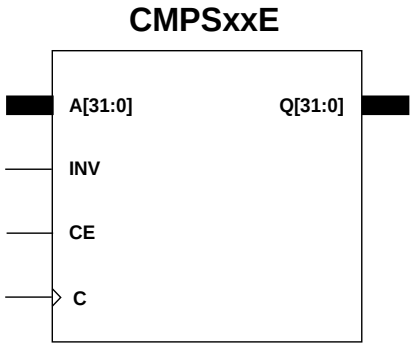


Figure 1: Core Schematic Symbol

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
A[N-1, 0]	Input	A data input – value is inverted or 2's complemented when the INV signal is asserted.
INV	Input	Invert data signal – the incoming data on A is bit-wise inverted or 2's complemented when this signal is asserted.
CE	Input	Clock enable – active high signal used to enable the transfer of data into the register.
C	Input	Clock – for the register.
Q[N-1, 0]	Output	The registered output of the inverter.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.

- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32. The output size will be the same as the input width.
- **Create RPM**: When checked, a Relational Placed Macro is created. This parameter is enabled only when the 1's Complementer type is selected.
- **Complementer Type**: Select 1's Complementer or 2's Complementer.

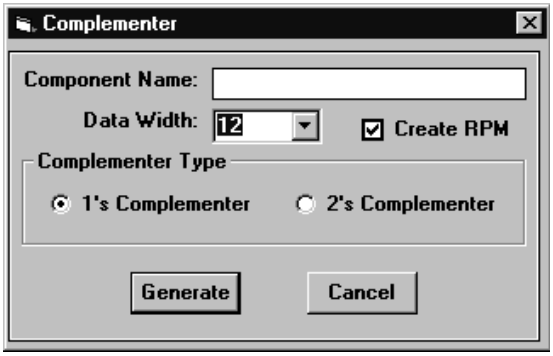


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 3 shows the number of CLBs required for each available bit width.

Ordering information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 3: Bit Width versus CLB Count

Bit Width	CLB Count
2	1
3	2
4	2
5	3
6	3
7	4
8	4
9	5
10	5
11	6
12	6
13	7
14	7
15	8
16	8
17	9
18	9
19	10
20	10
21	11
22	11
23	12
24	12
25	13
26	13
27	14
28	14
29	15
30	15
31	16
32	16

LogiCORE™ Scaled By 1/2 Accumulator

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts a two's complement input value, N-bits wide, sign extends it to N+1 bits, and adds the registered and scaled feedback data. The feedback data is sign extended two bits and shifted right by one bit. The shifting of the feedback is equivalent to dividing the sum by 2.

The LOAD Pin, L, is used to load the initial value, B[n-1,0], directly into the accumulator's output register, S[n,0].

A typical use is in a bit-wise multiplier. Data is processed from LSB to MSB and each successive bit has twice the binary weight as its predecessor. The LSB of the Output can be discarded for a single precision output, or registered in a serial-to-parallel register bank to maintain a double precision output. The Truth Table for this function is shown in Table 1.

Table 1: Truth Table

LOAD	CE	C	Feedback	S[n]
X	0	≠	No Change	No Change
0	1	≠	1/2 * S[n]	1/2 * S[n-1] + B[n,0]
1	1	≠	1/2 * S[n]	B[n,0]

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 2.

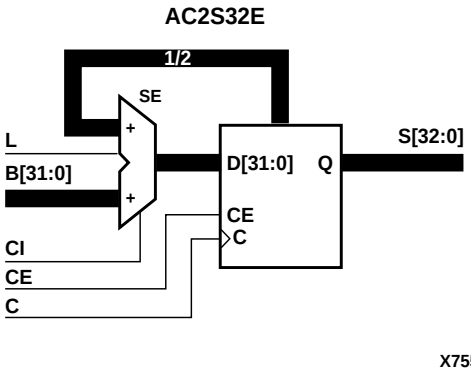


Figure 1: Core Schematic Symbol

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
L	Input	LOAD – active high signal bypasses the adder and loads the B data directly into the output register.
B[N-1, 0]	Input	B data input – data is added to 1/2 the binary weight of the registered output sum.
CI	Input	CARRY IN – Initial carry logic input. Set this to GND if not used.
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data into the output register.
C	Input	CLOCK – clocks the output register. CE must be active (high) to enable the register.
S[N, 0]	Output	SUM DATA OUTPUT – the registered output of the adder.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32. The output size will be the input width plus one.

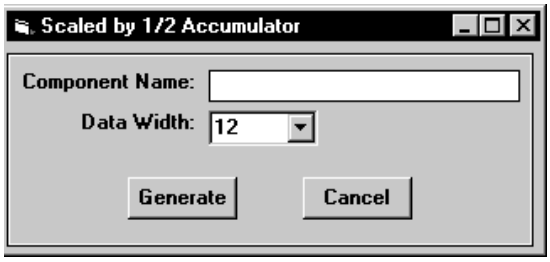


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 3 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 3: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	2
4	3
5	3
6	4
7	4
8	5
9	5
10	6
11	6
12	7
13	7
14	8
15	8
16	9
17	9
18	10
19	10
20	11
21	11
22	12
23	12
24	13
25	13
26	14
27	14
28	15
29	15
30	16
31	16
32	17

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

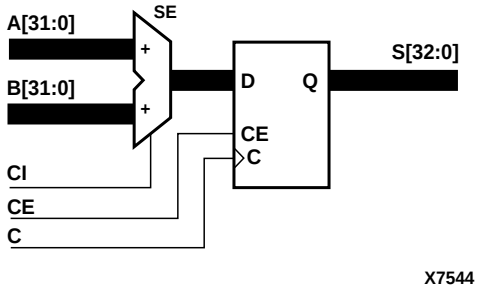


Figure 1: Core Schematic Symbol

Features

- Two data bus inputs: 2 to 32 bits wide
- Supports both 2's complement signed and magnitude-only unsigned data
- Drop-in modules for the XC4000E, EX, and XL families
- Registered output
- Clock Enable for output register
- Asynchronous Clear for output register
- Uses Fast Carry logic for high speed
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Registered Adder module accepts two input buses, **A** and **B**, adds them, and registers the sum, **S**. The input data buses can be in either 2's complement signed or magnitude-only unsigned numbers.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A[N-1, 0]	Input	A data input – value is added to B data
B[N-1, 0]	Input	B data input – value is added to A data
CI	Input	CARRY IN – initial carry logic input. Set this to GND if not used
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data from the internal registers to output
C	Input	CLOCK – clocks the output register
S[N, 0]	Output	SUM DATA OUTPUT – the registered output of the adder

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32. The same data width is applied to both the A and B inputs. The output size is automatically set to the input width plus one.
- **Sign**: Select Signed or Unsigned.

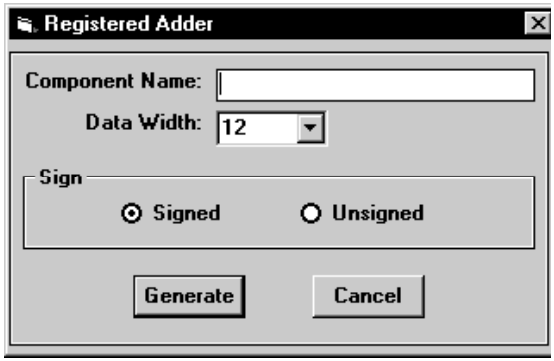


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	2
4	3
5	3
6	4
7	4
8	5
9	5
10	6
11	6
12	7
13	7
14	8
15	8
16	9
17	9
18	10
19	10
20	11
21	11
22	12
23	12
24	13
25	13
26	14
27	14
28	15
29	15
30	16
31	16
32	17

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Two data bus inputs: 2 to 32 bits wide
- Supports both 2's complement signed and magnitude-only unsigned data
- Registered output
- Load feature to directly load the output register
- Clock Enable for output register
- Asynchronous Clear for output register
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Registered Loadable Adder module accepts two input buses, **A** and **B**, adds them, and registers the sum, **S**. The input data buses can be in either 2's complement signed or magnitude-only unsigned numbers.

When the Load (**L**) and Clock Enable (**CE**) signals are asserted, the B input data bus is loaded directly into the output register on the rising edge of the Clock.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

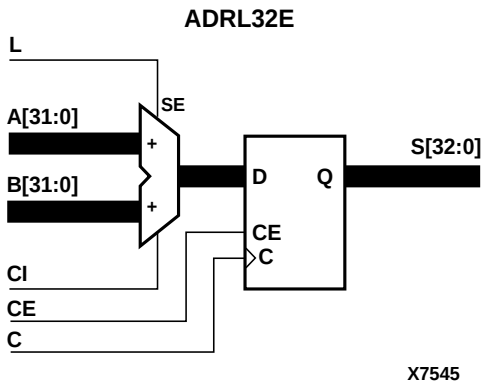


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
L	Input	LOAD – active high signal bypasses the adder and directly loads the B data into the output register.
A[N-1, 0]	Input	A data input – value is added to B data.
B[N-1, 0]	Input	B data input – value is added to A data.
CI	Input	CARRY IN – Initial carry logic input. Set this to GND if not used.
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data from the internal registers to output.
C	Input	CLOCK – clocks the output register.
S[N, 0]	Output	SUM DATA OUTPUT – the registered output of the adder

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32. The same data width is applied to both the A and B inputs. The output size is automatically set to the input width plus one.
- **Sign**: Select Signed or Unsigned.

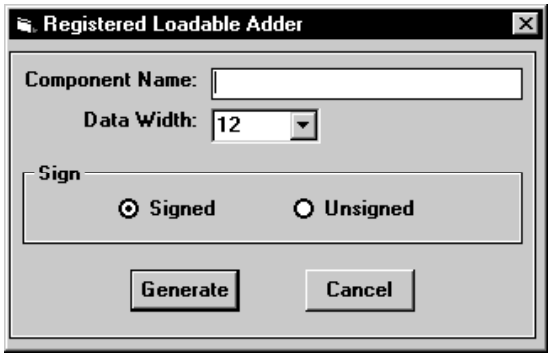


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	2
4	3
5	3
6	4
7	4
8	5
9	5
10	6
11	6
12	7
13	7
14	8
15	8
16	9
17	9
18	10
19	10
20	11
21	11
22	12
23	12
24	13
25	13
26	14
27	14
28	15
29	15
30	16
31	16
32	17

LogiCORE™ Registered Scaled Adder

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

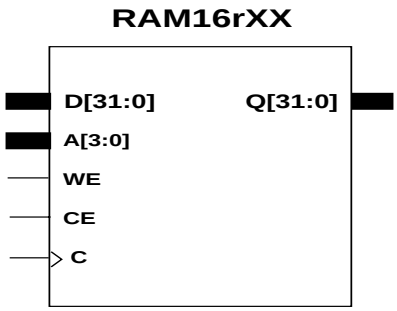


Figure 1: Core Schematic Symbol

Features

- Two data bus inputs: 2 to 31 bits wide
- The size of the output data bus is adjustable to the input bus width plus one or more, up to 32 bits
- Supports 2's complement signed data
- Registered output
- Clock Enable for output register
- Asynchronous Clear for output register
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Registered Scaled Adder module accepts two input buses, **A** and **B**, adds them, and registers the sum, **S**. The input data buses are in 2's complement signed numbers. The output of the module can be scaled to larger output values than are required by the input values bus width to insure proper overflow handling.

A typical use is in a multiplier adder tree. Data is processed, such that each partial sum is scaled and added to each successive partial sum which has twice the binary weight as its predecessor.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A[N-1, 0]	Input	A data input – value is added to B data divided by k.
B[N-1, 0]	Input	B data input – value divided by k is added to A data.
CI	Input	CARRY IN – Initial carry logic input to A data. The Carry-In is scaled to the bit weighting of the A Data LSB, which equates to a shift of k bits for the B data. Set this input to GND if not used.
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data from the internal registers to output.
C	Input	CLOCK – clocks the output register.
S[N, 0]	Output	SUM DATA OUTPUT – the registered output of the adder.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Input Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-31. The same data width is applied to both the A and B inputs.
- **Output Data Width**: Select an output bit width from the pull-down menu. The valid range is 3-32. The Scaling Function is:

  K = 2**(Output_width - Input_width - 1)

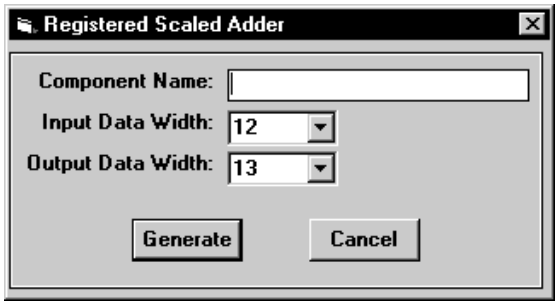


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

The number of CLBs required depends on both the Input and Output data widths. The equation is:

# of CLBs = floor((Input_width+2)/2) + 1 +
floor((Output_width-Input_width-2)/2)

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

LogiCORE™ Registered Serial Adder

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts 2 to 32 1-bit data pairs and performs a bit-wise add on each pair. The incoming data is composed of bit-serial data, LSB to MSB. When the CLEAR CARRY-IN (CLRCI) signal is not asserted, the input data A(n) is added to B(n) along with the Carry-Out from the previous addition. The resulting Sum, SD(n), is registered and the Carry-Out is saved internally. When the CLRCI is asserted, the input data A(n) is added to B(n), with the Carry-Out from the previous addition set to zero. The resulting Sum, SD(n), is registered.

For example, the macro shown in Figure 1 has 8 input bits, four each in bus A and bus B. Internal to the macro, input A(n) is added to B(n) which produces a Sum SD(n) and an internal carry-out signal used as a ripple carry in the next addition. To add two N-bit words requires N+1 clock cycles. Processing from LSB to MSB produces N-Carry-Out signals. To maintain full precision in the final result, the MSB must be processed twice to sign-extend the input data for each input pair A(n) and B(n), for signed data. It is during the first addition on the Least Significant Bits of data that the Carry-Out is cleared to initiate the Carry for the next set of serial data bits to be added.

The data is added from LSB to MSB. The CLRCI signal must be asserted during the LSB add.

Table 1: Truth Table

CLRCI	CE	A[n], B[n]	C	SD[n]
X	0	X	≠	No Change
0	1	A[n], B[n]	≠	A[n]+B[n]+CO[n]
1	1	A[n], B[n]	≠	A[n]+B[n]

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 2.

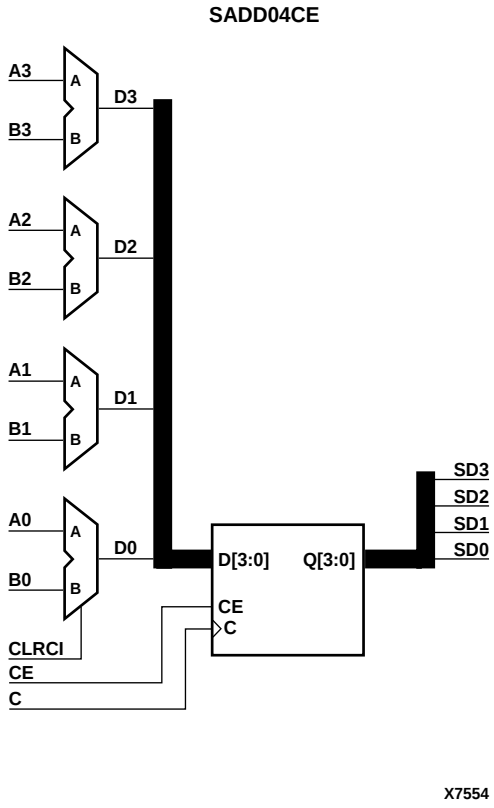


Figure 1: Core Schematic Symbol

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
A[N]	Input	A data input – value is added to B data.
B[N]	Input	B data input – value is added to A data
CLRCI	Input	CLEAR CARRY IN – Initializes the internal ripple carry logic to zero. This is required for the first (LSB) bit add.
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of the summed data into the output register.
C	Input	CLOCK – clocks the output register.
SD[N]	Output	SUM DATA REGISTERED OUTPUT – the registered output of the sum resulting from the addition of A[n]+B[n]+CO[n].

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32. The same data width is applied to both the A and B inputs. The output size is automatically set to the input width plus one.
- **Create RPM**: When checked, a Relational Placed Macro is created.

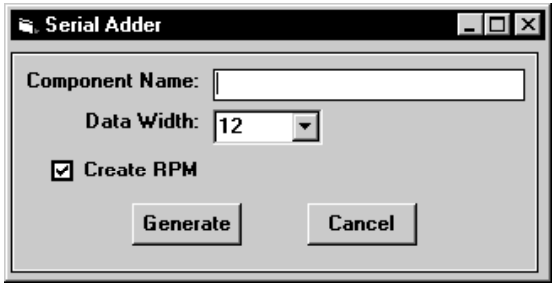


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 3 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 3: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	2
4	3
5	3
6	4
7	4
8	5
9	5
10	6
11	6
12	7
13	7
14	8
15	8
16	9
17	9
18	10
19	10
20	11
21	11
22	12
23	12
24	13
25	13
26	14
27	14
28	15
29	15
30	16
31	16
32	17

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Multiply-free response computation
- Performance limited by parallel adder carry propagation delay – Fast Carry logic used for high speed
- Fractional 2's complement data format
- Input data widths from 2 to 31 bits
- Output data widths from 2 to 32 bits
- Single delay feedback
- Registered output
- Can be used as a general purpose parameterized accumulator for building a parallel MAC or an NCO
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Ideal Integrator transfer function, $H_I(z)=1/(1-z^{-1})$, has a pole at f=0. It is ideal in the sense that its response to an impulse is a step function, and its response to a step function (i.e., a dc input) is a linear ramp whose slope is proportional to step size. The Integrator is highly susceptible to overflow and adequate word growth of the accumulator register must be provided.

Integrators can be cascaded by simply connecting the output of one stage to the input of the next. Interstage scaling and truncation of data can alleviate overflow problems.

High order, multiplier-free decimation and interpolation filters can be implemented with cascaded stages of Ideal Integrators and Comb Filters [1].

With periodic clearing of the accumulator register, an “integrate and dump” accumulator is realized. This versatile device can serve as a low-pass or averaging filter per se, as the output stage of a polyphase filter, or, in conjunction with a single multiplier, as a minimal FIR filter [2].

1. E. B. Hogenauer, *“An Economical Class of Digital Filters for Decimation and Interpolation,”* IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. ASSP-29, April 1981, pp. 155-162.

2. D. F. Elliott, *“Handbook of Digital Signal Processing,”* 1987, Academic Press, pp. 245-252.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

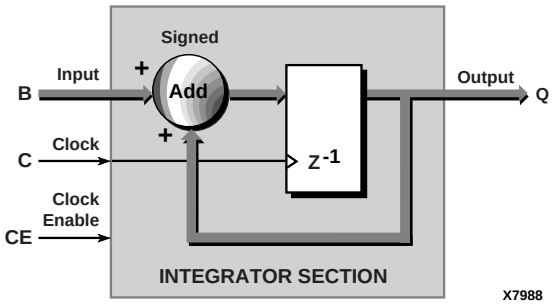


Figure 1: Block Diagram

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
B	Input	Input data N bits wide
C	Input	Clock – active on rising edge
CE	Input	Clock Enable – active high
L	Input	LOAD – active high signal bypasses the adder and directly loads the B data into the output register.
S	Output	Output data M bits wide

CORE Generator Parameters

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe.

VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Input Data Width**: Select an input bit width from the pull-down menu. The valid range is 2 to 31.
- **Output Data Width**: Select an output bit width from the pull-down menu. The valid range is 2 to 32.

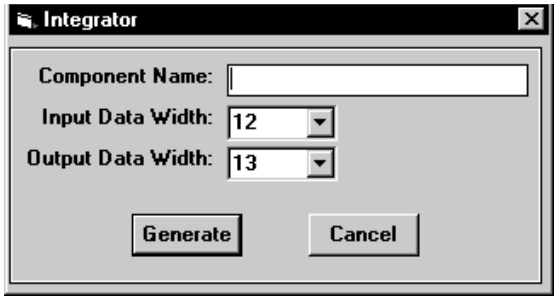


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for some available bit widths.

Table 2: Bit Width versus CLB Count

Output Bit Width	CLB Count
4	3
8	5
12	7
16	9
20	11

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

LogiCORE™ Constant Coefficient Multiplier

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Multiplies a variable **A** times a constant **B**
- The **A** value can range from 4 to 32 bits
- The **B** value can range from 2 to 26 bits
- Independent **A** and **B** word size
- Independent control on signed and unsigned for each operand
- Full precision output
- Supports 2's complement signed and unsigned magnitude numbers
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This parameterized module multiplies an N-bit wide variable times an M-bit fixed coefficient and produces an N+M bit result. The coefficient and multiplication tables are stored in distributed ROM-based look-up tables (LUTs), taking advantage of the FPGA look-up table architecture. It is an efficient, high speed, parallel implementation.

The variable and constant widths can be set independently, allowing the user to multiply different word widths together. The module automatically adjusts the signed and unsigned data to properly handle different sizes.

The variable and constant can be independently selected to be 2's complement signed or unsigned magnitude-only values. Note that the module will automatically handle a signed times an unsigned number, two signed numbers, or two unsigned numbers.

The Constant Coefficient Multiplier (KCM) is used where the value of an incoming variable needs to be multiplied by a number that does not change, a constant. This is often the case in scaling functions, FIR filters, IIR filters, etc.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

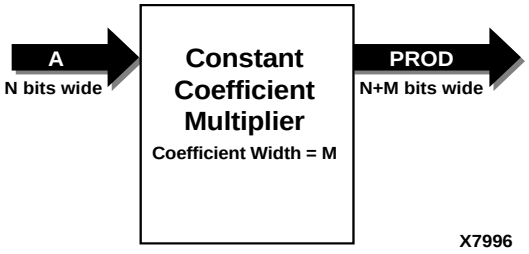


Figure 1: Block Diagram

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A	Input	Parallel Data In – N-bits wide
PROD	Output	Parallel Data Out – N+M bits wide

CORE Generator Parameters

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL or Verilog instantiation code and a schematic symbol are created along with the netlist for the design.

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Variable A Data Width**: Select an input bit width from the pull-down menu for the variable width. The valid range is 4 to 32.
- **Variable A Sign**: Set the sign of the variable to Signed or Unsigned.
- **Constant B Data Width**: Select an input bit width from the pull-down menu for the constant width. The valid range is 2 to 26.

- **Constant**: Enter the value of the desired constant.
- **Constant B Sign**: Set the sign of the constant to Signed or Unsigned.
- **Radix**: Hexadecimal or Decimal representation of the constant.

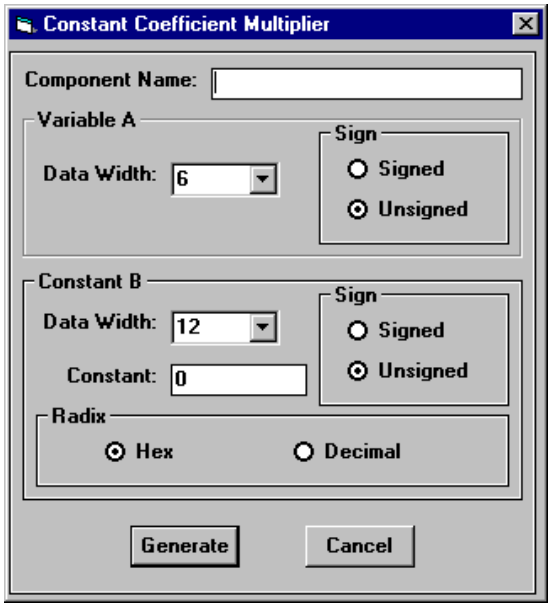


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for some available bit widths.

Table 2: Bit Width versus CLB Count

A Bit Width	B Bit Width	CLB Count
4	2	3
4	17	11
4	26	15
5	5	10
6	6	14
7	7	17
8	8	19
9	10	31
10	10	39
12	8	32
13	13	62
14	13	65
15	15	72
16	16	75
17	11	69
20	20	114
22	20	126
24	24	163
27	26	207
28	26	207
32	2	62
32	26	242

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

LogiCORE™ Constant Coefficient Multiplier (Pipelined)

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Multiplies a variable **A** times a constant **B**
- The **A** value can range from 4 to 32 bits
- The **B** value can range from 2 to 26 bits
- Independent **A** and **B** word size
- Independent control on signed and unsigned for each operand
- Full precision output
- Supports 2's complement signed and unsigned magnitude nubmers
- The inputs are not registered, but the outputs are registered
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This parameterized module multiplies an N-bit wide variable times an M-bit fixed coefficient and produces an N+M bit result. The coefficient and multiplication tables are stored in distributed ROM-based look-up tables (LUTs), taking advantage of the FPGA look-up table architecture. It is an efficient, high speed, parallel implementation.

The variable and constant widths can be set independently, allowing the user to multiply different word widths together. The module automatically adjusts the signed and unsigned data to properly handle different sizes.

The variable and constant can be independently selected to be 2's complement signed or unsigned magnitude-only values. Note that the module will automatically handle a signed times an unsigned number, two signed numbers, or two unsigned numbers.

The Constant Coefficient Multiplier (KCM) is used where the value of an incoming variable needs to be multiplied by a number that does not change, a constant. This is often the case in scaling functions, FIR filters, IIR filters, etc.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

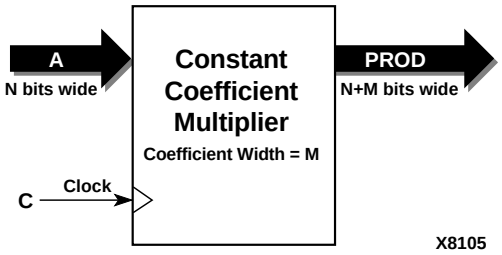


Figure 1: Block Diagram

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A	Input	Parallel Data In – N-bits wide
C	Input	Clock – active on rising edge
PROD	Output	Parallel Data Out – N+M bits wide

CORE Generator Parameters

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL or Verilog instantiation code and a schematic symbol are created along with the netlist for the design.

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Variable A Data Width**: Select an input bit width from the pull-down menu for the variable width. The valid range is 4 to 32.
- **Variable A Sign**: Set the sign of the variable to Signed or Unsigned.

- **Constant B Data Width**: Select an input bit width from the pull-down menu for the constant width. The valid range is 2 to 26.
- **Constant**: Enter the value of the desired constant.
- **Constant B Sign**: Set the sign of the constant to Signed or Unsigned.
- **Radix**: Hexadecimal or Decimal representation of the constant.

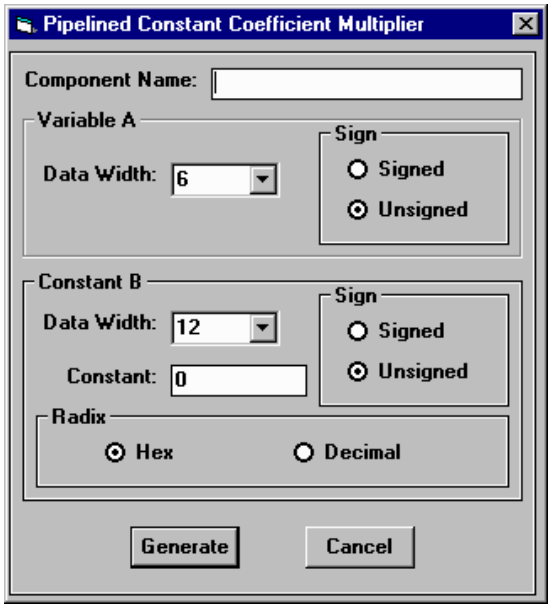


Figure 2: CORE Generator Dialog Box

Multiplier Latency

The total latency (number of clocks required to get the first output) is a function of the width of variable **A**. Table 2 shows the latency for the possible bit widths of **A**.

Table 2: Multiplier Latency

A Data Width	Latency (# Clocks)
4 to 5 bits	1
6 to 10 bits	2
11 to 17 bits	3
18 to 32 bits	4

Core Resource Utilization

Table 3 shows the number of CLBs required for some available bit widths.

Table 3: Bit Width versus CLB Count

A Bit Width	B Bit Width	CLB Count
4	2	3
4	17	11
4	26	15
5	5	10
6	6	16
7	7	19
8	8	21
9	10	34
10	10	42
12	8	44
13	13	68
14	13	73
15	15	80
16	16	83
17	11	77
20	20	154
22	20	169
24	24	197
27	26	244
28	26	244
32	2	86
32	26	266

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Parallel Multipliers with parameterizable data widths
- Area optimized design
- Speed almost equal to our performance optimized version, but with fewer CLBs
- Independently adjustable input variable widths from 6 to 32 bits
- Full precision outputs
- Two's complement and magnitude data formats
- Registered inputs and outputs
- Predictable timing and reduced routing times with pre-designed drop-in modules
- Supports Foundation and Viewlogic as well as VHDL and Verilog design methodologies
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This parameterized module is a high-speed parallel implementation that multiplies an N-bit wide variable times an M-bit variable and produces an N+M bit result.

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

An area-efficient, high-speed algorithm is used to give an efficient, tightly packed design. Each stage is pipelined for maximum performance.

In addition to this area-efficient design, the CORE Generator contains a performance optimized design that yields a 10% to 20% increase in speed, but uses more CLB resources.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

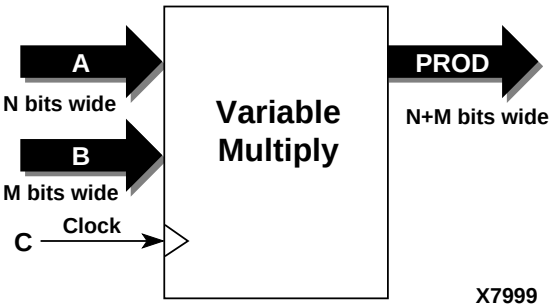


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A	Input	Parallel Data In, N-bits wide
B	Input	Parallel Data In, M-bits wide
C	Input	Clock, processes data on the low to high transition
PROD	Output	Parallel Data Out, N + M bits wide

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Width A**: Select an input bit width from the pull-down menu for the input A. The valid range is 6 to 32.
- **Width B:** Select an input bit width from the pull-down menu for the input B. The valid range is 6 to 32.
- **Signed or Unsigned:** Representation of the variables.

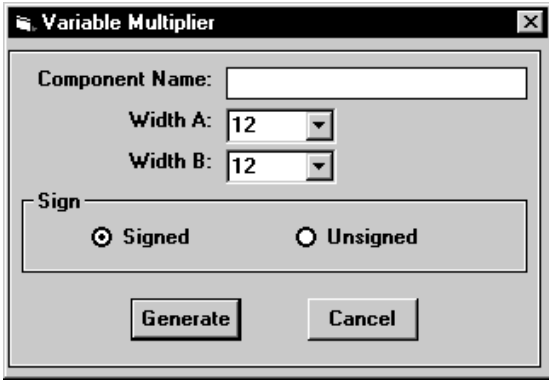


Figure 2: CORE Generator Dialog Box

Module Layout

Figure 3 shows the organization of CLBs in a 12x12 multiplier. This module fits in an array of 132 CLBs organized as a matrix of 12 rows by 11 columns. The module uses 122 CLBs. The white squares represent the unused CLBs, demonstrating the efficiency with which the matrix is utilized. The unused CLBs are available for use in other parts of the system design.

Data enters the module from the left side (A) and the bottom left side (B). The resulting product is available on the right side (O). If all 24 outputs are not connected to the next stage of the design, the Xilinx implementation software eliminates all unused CLBs.

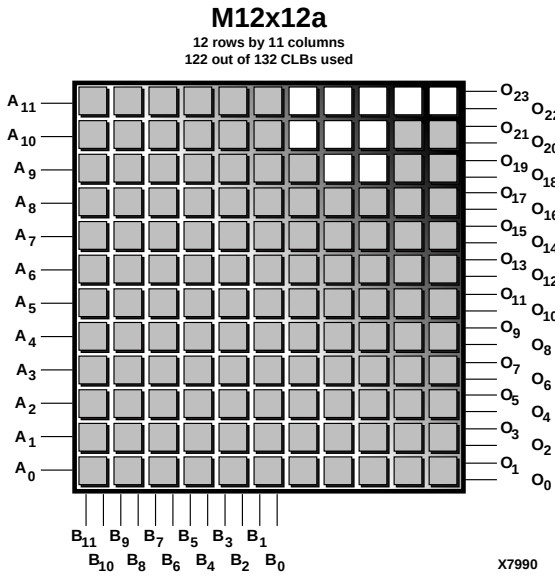


Figure 3: CLB Organization for Module

All of the area efficient series of modules can be built in a rectangular format. For example, an 8-bit by 12-bit multiplier will fit in a rectangular matrix of 8 rows by 11 columns.

Figure 4 shows a 12x12 multiplier in place in an XC4005E chip. The 4005E contains 196 CLBs, of which 122 are used by the 12x12 multiplier and 74 are available for other functions.

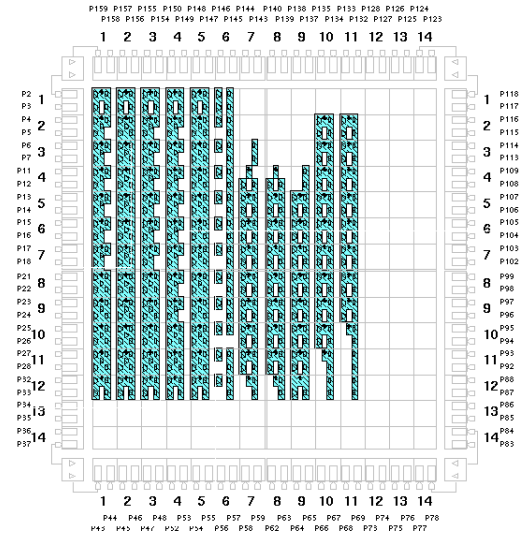


Figure 4: Xilinx Floorplanner View of XC4005E with a 12x12 multiplier

The module contains both relative placement information and timing constraints, allowing the Xilinx place and route software to consistently produce the same timing specifications. CLB relative placement information (R-LOCs) speeds up the routing place and route time by a factor of 10, thus permitting rapid design changes.

Multiplier Latency

Data is buffered on the input and output of the multiplier cores. The total latency (number of clocks required to get the first output) is a function of the width of the **B** variable only.

Table 2: Multiplier Latency

B Data Width	Latency (# Clocks)
6 to 8 bits	4
9 to 16 bits	5
17 to 31 bits	6

Core Resource Utilization

Tables 3 to 8 show the number of CLBs required for some of the available bit widths. The maximum speed is for XC4000E-1 devices.

All the variable multipliers are built to fit in a rectangular or square matrix of N rows by M (or M-1) columns.

Table 3: Bit Width versus CLB Count for A = 8 bits

A Bit Width	B Bit Width	CLB Count	4000E-1 MHz	4000E-3 MHz
8	6	44	86	54
8	8	54	83	53
8	10	86	86	53
8	12	96	78	46
8	14	111	77	45
8	16	124	74	44
8	20	180	71	43
8	24	211	68	41
8	32	282	60	35

Table 4: Bit Width versus CLB Count for A = 10 bits

A Bit Width	B Bit Width	CLB Count	4000E-1 MHz	4000E-3 MHz
10	6	53	81	51
10	8	65	77	49
10	10	100	78	47
10	12	113	77	47
10	14	133	70	42
10	16	146	72	44
10	20	209	65	39
10	24	238	62	38
10	32	330	57	34

Table 5: Bit Width versus CLB Count for A = 12 bits

A Bit Width	B Bit Width	CLB Count	4000E-1 MHz	4000E-3 MHz
12	6	61	75	47
12	8	75	72	46
12	10	116	70	42
12	12	130	67	41
12	14	153	65	40
12	16	167	65	39
12	20	238	63	38
12	24	274	59	36
12	32	371	51	31

Table 6: Bit Width versus CLB Count for A = 16 bits

A Bit Width	B Bit Width	CLB Count	4000E-1 MHz	4000E-3 MHz
16	6	79	64	41
16	8	98	63	39
16	10	148	61	37
16	12	165	59	36
16	14	192	56	35
16	16	213	58	35
16	20	299	56	34
16	24	349	51	33
16	32	473	49	31

Table 7: Bit Width versus CLB Count for A = 20 and 24 bits

A Bit Width	B Bit Width	CLB Count	4000E-1 MHz	4000E-3 MHz
20	6	98	56	36
20	8	120	55	35
20	12	202	55	34
20	16	259	50	31
20	20	358	50	30
24	6	116	50	32
24	8	142	50	32
24	12	238	46	28
24	16	305	45	28
24	20	425	45	28

Table 8: Bit Width versus CLB Count for Other Widths of A

A Bit Width	B Bit Width	CLB Count	4000E-1 MHz	4000E-3 MHz
6	6	37	96	60
9	9	94	81	49
11	11	121	74	45
13	13	161	61	38
14	14	173	62	37
15	15	203	58	36
17	17	293	55	33
18	18	308	53	32
19	19	343	52	32
32	6	151	41	27
32	8	187	41	27
32	12	309	41	26

Multiplier Trade-offs

Two different implementations of parallel multipliers trade area for speed. The area efficient designs consume about one-fourth less CLB resources than the high speed designs in the 4000E family.

The additional routing resources in the 4000EX and 4000XL families will increase the performance for the area efficient designs. In addition, both structures will benefit from the overall performance increase derived from the 4000XL .35 micron process technology.

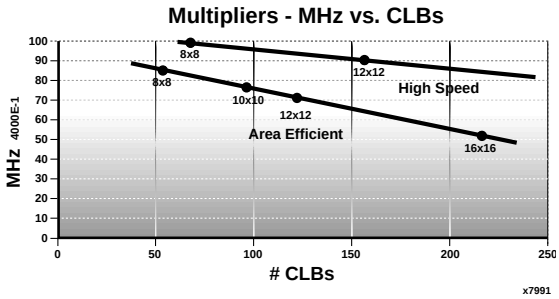


Figure 5: Trade-offs of Area Versus Speed Optimization for Multipliers

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

LogiCORE™

Parallel Multipliers – Performance Optimized

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- High performance compact implementation
- Drop-in modules for the XC4000E, EX, and XL families
- Two variable operands, 2's complement arithmetic
- Signed data, full precision outputs
- Registered inputs and outputs
- Pre-designed modules with relative placement gives predictable timing
- Fast place and route times
- Supported in Viewlogic and Foundation schematic entry and simulation tools
- VHDL and Verilog instantiation code supplied for HDL designs
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

General Description

Xilinx LogiCORE™ high-speed parallel multipliers are predefined drop-in modules ideal for fast, real-time DSP applications or any application where multiplication speed, area efficiency, and design time are important. The multipliers use highly efficient algorithms, tuned and optimally implemented in the Xilinx XC4000E, EX, and XL series of FPGAs.

Two parallel operands can be input to the multiplier core every clock cycle. A new double precision output will be available every clock cycle after an initial latency period. For example, a 12 by 12 multiplier can produce a result every 11 nsec. in a 4000E-1.

Two pre-defined modules are available: an 8x8 multiplier and a 12x12 multiplier. Table 1 gives the implementation statistics for these modules.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 2.

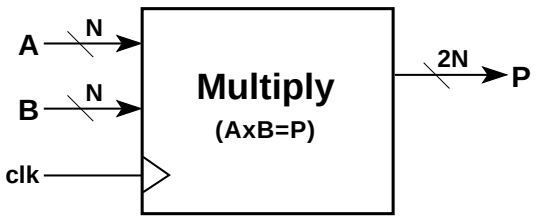


Figure 1: Core Schematic Symbol

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
A	Input	Parallel Data In, N-bits wide
B	Input	Parallel Data In, N-bits wide
clk	Input	Clock, processes data on the low to high transition
P	Output	Parallel Data Out, 2N-bits wide

Table 1: Parallel Multiplier Implementation Statistics

Design Name	Input Data	Output Data	Pin to Pin Performance[1] XC4000E-1	Latency	CLBs Used	CLB Array Size	Smallest Device
M8x8	8 x 8 Signed	16 bit	97 MHz	4 clks	70	10 rows x 8 cols	4003E
M12x12	12 x 12 Signed	24 bit	89 MHz	5 clks	156	14 rows x 13 cols	4005E

Note:
1. Based on XC4000E-1 advanced speed files.

Multiplier Trade-offs

Two different implementations of parallel multipliers trade area for speed. The area efficient designs consume about one-fourth less CLB resources than the high speed designs in the 4000E family.

The additional routing resources in the 4000EX and 4000XL families will increase the performance for the area efficient designs. In addition, both structures will benefit from the overall performance increase derived from the 4000XL .35 micron process technology.

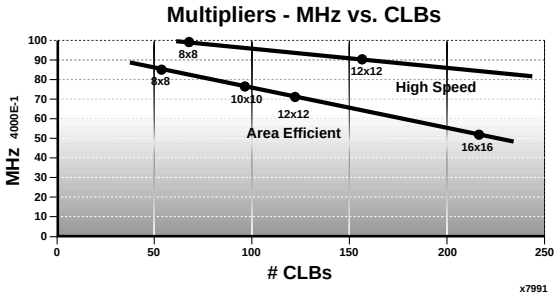


Figure 2: Trade-offs of Area Versus Speed Optimization for Multipliers

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Input data widths from 4 to 64 bits
- Output data widths from 4 to 64 bits
- Registered output
- User can determine the accuracy of the result by selecting the desired output width
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Square Root function derives the square root of an input value.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

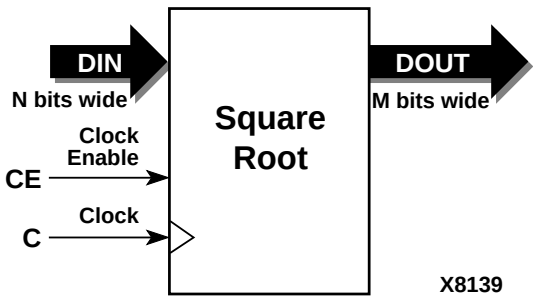


Figure 1: Block Diagram

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
DIN	Input	Input data N bits wide
C	Input	Clock – active on rising edge
CE	Input	Clock Enable – active high
DOUT	Output	Output data M bits wide

CORE Generator Parameters

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

The default output width equals one-half the input width. If higher accuracy is desired, choose a higher output width, up to the value of the input width. The decimal point in the result is always at the input_width/2 bit position.

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Input Data Width**: Select an input bit width from the pull-down menu. The valid range is 4 to 64.
- **Output Data Width**: Select an output bit width from the pull-down menu. The valid range varies with the size of input data width.

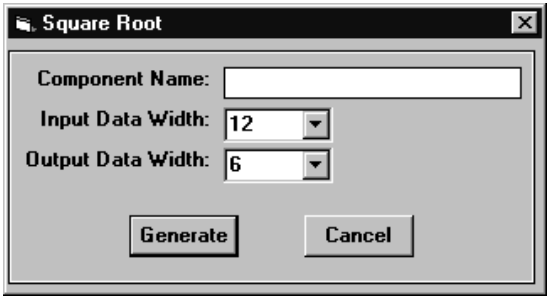


Figure 2: CORE Generator Dialog Box

Latency

The latency equals one-half the output width.

Core Resource Utilization

The number of CLBs required depends on the size of the input and output data widths selected in the CORE Generator dialog box.

If outwidth = inwidth/2:

$$CLBs = (inwidth/2 + 1)^2$$

If outwidth > inwidth/2:

$$CLBs = (inwidth/2 + 1)(outwidth + 1) + (outwidth - inwidth/2) * (outwidth - inwidth/2 + 1)/2$$

Table 2 shows the number of CLBs required for some available bit widths.

Table 2: Bit Width versus CLB Count

Input Bit Width	Output Bit Width	CLB Count
4	2	9
4	4	18
8	4	25
12	12	112
16	8	81
32	16	289

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

LogiCORE™ Registered Subtracter

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

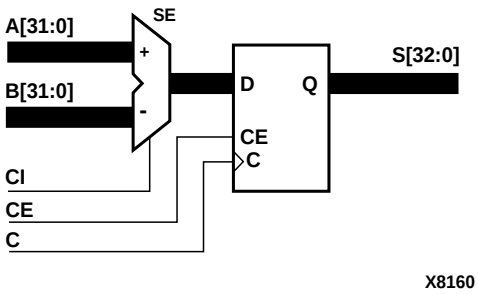


Figure 1: Core Schematic Symbol

Features

- Two data bus inputs: 2 to 32 bits wide
- Supports both 2's complement signed and magnitude-only unsigned data
- Registered output
- Clock Enable for output register
- Asynchronous Clear for output register
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Registered Subtracter module accepts two input buses, **A** and **B**, subtracts **B** from **A**, and registers the difference, **S**. The input data buses can be in either 2's complement signed or magnitude-only unsigned numbers.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A[N-1, 0]	Input	A data input
B[N-1, 0]	Input	B data input – value is subtracted from A data
CI	Input	CARRY IN – initial carry logic input. Set this to VCC if not used
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data from the internal registers to output
C	Input	CLOCK – clocks the output register
S[N, 0]	Output	DATA OUTPUT – the registered output of the subtracter

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32. The same data width is applied to both the A and B inputs. The output size is automatically set to the input width plus one.
- **Sign**: Select Signed or Unsigned.

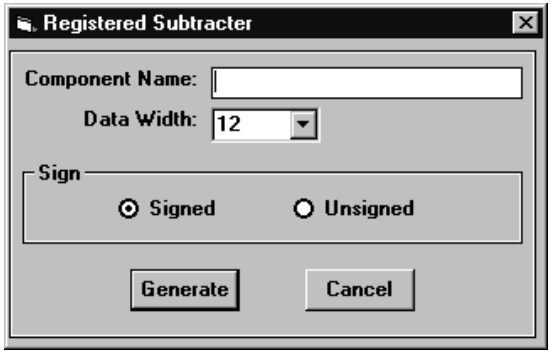


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	2
4	3
5	3
6	4
7	4
8	5
9	5
10	6
11	6
12	7
13	7
14	8
15	8
16	9
17	9
18	10
19	10
20	11
21	11
22	12
23	12
24	13
25	13
26	14
27	14
28	15
29	15
30	16
31	16
32	17

LogiCORE™ Registered Loadable Subtracter

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Two data bus inputs: 2 to 32 bits wide
- Supports both 2's complement signed and magnitude-only unsigned data
- Registered output
- Load feature to directly load the output register
- Clock Enable for output register
- Asynchronous Clear for output register
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

The Registered Loadable Subtracter module accepts two input buses, **A** and **B**, subtracts **B** from **A**, and registers the difference, **S**. The input data buses can be in either 2's complement signed or magnitude-only unsigned numbers.

When the Load (**L**) and Clock Enable (**CE**) signals are asserted, the **B** input data bus is loaded directly into the output register on the rising edge of the Clock.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

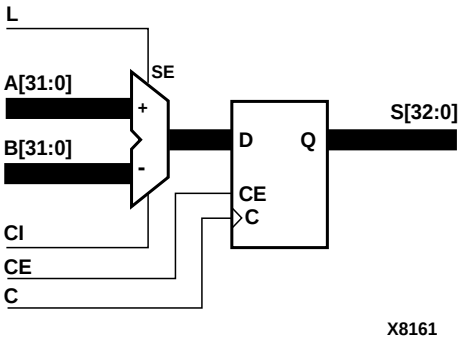


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
L	Input	LOAD – active high signal bypasses the subtracter and directly loads the B data into the output register
A[N-1, 0]	Input	A data input
B[N-1, 0]	Input	B data input – value is subtracted from A data
CI	Input	CARRY IN – Initial carry logic input. Set this to VCC if not used
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data from the internal registers to output
C	Input	CLOCK – clocks the output register
S[N, 0]	Output	DATA OUTPUT – the registered output of the subtracter

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32. The same data width is applied to both the A and B inputs. The output size is automatically set to the input width plus one.
- **Sign**: Select Signed or Unsigned.

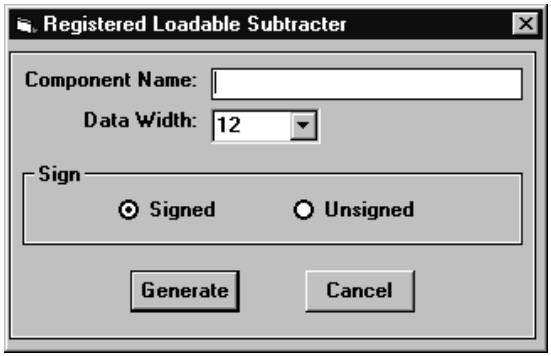


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	2
4	3
5	3
6	4
7	4
8	5
9	5
10	6
11	6
12	7
13	7
14	8
15	8
16	9
17	9
18	10
19	10
20	11
21	11
22	12
23	12
24	13
25	13
26	14
27	14
28	15
29	15
30	16
31	16
32	17

LogiCORE™ Delay Element

February 8, 1998 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

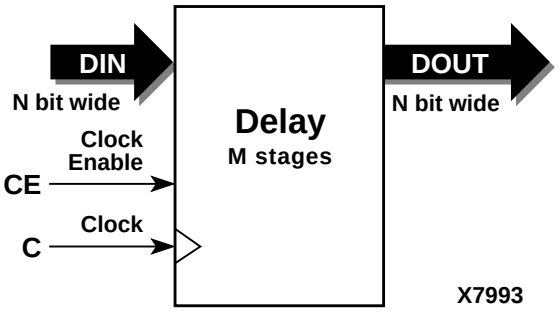


Figure 1: Core Schematic Symbol

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This parameterized module implements an M-stages by N-bits wide delay by storing data samples in a 16-word by N-bits wide RAM. If more than 17 delay stages are required (16-word deep RAM plus one register) multiple 16-word RAMs are used. The function is equivalent to an N-bit FIFO with a common input and output clock. Each 16-word RAM requires N/2 XC4000 series FPGA configurable logic blocks (CLBs) for even N and N/2 + 1 CLBs when N is odd.

Typical applications include synchronization of multiple data paths in DSP applications as they go through filtering and other signal processing.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
DIN	Input	Parallel Data In
CE	Input	Clock Enable, active high
C	Input	Clock accepts data on the low-to-high transition
DOUT	Output	Parallel Data Out – DIN delayed 1 clock. (Data changes after a low-to-high clock transition.)

CORE Generator Parameters

The CORE Generator accepts the parameters entered through the dialog box and creates the specific design from the values entered using a parameterized VHDL recipe. VHDL instantiation code and a schematic symbol are created along with the netlist for the design.

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-32.
- **Delay Stages:** 2 to 600 stages may be defined.
- **Create RPM:** Option to have the CORE generator add placement information to the module or to let the place and route software place each CLB independently. RPMs usually result in higher performance and faster implementation times.

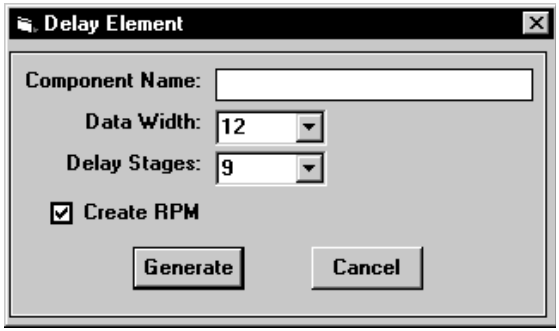


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Tables 2 and 3 show the number of CLBs required for bit widths from 2 to 17 and delays from 1 to 17.

Table 2: Utilization for Delays from 1 to 4

Data Bit Width	CLB Count
2	2
3	3
4	3
5	4
6	4
7	5
8	5
9	6
10	6
11	7
12	7
13	8
14	8
15	9
16	9
17	10

Table 3: Utilization for Delays from 5 to 17

Data Bit Width	CLB Count
2	4
3	5
4	5
5	6
6	6
7	7
8	7
9	8
10	8
11	9
12	9
13	10
14	10
15	11
16	11
17	12

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Synchronous FIFO with data width from 4 to 80 bits
- Registered output
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

A FIFO is a first-in first-out buffer. The synchronous FIFO has just one clock for both read and write operations. The input data (D) gets written to the FIFO at the rising clock edge if WE is high. The output port (Q) gets valid data if RE is high at the rising clock edge.

The FULL signal goes high if the buffer is full. EMPTY is high if the buffer is empty.

RESET is a synchronous reset pulse.

In addition, you can keep track of how full the buffer is by attaching an external up-down counter. The clock enable of this counter should be connected to the BUFCTR_CE pin and the up/down pin of the counter should be connected to BUFCTR_UPDN. Figure 1 illustrates these connections. The counter should count up if BUFCTR_UPDN is high and down if the signal is low. The counter is incremented for a write operation and decremented for a read operation.

The WE and RE pulses are internally qualified to ensure that an invalid operation never occurs. No error signal is issued in these cases, so the user must avoid these conditions.

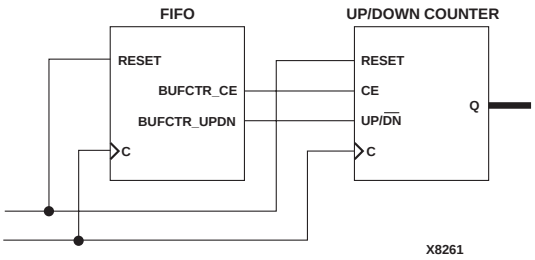


Figure 1: Connections for an Up/Down Counter

The cases to avoid are:

- No write is done (that is, the WE signal is masked) if the FIFO is full
- No read is done (that is, the RE signal is masked) if the FIFO is empty
- If single-port RAM is used, no read is done if WE is high. Similarly, no write is done if RE is high. A user must prioritize read and write externally because both of these cannot be performed simultaneously in single-port RAM.

Read and write operations cannot be done simultaneously if single-port RAM is used as a buffer. For simultaneous read and write, use dual port RAM.

Pinout

Signal names for the schematic symbol are shown in Figure 2 and described in Table 1. Timing diagrams for the single and dual ports FIFO are shown in Figures 3 and 4.

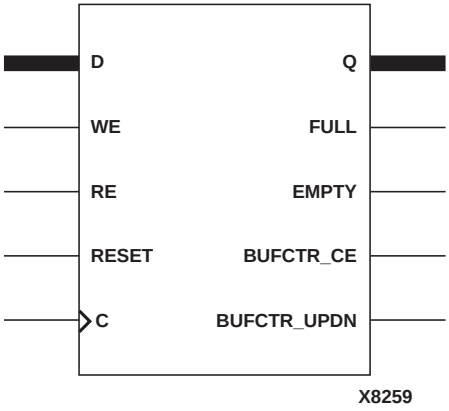


Figure 2: Core Schematic Symbol

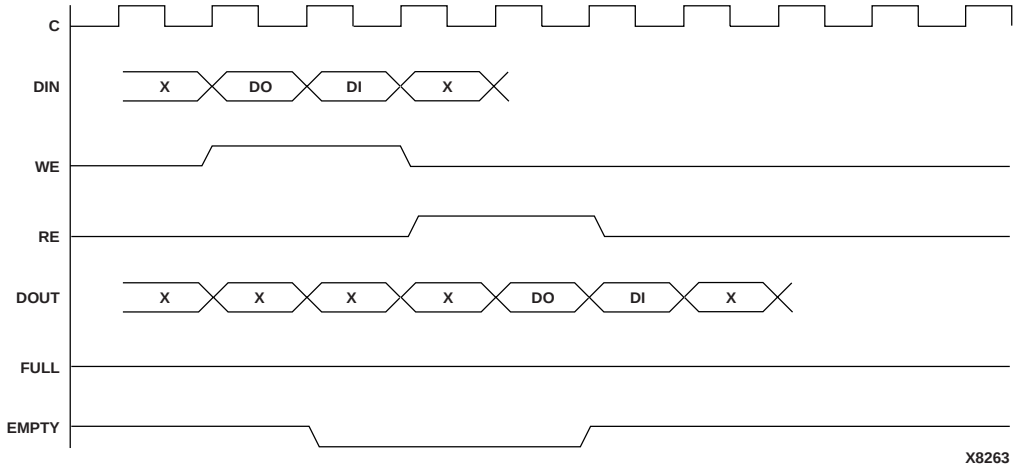


Figure 3: Timing Diagram for Single and Dual Ports

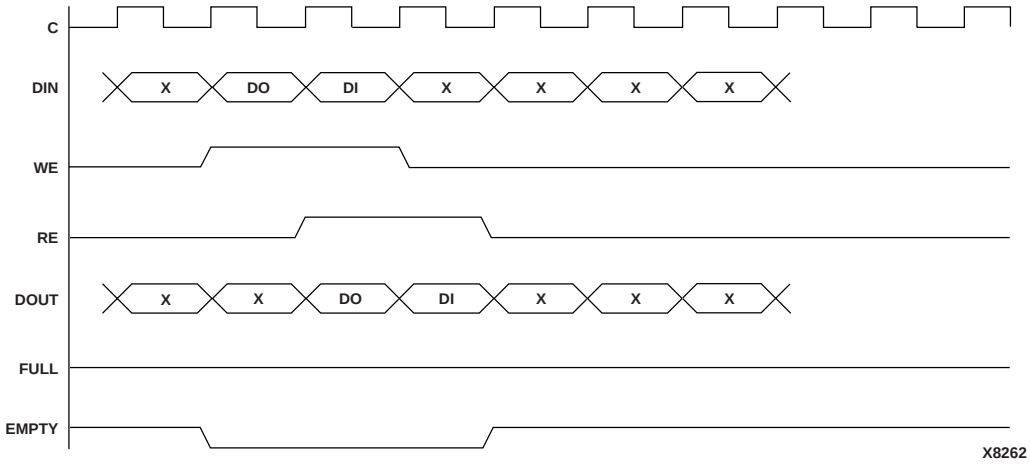


Figure 4: Timing Diagram for Dual Ports

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D	Input	DATA INPUT – input that is stored into the FIFO.
WE	Input	WRITE ENABLE – active high signal used to allow the transfer of data from the input data pins into the FIFO.
RE	Input	READ ENABLE – active high signal used to allow the transfer of data from the FIFO into the output data pins.
RESET	Input	SYNCHRONOUS RESET – clears the FIFO
C	Input	CLOCK – clocks the output register on the rising edge.
Q	Output	REGISTERED DATA OUT – the registered output of the FIFO.
FULL	Output	FULL – this signal is asserted if the FIFO is full.
EMPTY	Output	EMPTY – this signal is asserted if the FIFO is empty.
BUFCTR_CE	Output	Pin connected to the clock enable of an external up-down counter.
BUFCTR_UPDN	Output	Pin connected to the up-down pin of an external up-down counter.

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 5. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 4-80.
- **FIFO Depth**: Select the number of words in the FIFO from the pull-down menu. The valid range is 16-256 in multiples of 16.
- **Internal Memory**: Select either Single Port or Dual Port RAMs. You must choose Dual Port to perform simultaneous read/write operations.

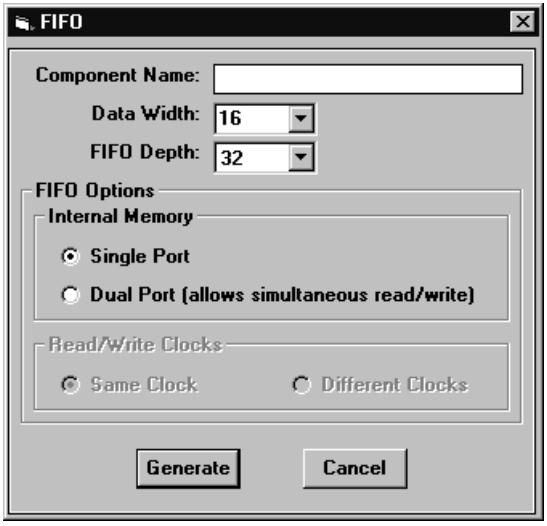


Figure 5: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for a sample of the available depths and bit widths.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information, contact your local Xilinx sales representative or e-mail requests to dsp@xilinx.com.

Table 2: CLB Utilization for Example FIFO Implementations

Depth	Ports[1]	CLB Counts for Example Data Word Widths						
		4-bit	8-bit	16-bit	32-bit	48-bit	64-bit	80-bit
16	Single	14	16	20	28	36	44	52
	Dual	14	18	26	42	58	74	90
32	Single	17	21	29	45	61	77	93
	Dual	22	34	58	106	154	202	250
48	Single	24	32	48	80	112	144	176
	Dual	28	44	76	140	204	268	332
64	Single	27	35	51	83	115	147	179
	Dual	32	48	80	144	208	272	336
96	Single	34	46	70	118	166	214	262
	Dual	42	66	114	210	306	402	498
128	Single	40	56	88	152	216	280	344
	Dual	52	84	148	276	404	532	660
256	Single	66	98	162	290	418	546	674
	Dual	94	158	286	542	798	1054	1310

Note 1: These are ports in the internal memory

LogiCORE™

16-Word Deep Registered Look-Up Table

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro is a registered look-up table, or ROM-based storage array. This is equivalent to an N-bit wide, 16-bit deep ROM. The address pins, A[3:0], are used to address the LUT data. The output of the LUT is registered on the rising edge of the C (CLOCK) signal with the CE (CLOCK ENABLE) signal asserted (HIGH).

The ROM data is defined via the INIT attribute. The INITs are attached to the LUT by the CORE Generator, which defines the memory's contents.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

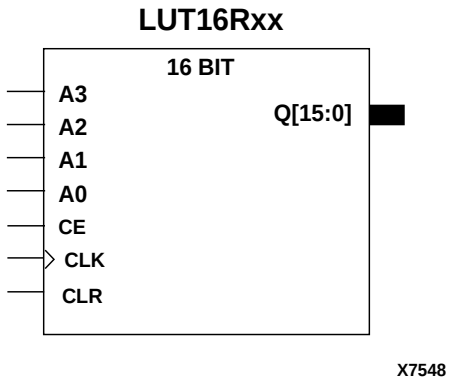


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A[3:0]	Input	ADDRESS – the ROM location, predefined by the INIT value, is decoded from these Address pins and presented to the register's input
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data from the LUT to the output register
CLK	Input	CLOCK – clocks the output register on the rising edge
CLR	Input	ASYNCHRONOUS CLEAR – clears the register. Does not require the Clock or CE signal
Q[N-1, 0]	Output	REGISTER LUT DATA OUTPUT – the registered output of the look-up table

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.

- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-31.
- **LUT Data**: Enter the data for each address location.
- **Radix**: Select the radix of the LUT data: Hexadecimal or Decimal.
- **Sign**: If the radix is decimal, select the sign of the constant value: Signed or Unsigned. If the radix is hex, the constant is always signed.
- **Read From File**: In the pop-up window, set the name of the file that contains your LUT data.

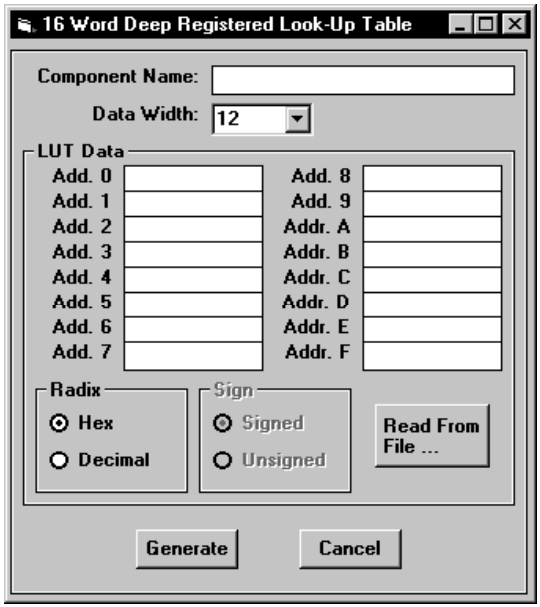


Figure 2: CORE Generator Dialog Box

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	1
3	2
4	2
5	3
6	3
7	4
8	4
9	5
10	5
11	6
12	6
13	7
14	7
15	8
16	8
17	9
18	9
19	10
20	10
21	11
22	11
23	12
24	12
25	13
26	13
27	14
28	14
29	15
30	15
31	16
32	16

LogiCORE™

32-Word Deep Registered Look-Up Table

February 8, 1998 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro is a registered look-up table, or ROM-based storage array. This is equivalent to an N-bit wide, 32-bit deep ROM. The address pins, A[4:0], are used to address the LUT data. The output of the LUT is registered on the rising edge of the C (CLOCK) signal with the CE (CLOCK ENABLE) signal asserted (HIGH).

The ROM data is defined via the INIT attributes. The INITs are attached to the LUT by the CORE Generator, which defines the memory's contents.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

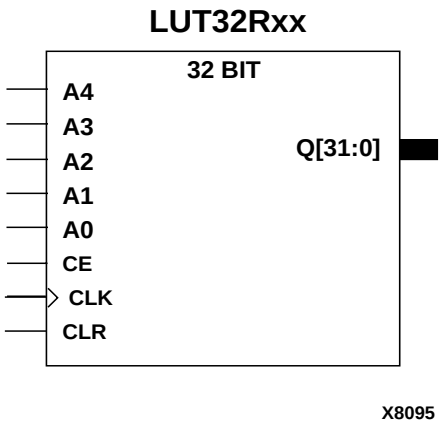


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A[4:0]	Input	ADDRESS – the ROM location, predefined by the INIT value, is decoded from these Address pins and presented to the register's input
CE	Input	CLOCK ENABLE – active high signal used to enable the transfer of data from the LUT to the output register
CLK	Input	CLOCK – clocks the output register on the rising edge
CLR	Input	ASYNCHRONOUS CLEAR – clears the register. Does not require the Clock or CE signal
Q[N-1, 0]	Output	REGISTER LUT DATA OUTPUT – the registered output of the look-up table

CORE Generator Parameters

The CORE Generator dialog box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-31.
- **LUT Data**: Enter the data for each address location.
- **Radix**: Select the radix of the LUT data: Hexadecimal or Decimal.
- **Sign**: If the radix is decimal, select the sign of the constant value: Signed or Unsigned. If the radix is hex, the constant is always signed.
- **Read From File**: In the pop-up window, set the name of the file that contains your LUT data.

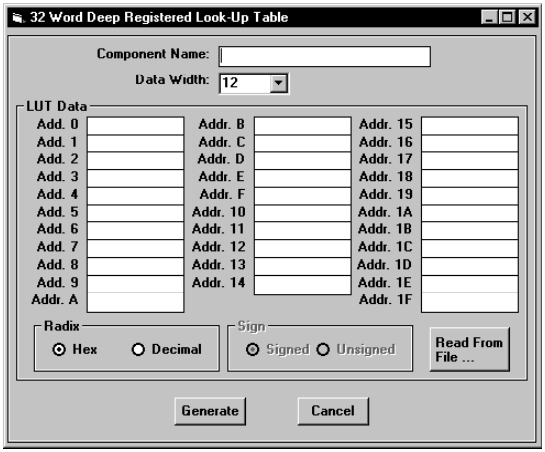


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

Table 2 shows the number of CLBs required for each available bit width.

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

Table 2: Bit Width versus CLB Count

Bit Width	CLB Count
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com

Features

- Synchronous RAM with data width from 2 to 31 bits
- Registered output
- Clock Enable for output register
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro accepts an N-bit value and stores it in a RAM. The depth of the RAM is determined by the user. The output data is read from the RAM and registered. Data is both written to and read from the address selected on the A address pins. Access of the RAM is accomplished on the rising edge of the Clock signal with the CE pin asserted (HIGH).

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

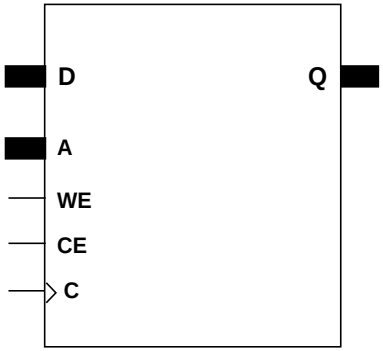


Figure 1: Core Schematic Symbol

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D[N-1, 0]	Input	DATA INPUT – input that is stored into the RAM.
A	Input	RAM ADDRESS – the RAM location addressed by these signals is read and written.
WE	Input	WRITE ENABLE – active high signal used to allow the transfer of data from the input data pins D[N-1, 0] into the RAM (write).
CE	Input	CLOCK ENABLE – active high signal used to allow the transfer of RAM data from the output data pins of the RAM into the output register.
C	Input	CLOCK – clocks the output register on the rising edge.
Q[N-1, 0]	Output	REGISTERED OUTPUT DATA – the registered output of the RAM.

CORE Generator Parameters

The core generator dialogue box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Address Width**: The number of bits needed to address all of the words in the RAM.
- **Depth**: Select the number of words in the RAM from the pull-down menu. The number must be a multiple of 16.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-31.
- **Load Init Values**: Specifies the file that contains the initial values for each address.
- **Show Init Values**: Display the initial values after they have been loaded.
- **coe file**: Displays the name of the coefficient file. This field is read-only.

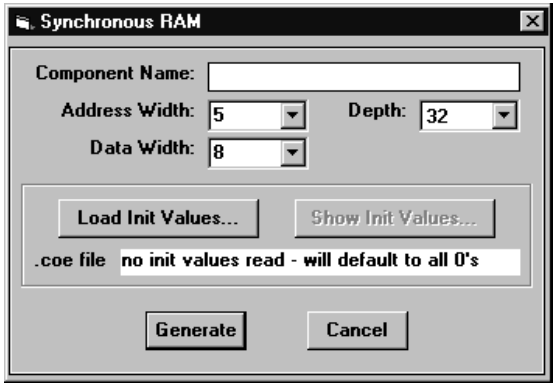


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

The number of CLBs required depends on the values of the depth and data width fields selected in the CORE Generator dialog box.

Table 2 shows the equations to calculate the maximum number of CLBs required for selected depths and data widths.

Table 2: Bit Width versus CLB Count

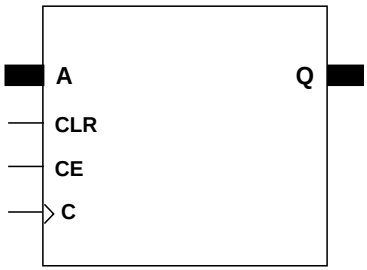
Depth	Data Width	CLB Count
16	2N	N
16	2N+1	N
32	N	N
48	N	2N+1
>=64	2N	(Depth/16) (N+1)
>=64	2N+1	(Depth/16) (N+2)

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: dsp@xilinx.com
URL: www.xilinx.com



X8166

Figure 1: Core Schematic Symbol

Features

- Registered ROM with data width from 2 to 31 bits
- Registered output
- Clock Enable for output register
- Asynchronous clear of the output register
- Drop-in modules for the XC4000E, EX, and XL families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

Functional Description

This macro creates a ROM whose size is determined by the data width and depth variables in the dialog box. The output data is read from the ROM and registered. Data is read from the address selected on the A address pins. Access of the ROM is accomplished on the rising edge of the Clock signal with the CE pin asserted (HIGH). A CLR signal can be used to clear the register asynchronously.

Pinout

Signal names for the schematic symbol are shown in Figure 1 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A	Input	ROM ADDRESS – the address of a particular ROM location.
CLR	Input	CLEAR – asynchronous clear of the register.
CE	Input	CLOCK ENABLE – active high signal used to allow the transfer of ROM data from the output data pins of the ROM into the output register.
C	Input	CLOCK – clocks the output register on the rising edge.
Q[N-1, 0]	Output	REGISTERED OUTPUT DATA – the registered output of the ROM.

CORE Generator Parameters

The core generator dialogue box for this macro is shown in Figure 2. The parameters are as follows:

- **Component Name**: Enter a name for the output files generated for this module.
- **Address Width**: The number of bits needed to address all of the words in the ROM.
- **Depth**: Select the number of words in the ROM from the pull-down menu. The number must be a multiple of 16.
- **Data Width**: Select an input bit width from the pull-down menu. The valid range is 2-31.
- **Load Init Values**: Specifies the file that contains the initial values for each address.
- **Show Init Values**: Display the initial values after they have been loaded.
- **coe file**: Displays the name of the coefficient file. This field is read-only.

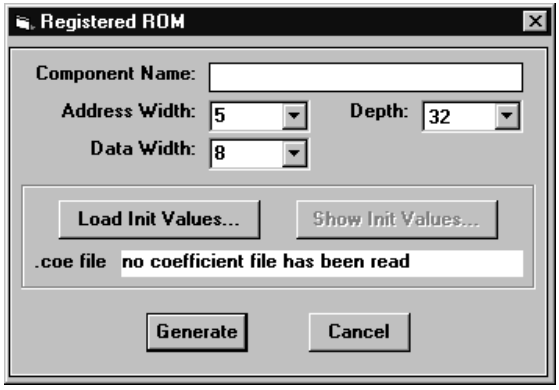


Figure 2: CORE Generator Dialog Box

Core Resource Utilization

The number of CLBs required depends on the values of the depth and data width fields selected in the CORE Generator dialog box.

Table 2 shows the equations to calculate the maximum number of CLBs required for selected depths and data widths.

Table 2: Bit Width versus CLB Count

Depth	Data Width	CLB Count
16	2N	N
16	2N+1	N
32	N	N
48	N	2N
>=64	2N	(Depth/16) (N+1)
>=64	2N+1	(Depth/16) (N+1)

Ordering Information

This macro comes free with the Xilinx CORE Generator. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

AllianceCORE Products

AllianceCORE Products

February 8, 1998

Description

The AllianceCORE program is a cooperative effort between Xilinx and independent third-party core developers. It is designed to produce a broad selection of industry-standard solutions dedicated for use in Xilinx programmable logic.

Xilinx takes an active role with its partners in the process of productizing AllianceCOREs. This is unique to the AllianceCORE program. Because the process is so involved, we work closely with our partners to select the right cores first which helps raise the quality and usability of the cores that are offered.

AllianceCORE Criteria

A core must meet a minimum set of criteria before it can receive the AllianceCORE label.

Core Selection

Xilinx looks at cores from a practical point of view. A programmable logic version of a core must have value over an ASIC or standard product version of the same function. It must be cost effective and make sense for use in a programmable device in a production system. If a candidate core does not pass these simple tests, then it does not make sense to invest the effort to convert it to an AllianceCORE product.

Core Qualification

Generic, synthesizable cores offer maximum flexibility for users with unique requirements. This is typically the format for cores provided to the ASIC market.

With programmable logic, however, this flexibility can come at the expense of efficiency and performance. It can take considerable effort to get a core to synthesize in a way that meets density and timing requirements. Time spent trying to accomplish this quickly reduces the time-to-market advantage of using programmable logic and cores in the first place.

As a result, Xilinx is does not promote generic, synthesizable cores as AllianceCORE products. Instead, they are generally provided as black-boxes that allow customization in critical areas. This guarantees that the implementation is optimized for density while still meeting performance, preserving the time-to-market value of programmable logic.

Flexibility is provided by allowing you to quickly implement your unique logic on the same device. Source code versions of the cores are often available from the partners at additional cost for those who need ultimate flexibility.

Announced AllianceCORE products have been implemented and verified in a Xilinx device. They are available immediately for purchase in a Xilinx netlist format such as .xnf or .ngd and are usable in either the Xilinx Alliance or Foundation series development software. When required, the cores also come with appropriate constraints files to guarantee functionality and critical timing. In all cases the partner is responsible for guaranteeing the operation of the function itself.

AllianceCORE products originated from either schematic or HDL entry tools.

Core Integration

AllianceCORE products are not just cores. They are complete solutions for system designs. While cores by themselves have value, in many cases it is often not enough to just supply the core. You need tools such as system software and prototyping boards to help you rapidly integrate the core into your design, perform system debug in a real-world environment, and then quickly convert the prototype to a production unit. This is particularly true of complex functions.

Many AllianceCORE functions are supported by Xilinx-based demonstration or prototyping boards. Some also have system simulation models or development software. This allows you to evaluate and work with the function before you have to layout your board.

These tools are provided by the AllianceCORE partner, usually at additional cost. Descriptions of available support tools for each core are included in this data book.

Complete solutions like these help preserve the value of using programmable logic while lowering the support burden for the core provider.

Acquiring AllianceCORE Products

AllianceCORE products are sold and serviced directly by the AllianceCORE partners since they are the experts for their particular products. They are responsible for pricing, licensing terms, delivery and technical support. Contact information for each partner is included in the AllianceCORE Partner Profiles section of this data book.

If you want additional information about the AllianceCORE program or are interested in becoming a partner, contact Xilinx directly.

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Attn: CORE Solutions Group
Phone: 408-879-5381
Fax: 408-377-3259

E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Maria Aguilar, Project Coordinator
Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, AZ 85202
Phone: +1 888-360-9044 (in the USA)
+1 602-491-4311 (international)
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Features

- Multi-master operation with arbitration and clock synchronization
- Support for reads, writes, burst reads, burst writes, and repeated start
- User-defined timing and clock frequency
- Fast mode and standard mode operation
- Compatible with popular protocols

Applications

- Embedded microprocessor boards
- Consumer and professional audio/video
- Home and automotive radio

General Description

The XF-TWSI is an industry standard two-wire serial interface supporting multiple masters.

AllianceCORE™ Facts

Core Specifics		
Device Family	XC4000E	XC5200
CLBs Used	137	64
IOBs Used	32[1]	32[1]
System Clock f_{max}	10+ MHz[2]	
Device Features Used	Tbufs, global clock buffers	

Supported Devices/Resources Remaining		
	I/O	CLBs
XC4005E-4[3] PC84C	29[1]	59
XC5202PC84-5[3]	33[1]	0

Provided with Core	
Documentation	Core schematics Implementation instructions
Design File Formats	ViewLogic schematic files Live hierarchical tree LCA Files
Schematic Symbols	ViewLogic
Verification Tool	Machine-readable simulation vectors for ViewLogic ViewSim
Constraint Files	None
Evaluation Model	None
Reference designs & application notes	None
Additional Items	Warranty by MDS

Design Tool Requirements	
Xilinx Core Tools	XACTstep 6.0.0
Entry/Verification Tools	ViewLogic PROcapture 6.1 or Workview Office

Support

Memec Design Services warrants that the design delivered by Memec Design Services will conform to the design specification. This warranty expires 3 months from the date of delivery of the design database. Contact Memec Design Services for the Design License Agreement with complete Terms and Conditions of Sale.

Notes:
1. Assuming all core signals are routed off-chip.
2. Minimum guaranteed speed.
3. Specific devices are minimum size and speed recommended - the core will work in any larger or faster devices from the same families.

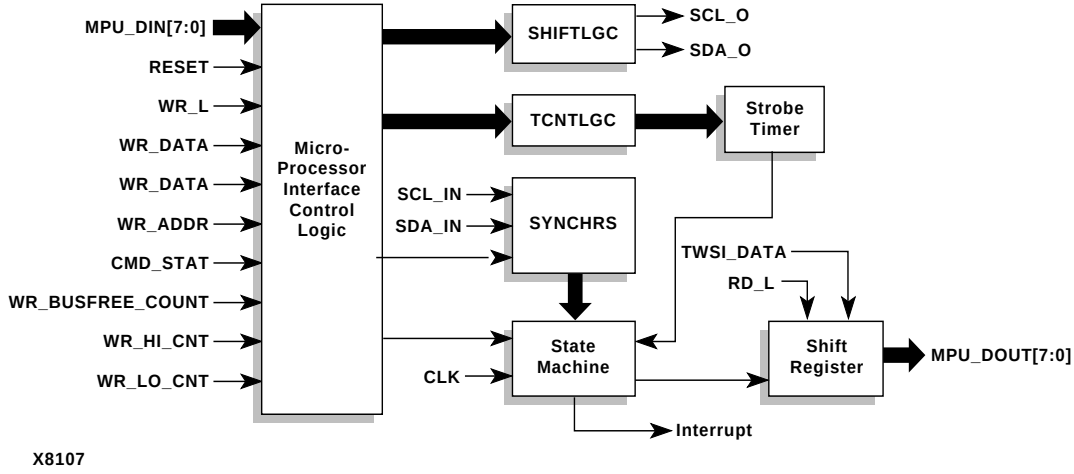


Figure 1: XF-TWSI Block Diagram

Functional Description

The XF-TWSI is partitioned into modules as shown in Figure 1 and described below.

Microprocessor Interface Control Logic

There are three registers used to interface to the host: the Data Register, the Address Register, and the Command Register. The strobes WR_DATA, WR_ADDR, and WR_CMD are directly connected to the clock pins of these register flip-flops for ease of interface.

There is a single parallel in, parallel out, serial in, serial out shift register called NUPSHIFT, which performs the shifting of data for address cycles, write cycles, and read cycles. The parallel output drives MPU_DOUT[7:0] which is used to return read data to the host**.**

Command Register: The Command Register consists of three bits: the READ bit (bit 0), the STOP bit (bit 1), and the REPEATED START bit (bit 2). These three bits control the operation performed on the serial interface.

The READ bit determines if the operation will be a read (high), or a write (low). The STOP bit will determine if the cycle will stop at the end of the data cycle, or continue on to a burst. The REPEATED START bit will cause the generation of a repeated start protocol.

To generate a single read cycle, the Address register is written with the desired address and the Command Register is written with B011 (binary 011, REPEATED START=0, STOP=1, READ=1) to perform the operation and stop. When the operation is completed (or aborted due to error) the Interrupt pin will go active and the data may be read on MPU_DOUT[7:0].

To generate a single write cycle, the Address Register is written first, followed by the Data Register, and the Command Register is written with B010 to perform the operation and stop.

A burst read operation is performed in the same manner as a single read, except that the Command Register is written with the STOP bit de-asserted (B001). At the end of the operation an interrupt is generated to the host and the SCL clock line is stretched. When the host has read the data it can write a B011 to the Command Register to perform one more read followed by a stop, or it can write a B001 to perform another read in anticipation of further bursting. Note that there is only one address cycle for a burst read cycle. Also, it is not permitted to follow a read operation or burst read operation with a write operation without first performing a repeated start or a stop.

A burst write operation is performed by setting up the Address and Data Registers, and writing the Command Register with B000. When an interrupt is detected, the next data item is written to the Data Register followed by either a B010 (last data written) or B000 (more data will follow).

The repeated start operation is used to directly follow a read with a write, or a write with a read, without a stop in-between. Some devices require this for the generation of a second address within the serial bus itself.

Any read or write operation may be followed by a repeated start by resetting the STOP bit and setting the REPEATED START bit when writing to the Command Register. For example, to perform a write cycle followed by a repeated start and a read, the Command Register is written with B100 to perform the write operation. An interrupt will be generated, as always, at the end of the write. Also at the

end of the write the clock line (SCL) will be stretched until the new command is written. If a single read is desired the Command Register is written with B011.

Timing: The primary clock, CLK, may be operated from 8 to 16 MHz by changing the timing dividers on BUSFREE_COUNT, HI_COUNT, and LO_COUNT core pins. These three bussed pins are used to define the timing for the part by attaching the appropriate divider values. These divider values are multiplexed into a counter which is used to time the low and high width of the SCL clock, the setup time and hold times, and the “bus free” time for arbitration purposes.

The BUSFREE_COUNT value is used to set the bus free time in terms of CLK periods. The XF-TWSI must see both the SDA and SCL pins in a high state for the bus free period before it issues a start or repeated start command on the bus. Note that this value must be set greater than the longest SCL high period in the system for arbitration to function properly. For Standard mode devices this should be set to at least 5 microseconds. For Fast mode devices this should be set to at least 1.5 microseconds.

The HI_COUNT value is used to define the low period of the SCL clock and the setup time for a repeated start command (rising edge of SCL to falling edge of SDA). This value is generally set to 4.7 microseconds for Standard mode and 1.3 microseconds for Fast mode.

The LO_COUNT value is used to define the high width of the SCL clock, the setup time for a stop command (rising edge of SCL to rising edge of SDA), as well as the start command hold time (falling edge of SDA to falling edge of SCL). This value is generally set to 4.0 microseconds for Standard mode and .6 microseconds for Fast mode.

The values can be added to busses attached to these pins as a series of VCCs and GNDs.Note that the actual values obtained from these dividers will be slightly larger due to state machine overhead.

SYNCHRS Block

The SDA and SCL inputs are passed through this module that performs a dual-rank synchronization and glitch filtering when enabled by the FILTER_EN signal. The synchronized versions of the SDA and SCL signals are used in all core modules.

The XF-TWSI core treats both the SDA and SCL lines as data lines. The SDA line is actually sampled some number of clocks after the rising edge of SCL is detected. This allows for greater noise immunity and more robust operation.

State Machine Block

The control for the serial interface comes from the TWSI_SM state machine. This state machine controls the loading and enabling of all shift registers and counters, and is responsible for implementing the basic interface protocol.

Arbitration: Before the XF-TWSI issues either a start or repeated start command it samples the synchronized versions of SDA and SCL for BUSFREE_COUNT x clock periods of high values. If at any time either of these pins is detected in a low (driven) state then the count starts over (it assumes that another master is on the bus). The automatic retrying to obtain the serial bus only occurs until the start command is issued. After that the responsibility of the retry is left to software, since cycles may be of infinite length.

After a start is issued and while the address or data is being written to the slave, the core samples the data on the SDA pin while SCL is high. If a data mismatch is detected then the operation is aborted with an interrupt and a BUSLOSS status. This can also occur during burst reads whenever a NACK is intended to be issued and an ACK is detected instead.

Performing Basic Cycles: The interface between the host and the XF-TWSI is interrupt-based. The host sets up the Address and Data Registers, writes the Command Register to initiate the desired operation, and performs other operations until the interrupt is generated by the core. Alternatively, if polled operation is desired, the Status Register may be read to see if the interrupt line is active.

Upon receiving the interrupt the host must read the Status Register to see if either of the two error conditions is present. Bit 2 in the Status Register is BUSLOSS, which occurs if the interface lost the bus to another master before completion of the cycle. Bit 1 in the Status Register is STATUS TIMEOUT, which occurs if the slave does not respond with an acknowledge when required. For either of these error conditions the host must retry the current cycle up until the last interrupt. Note that for burst operations the host is responsible for writing the incremented address into the Address Register before retrying the operation.

After the host receives the interrupt and determines the status, it sets up the registers again and writes the Command Register to start the next cycle. The act of writing the Command Register clears out the current interrupt.

Shift Logic (SHIFTLGC)

The basic cycle on theXF-TWSI serial interface consists of an address cycle followed by a data cycle. The address consists of seven bits and the read/write bit (the LSB). The MSB is always transmitted first on the SDA line. The data cycle can either be a read or a write. For a write operation the XF-IIC shifts the data from the Data Register onto the SDA line. For a read operation the core captures the data into the shift register. The data cycle can end in three different ways:

1. A stop can be generated which terminates the current cycle.
2. Another data cycle can take place (a burst).
3. A repeated start can be generated by the interface.

There is always one interrupt generated for each data cycle independent of the type of cycle. For example, for a burst read cycle an interrupt will be generated for each byte read. For a burst write cycle an interrupt will be generated when each byte transfer is completed.

A repeated start is used to turn the bus around; when a read cycle must be followed directly by a write cycle without a stop in-between. Since the READ bit is a part of the address, if a read followed by write is desired without a stop command, a second address must be issued following the data cycle. The sequence of events in a repeated start cycle is: start, address cycle, data cycle, repeated start, address cycle, data cycle, stop. Each of the data cycles can be repeated if bursting is desired, and the stop cycle could actually be another repeated start, if desired.

External Crystal Support

This core does not support connection of a crystal directly to the device. The XC4000 will require a clock input.

Table 1: Available Command Sequences

Operation	Subcycle	Command Register Contents		
		Repstart	Stop	Read
Single Read	Single Read	0	1	1
Single Write	Single Write	0	1	0
Burst Read	Burst Read Start	0	0	1
	Burst Read Continue	0	0	1
	Last Burst Read	0	1	1
Burst Write	Burst Write Start	0	0	0
	Burst Write Continue	0	0	0
	Last Burst Write	0	1	0
Read with Repeated Start followed by Burst Write	Read with RepStart	1	0	1
	Burst Write Start	0	0	0
	Burst Write Continue	0	0	0
	Last Burst Write	0	1	0
Write with Repeated Start followed by Burst Read	Write with RepStart	1	0	0
	Burst Read Start	0	0	1
	Burst Read Continue	0	0	1
	Last Burst Read	0	1	1
Illegal Sequence	Burst Read	0	0	1
	Burst Write	0	0	0
Illegal Sequence	Burst Write	0	0	0
	Burst Read	0	0	1

Pinout

The XF-TWSI may be implemented as stand-alone logic using the provided pinout or may be implemented internally with the users design. Both versions are included with the Core deliverables. Signal names are provided in the block diagram shown in Figure 1, and Table 2.

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
CLK	Input	Primary clock used for all circuits.
RESET	Input	Reset, active high.
RD_L	Input	When active “0” with proper read buffer/register selected allows reads on the MPU_DOUT[7:0].
TWSI_DATA	Input	When active “1” with a RD_L allows reads on MPU_DOUT{7:0].
WR_L	Input	Rising edge of this signal registers data in internal registers when addressed.
MPU_DIN[7:0]	Input	Microprocessor Data In lines used to program Data, Address, and Command Registers.
WR_DATA	Input	When active “1” with a WR_L strobes 8-bits of data from MPU_DIN[7:0] into Data Register for serial bus write cycle.
WR_ADDR	Input	When active “1” with a WR_L strobes least significant 7 bits of data on MPU_DIN[7:0] lines into Address Register for all Serial Bus Operations
CMD_STAT	In/Out	When active “1” with a WR_L strobes least significant 3 bits of data from MPU_DIN[7:0] lines into Command Register and initiates serial bus cycle; also clears Interrupt line at this time. When active “1” with a RD_L reads Status Register.
WR_HI_CNT	Input	When active “1” with a WR_L strobes 8-bits of data from MPU_DIN[7:0] into HI_COUNT Register used to set number of CLK clock periods for low period of SCL and setup time for a repeated start operation.
WR_LO_CNT	Input	When active “1” with a WR_L strobes 8-bits of data from MPU_DIN[7:0] into LO_COUNT Register used to set number of CLK clock periods for high count of SCL, hold time for start command, and setup time for stop command.
WR_BUSFREE_COUNT	Input	When active “1” with WR_L strobes 8-bits of data from MPU_DIN[7:0] into BUSFREE_COUNT register used to set bus free period.
INTERRUPT	Output	Interrupt line set upon completion or abort of serial cycle. It is cleared by writing to Command, active high.
MPU_DOUT[7:0]	Output	Returns read data after activation of Interrupt pin and error free status.
SDA_IN	In	Serial Data In.
SDA_O	Out	Serial Data Out.
SCL_IN	In	Serial Clock In.
SCL_O	Out	Serial Clock Out.

Core Modifications

The XF-TWSI meets or exceeds the industry standard. However, in most cases the Timespecs can be tightened significantly. Successful operation with 120ns bus cycles has been achieved. In all cases, a post-route timing analysis should be performed to verify performance. Implementation beyond 10MHz and other customizing is available through Memec Design Services.

Verification Methods

Complete functional and timing simulation has been performed on the XF-TWSI using Viewsim. (Simulation vectors used for verification are provided with the core.) The README.TXT file has a short description of the top level CMD files used to simulate the design. These CMD files include descriptions and comments.

Recommended Design Experience

Users should be familiar with ViewLogic PROcapture or Office schematic entry and Xilinx design flows. Users should also have experience with microprocessor systems and peripherals.

Ordering Information

The XF-TWSI Two-wire Serial Interface Core is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire™ gate arrays. To purchase or make further inquiries about this or other Memec Design Services products, contact MDS directly at the location listed on the front page.

Information furnished by Memec Design Services is believed to be accurate and reliable. Memec Design Services reserves the right to change specifications detailed in this data sheet at any time without notice, in order to improve reliability, function or design, and assumes no responsibility for any errors within this document. Memec Design Services does not make any commitment to update this information.

Memec Design Services assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction, if such be made, nor does the Company assume responsibility for the functioning of undescribed features or parameters. Memec Design Services will not assume any liability for the accuracy or correctness of any support or assistance provided to a user.

Memec Design Services does not represent that products described herein are free from patent infringement or from any other third-party right. No license is granted by implication or otherwise under any patent or patent rights of Memec Design Services.

Memec Design Services products are not intended for use in life support appliances, devices, or systems. Use of a Memec Design Services product in such application without the written consent of the appropriate Memec Design Services officer is prohibited.

All trademarks, registered trademarks, or servicemarks are property of their respective owners.

Related Documentation and Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products
/logicore/alliance/tblpart.htm

Mobile Media Research, Inc.

39675 Cedar Boulevard, #295A
Newark,CA 94560 USA
Phone: 1-800-799-MMRI (6674)
+1 510-657-4891
Fax: +1 510-657-4892
E-mail: sales@mobmedres.com
URL: www.mobmedres.com

Features

- 100% programmable single-chip interface to PCMCIA bus for I/O devices such as fax/modem cards
- Configuration registers per PCMCIA release 2.0
- Support for up to 2K-bytes of Attribute Memory
- Configurable address space
- Supports the following configuration options:
  - I/O Base Address (2E8h, 2F8h, 3E8h, 3F8h)
- Digital Audio (enable/disable)
- Ring Indicate (enable/disable)
- Card Enable/Disable
- Power Down (enable/disable)
- Card Reset
- Interface for an external 2Kx8 EEPROM
- Support for Speaker/Audio output through the PCMCIA bus
- Supports Rockwell, Sierra, Exar and Intel Fax/Modem chipsets
- ExCA™ compatible

Applications

- PCMCIA I/O devices such as fax/modem cards.

AllianceCORE™ Facts

Core Specifics		
Device Family	XC3x00A	
CLBs Used	50	
IOBs Used	46[1]	
System Clock f_{max}	10 MHz	
Device Features Used	Not Applicable	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC3x42A TQ100	36[1]	94
Provided with Core		
Documentation	User Documentation	
Design File Formats	ViewLogic Schematic	
Constraint Files	Not Applicable	
Verification Tool	ViewSim Command Files	
Schematic Symbols	Viewlogic	
Evaluation Model	Prototyping board Available extra	
Reference designs & application notes	None	
Additional Items	None	
Design Tool Requirements		
Xilinx Core Tools	XACTstep 5.2.1/6.0.1	
Entry/Verification Tools	ViewLogic Schematic	
Support		
Support provided by Mobile Media Research		

Notes:

1. Assuming all core signals are routed off-chip.

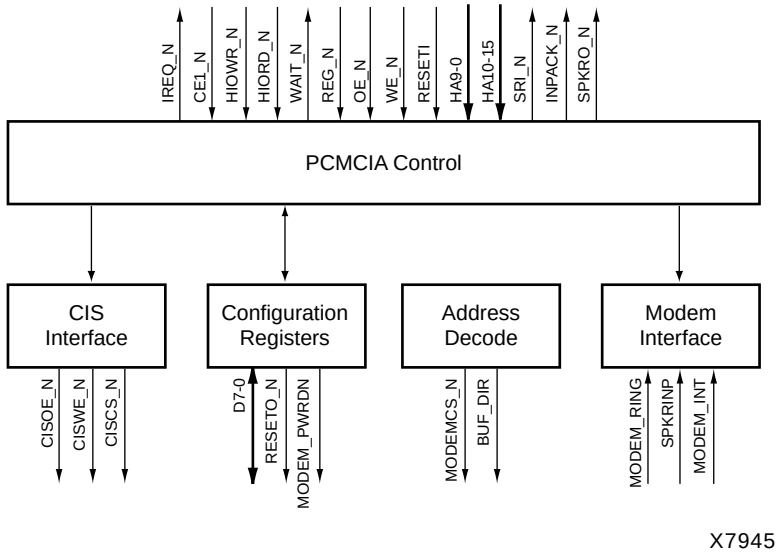


Figure 1: PCMCIA Fax/Modem Block Diagram

General Description

The PCMCIA Fax/Modem is a Xilinx FPGA-based macro for a PCMCIA fax/modem card using an external fax/modem chipset. The macro implements two PCMCIA Card Configuration Registers as well as the interface logic for controlling an external CIS EEPROM and the interface signals to a serial port. It provides a Modem Chip Select as well as inputs for a Ring Indicator and Speaker through an external DAA.

Functional Description

The fax/modem interface is partitioned into modules as shown in Figure 1 and described below.

PCMCIA Control

The PCMCIA control block decodes the control signals from PCMCIA and manages the data transfers from the Configuration Registers and the fax/modem.

CIS Interface

This interface decodes and controls accesses to the external CIS EEPROM.

Configuration Registers

This block implements the PCMCIA Function Configuration Registers (FCRs) which are used to control and configure the PC Card. These registers are 8-bit registers and include Configuration Option Register as well as the Configuration Status Register.

Address Decode

This decodes the addresses and generates the chip selects as well as transceiver controls. The decode is programmable based upon the configuration of the FCRs.

Modem Interface

This block controls some modem specific signals and passes these on the PCMCIA bus based on configuration.

Pinout

The pinout of the macro has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are provided in the block diagram shown in Figure 1, and Table 1 below.

Table 1: Pinout

Signal	Signal Direction	Description
PCMCIA Bus Interface Signals		
IREQ_N	Tri-state	Interrupt Request to host
CE1_N	Input	Card Enable from host
HIORD_N	Input	I/O Read from host
HIOWR_N	Input	I/O Write from host
WAIT_N	Tri-state	Wait indication to host
REG_N	Input	Attribute Memory Select
OE_N	Input	Output Enable from host
WE_N	Input	Write enable from PCMCIA
RESETI	Input	Reset from host
HA0-9	Input	Address inputs from host
HA10-15	Input	Address inputs reserved for future extensions
SRI_N	Tri-state	Ring Indicator output to host
INPACK_N	Output	Input Acknowledge to host
SPKRO_N	Output	Digital audio output
Back End Interface Signals		
CISOE_N	Output	CIS Output Enable
CISWE_N	Output	CIS Write Enable
CISCS_N	Output	CIS Chip Select
D7-0	In/Out	Data bus
RESETO_N	Output	Reset output to all other devices on the PC Card
MODEM_PWR DWN	Output	Power Down Mode select for fax/modem
MODEMCS_N	Output	Chip Select from fax/modem chipset
BUF_DIR	Output	Transceiver Control
MODEM_RING	Input	Ring indication from DAA#
SPKRINP	Input	Speaker input from fax/modem
MODEM_INT	Input	Interrupt from fax/modem chipset

Verification Methods

The macro has been fully tested for compatibility with all major applications and Card Services. It has been used in many production fax/modem cards. It has recently been upgraded to use the latest Xilinx unified library elements.

Recommended Design Experience

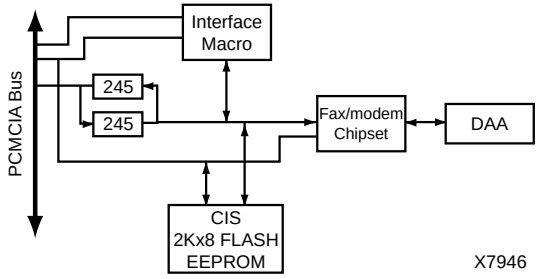
Designers should be familiar with the PCMCIA specification, ViewLogic schematic entry and Xilinx design flows.

Available Support Products

Support products available fromMobile Media Research (further details provided in this product description):

- Xilinx-based PCMCIA Fax/Modem Macro (SW-XM201)
- PCMCIA Prototyping Card (SW-007)
- PCMCIA Card Debugger/Exerciser software
- CIS Generator 1.2 software

Figure 2: PCMCIA Card Using the Fax/Modem Macro



Ordering Information

To purchase or make further inquiries about this or other Inventra products, contact your local Mobile Media Research sales representative.

Related Information

Personal Computer Memory Card International Association

The PCMCIA publishes PC-Card specifications and related documents. For information contact:

PCMCIA Headquarters
2635 North First Street, Suite 209
San Jose, CA 95134 USA
Phone: +1 408-433-CARD (2273)
Fax: +1 408-433-9558
E-mail: office@pcmcia.org
URL: www.pc-card.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Mobile Media Research, Inc.

39675 Cedar Boulevard, #295A
Newark,CA 94560 USA
Phone: 1-800-799-MMRI (6674)
+1 510-657-4891
Fax: +1 510-657-4892
E-mail: sales@mobmedres.com
URL: www.mobmedres.com

Features

- Simple building blocks for user customization
- Built-in PCMCIA Interface
- Support for Function Configuration Registers (FCR)
- Support for external CIS EEPROM
- FCR bit decode implemented by user
- Includes a test primitive which can be used to test socket

General Description

The PCMCIA Library is a library of very simple PCMCIA interface primitives that can be fully customized for a specific user application. The Function Configuration Register (FCR) bits are not decoded and can be implemented in any way necessary. These primitives provide a generic PCMCIA interface and can be customized to work with any I/O device on the PC Card. The library includes a test primitive to help test the PCMCIA socket.

.

AllianceCORE™ Facts

Core Specifics		
Device Family	XC3x00A	
CLBs Used	20-30, see block descriptions	
IOBs Used	32/33[1]	
System Clock f_{max}	10 MHz	
Device Features Used	Not Applicable	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC3x42A TQ100	50/49[2]	114-124
Provided with Core		
Documentation	User Documentation	
Design File Formats	ViewLogic Schematic	
Constraint Files	Not Applicable	
Verification Tool	ViewSim Command Files	
Schematic Symbols	Viewlogic	
Evaluation Model	Prototyping board Available extra	
Reference designs & application notes	None	
Additional Items	None	
Design Tool Requirements		
Xilinx Core Tools	XACTstep 5.2.1/6.0.1	
Entry/Verification Tools	ViewLogic Schematic	
Support		
Support provided by Mobile Media Research		

Notes:
1. IOB count is for CIS, PCMCG1 (both 33 I/O) and PCMCG1 (32 I/O) primitives only; test primitive uses 58 I/O; I/O counts assume all signals are routed off-chip.

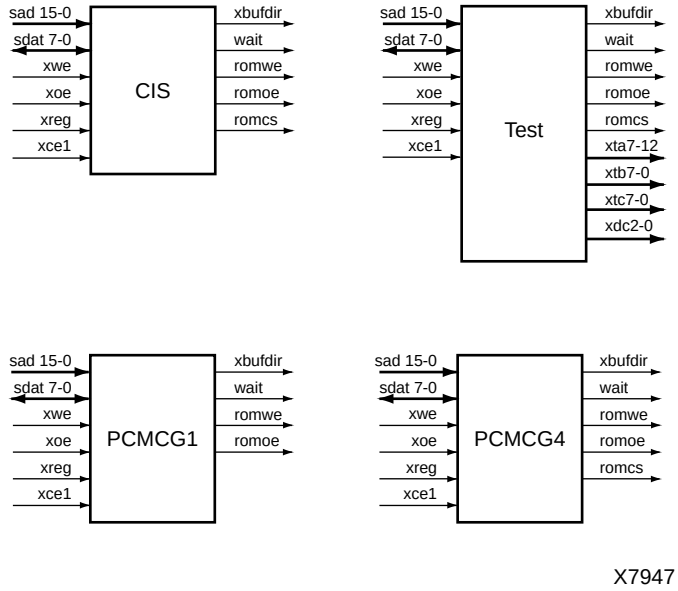


Figure 1: Xilinx PCMCIA Library Primitives

Functional Description

The PCMCIA Library consists of four primitives, as shown in Figure 1, and described below.

CIS Primitive (20 CLBs)

This primitive provides an interface to PCMCIA and allows the host to read/write directly to the CIS ROM on the PC Card. It implements all four Function Configuration Registers without implementing any specific bit decodes. The registers occupy addresses 100h, 102h, 104h and 106h in attribute memory and are read/write.

Test Primitive (28 CLBs)

This primitive is similar to the CIS primitive except that some bits of all four of the PCMCIA Function Configuration Registers are brought out and available for decode and verification. The individual bits of the registers are not decoded but are read/write

PCMCG1 Primitive (20 CLBs)

This primitive is similar to the CIS primitive except that it only implements one Function Configuration Register and CIS accesses are not supported.

PCMCG4 Primitive (30 CLBs)

This primitive is similar to PCMCG1 except that it implements four FCRs

Pinout

The pinout of each primitive has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are provided for each primitive shown in Figure 1, and Table 1 below.

Table 1: PCMCIA Library Primitives Signal Pinout

Signal	Signal Direction	Description
Common Signals for all Primitives		
sad 15-0	Input	Host address
sdat 7-0	In/Out	Host data
xwe	Input	Write Enable
xoe	Input	Output Enable
xreg	Input	Attribute Memory Select from host
xce1	Input	Card Enable from host
xbufdir	Output	Controls direction of data buffer
wait	Output	Inserts delay in completion of cycle to the host
romwe	Output	Write enable to CIS EE-PROM
romoe	Output	Output Enable to CIS EE-PROM
Additional Signals for CIS and PCMCG4 Primitives		
romcs	Output	Chip Select to CIS EEPROM
Additional Signals for Test Primitive		
romcs	Output	Chip Select to CIS EEPROM
xta 7-2	Output	FCR 0 bits 7-2
xtb 7-1	Output	FCR 1 bits 7-1
xtc 7-0	Output	FCR 2 bits 7-0
xdc 2-0	Output	FCR 3 bits 2 -0

Verification Methods

The primitives have been fully tested for compatibility with all major applications and Card Services. They have recently been upgraded to use the latest Xilinx unified library elements.

Recommended Design Experience

Designers should be familiar with the PCMCIA specification, ViewLogic schematic entry and Xilinx design flows.

Available Support Products

Support products available fromMobile Media Research:

- PCMCIA Prototyping Card
- PCMCIA Card Debugger/Exerciser software
- CIS Generator 1.2 software

Ordering Information

To purchase or make further inquiries about this or other Mobile Media Research products, contact MMR directly.

Related Information

Personal Computer Memory Card International Association

The PCMCIA publishes PC-Card specifications and related documents. For information, contact:

PCMCIA Headquarters
2635 North First Street, Suite 209
San Jose, CA 95134 USA
Phone: +1 408-433-CARD (2273)
Fax: +1 408-433-9558
E-mail: office@pcmcia.org
URL: www.pc-card.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

PCMCIA Prototyping Card

February 8, 1998 Product Specification

Mobile Media Research, Inc.

39675 Cedar Boulevard, #295A

Newark,CA 94560 USA
Phone: 1-800-799-MMRI (6674)
+1 510-657-4891
Fax: +1 510-657-4892
E-mail: sales@mobmedres.com
URL: www.mobmedres.com

Features

- Cuts PC-Card development time by half
- All PCMCIA interface components predesigned onto board
  - Xilinx XC3042 or XC3042A in 100-pin TQFP
  - 32Kx8 CIS EEPROM
  - PCMCIA and I/0 connectors
- Includes 12 sq. in of bread board area for I/O devices such as Ethernet, Fax/Modem, etc.
- Separate analog Ground and Vcc for analog devices
- Support for logic analyzer and scope probes to ease debugging of board
- Optional PCMCIA macros for Xilinx for further reduction in PC Card development cycle
- Complete documentation and design support available from Mobile Media Research

General Description

The PC-Card Prototyping Board is intended for PC Card developers and can be used to verify PC-Card logic without spending a considerable amount of time on proper package availability, PCB layout or assembly.

Functional Description

The PC-Card Prototyping Board contains a pre-connected Xilinx XC3042 or XC3042A FPGA in a 100 TQFP package. The PCMCIA signals are connected to the FPGA.

The FPGA provides the interface from the I/O devices to the PCMCIA bus. It also controls the CIS ROM. The Card Configuration registers are implemented on the XC3042. This requires that address decoding, reset and interrupts all go through the FPGA. The unconnected pins on the FPGA are brought out to test points from where they can easily be wired to other devices.

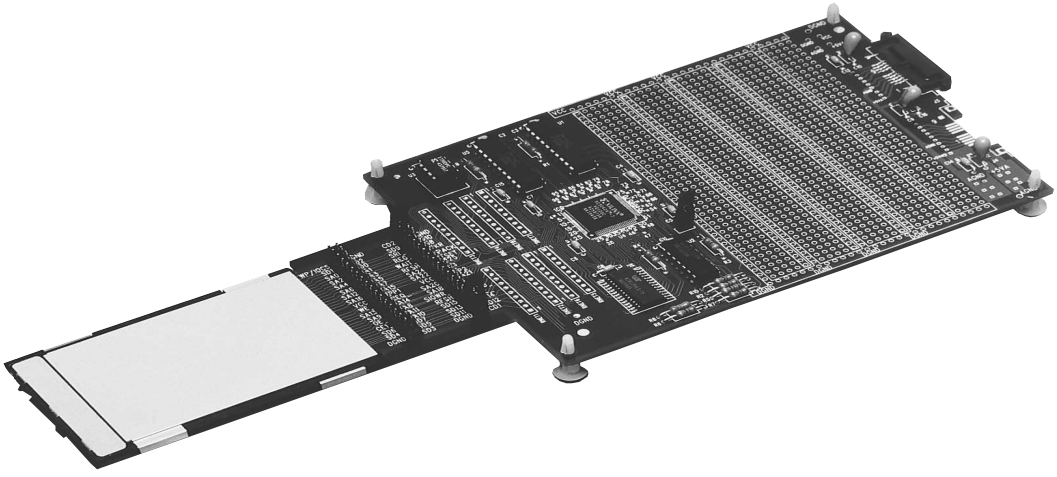


Figure 1: PCMCIA Prototyping Board

The FPGA can be configured either from the XC1736 PROM or from a host via a download cable. The jumper J1 allows the user to select the programming option.

A 12 square-inch breadboard area is provided for I/O devices. Up to four memory or I/O devices can be put onto the bread-board area. Power and ground strips are provided throughout to support additional devices. Separate analog Ground and Vcc are provided for I/O devices requiring analog signals.

Also included on the board is a logic analyzer strip to assist in tracing hardware or software problems. All PCMCIA signals are routed to the logic analyzer strip. Other signals (from the FPGA, CIS or I/O) can easily be accessed either through test points or in the breadboard area.

Documentation

The following documentation is provided with the PC-Card Prototyping Card:

- Board schematic
- Board specifications
- Application brief on developing PC-Cards
- Sample modem card CIS
- A list of soft macros for the Xilinx device available from Mobile Media Research.

Available Support Products

Mobile Media Research supplies a complete line of PCMCIA design products. Contact DO for additional information.

- PCMCIA Fax/Modem Macro
- PCMCIA Library V1.2
- CIS Generator 1.2 software

Ordering Information

This product is available from the AllianceCORE™ partner listed on the first page. Please contact the partner for pricing and more information.

Related Information

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Mobile Media Research, Inc.

39675 Cedar Boulevard, #295A
Newark,CA 94560 USA
Phone: 1-800-799-MMRI (6674)
+1 510-657-4891
Fax: +1 510-657-4892
E-mail: sales@mobmedres.com
URL: www.mobmedres.com

Overview

Mobile Media Research, Inc. PCMCIA Card Debugger/Exerciser is designed for both hardware and software engineers in the development phase of a PCMCIA product.

Professional and Lite versions are available depending upon the needs and budget of the user. The features of each are listed below.

Lite Version Features (SW-002-LV)

- Engineering debug software for PCMCIA hosts and cards
  - Supports 82365SL-compatible based hosts
- Fully compatible Microsoft Windows-based GUI
  - Provides easy control of host functions
  - Generates PCMCIA bus cycles on the selected socket
  - Allows direct access to PCIC registers
  - Integrates all functionality necessary for PCMCIA compatibility testing in one screen

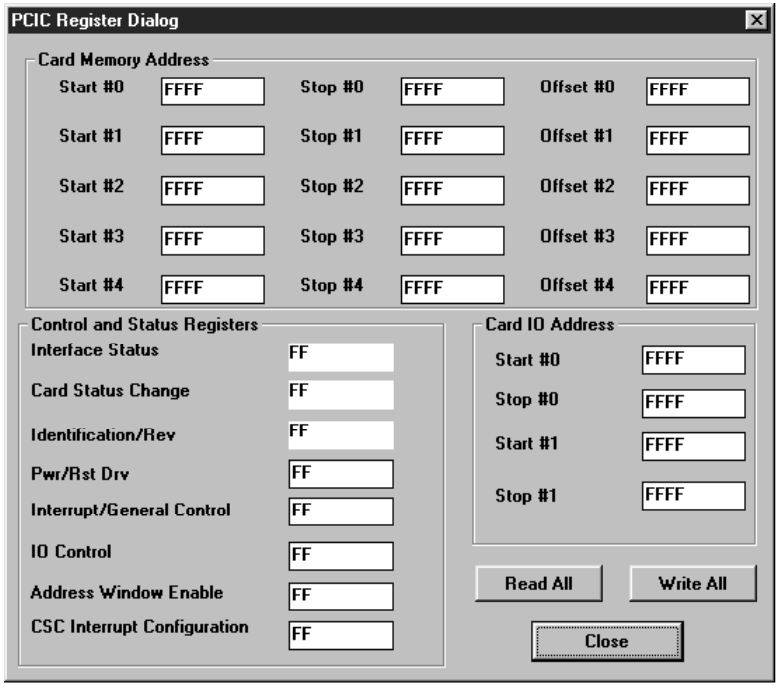


Figure 1: PCMCIA Card Debugger/Exerciser Window

- Supports basic and necessary PCMCIA debug and testing features:
  - Register access
  - I/O and memory window set-up
  - Socket configuration access

Professional Version Features (SW-002-PV)

- All features of Lite version plus these debug and test features:
  - Card Configuration Register support, up to 16 registers, especially useful for multi-function cards
  - ATA card debug module (supports ATA interface)
  - Built-in CIS Parcel, with tuple browser to read the CIS
- Built-in command Macro/Interpreter
  - Supports user programmable “Macro” commands to generate system cycles for PCMCIA host/card debug
  - Up to eight user-defined command lines
  - Trace, Normal and Step options
  - Loop control capability
  - Results displayed in monitor window
- Excellent for high-volume testing
- Ideal for Host or PC-Card debug, characterization and evaluation

Available Support Products

Support products available from Mobile Media Research, Inc. (further details provided in this product description)

- PCMCIA Prototyping Card
- CIS Generator 1.2 software

Ordering Information

To purchase or make further inquiries about this or other MMRI products, contact Mobile Media Research, Inc. directly.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Mobile Media Research, Inc.

39675 Cedar Boulevard, #295A
Newark,CA 94560 USA
Phone: 1-800-799-MMRI (6674)
+1 510-657-4891
Fax: +1 510-657-4892
E-mail: sales@mobmedres.com
URL: www.mobmedres.com

Features

- Windows 3.1 tool to enter and generate a PCMCIA compatible CIS
- Graphical entry tool for design of PCMCIA Card Information Structures (CIS)
- Generates Level One Tuple chains
- Outputs Hex image of the CIS to a file. The file can then be written to PC-Card attribute memory space to implement the CIS
- Provides options for PCMCIA recommended minimum Tuples or user defined Tuple chain
- Provides the option of generating a Tuple chain compatible with Microsoft Windows 95 or SystemSoft's *CardSoft* card services
- Option for support of minimum CIS configurations of various PCMCIA memory and I/O cards
- Form-based layout of Tuples, allows users to enter CIS information logically and in a structured manner
- Greatly reduces CIS development time

General Description

In PCMCIA, one of the biggest problems engineers face is the design and implementation of the PC-Card Card Information Structure (CIS). It is a tedious job, similar to writing CPU machine code. Doing this manually can significantly affect productivity, increase the number of mistakes and thereby affect time-to-market. The PC-Card market is replete with products which are declared incompatible because of wrong or improperly designed CIS.

Mobile Media Research, Inc. solution is the CIS Generator which will allow the user to compile the CIS using visual forms instead of error-prone manual methods.

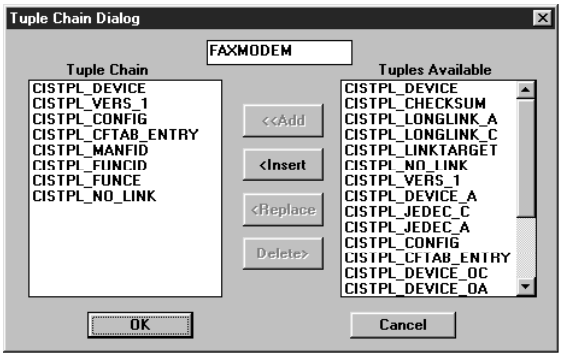


Figure 1: Tuple Chain Dialog Box

Available Support Products

Mobile Media Research, Inc. supplies a complete line of hardware and software products for PCMCIA design. Contact them directly for additional information.

- PCMCIA Fax/Modem Macro (SW-XM201)
- PCMCIA library R1.2 (SW-XL200)
- PCMCIA Prototyping Card (SW-007)
- PCMCIA Card Debugger/Exerciser (SW-002)

Ordering Information

This product is available from the AllianceCORE™ partner. Please contact the partner for pricing and more information.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com

URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Inventra™

A Business Unit of Mentor Graphics
1001 Ridder Park Drive
San Jose, CA 95134-2314 USA
URL: www.mentorg.com/inventra

Features

- Fully compliant to USB 1.0 specification
- 100% programmable single-chip solution with customizable back-end functionality
- Pre-defined implementation for predictable timing in Xilinx FPGA or HardWire™
  - Push button scripts to place and route, and generate Xilinx bit files
  - Xilinx-optimized place and route constraint and guide files
- Fully verified design
  - Simulated using Inventra USB Simulation Model
  - Hardware-proven in a Xilinx FPGA at USB-IF sponsored interoperability workshop
  - Complete synchronous design
- Provides a high level interface that shields the firmware from USB protocol details
- Complete device configuration
- Compatible with both OpenHCI and Intel UHCI standards
- Supports low-speed (1.5 Mbps) functions
- Endpoint 0 is Control endpoint
  - Maximum packet capacity is 8 Bytes
- IN Endpoint for Interrupt data transfers
  - Maximum packet capacity for In Endpoint is user selectable from 1-8 Bytes
- Automatic Data Retry, Error recovery, and Data Toggle synchronization performed in hardware
- Includes the following error handling capabilities
  - CRC errors
  - Response Time Out
- ID error
- External interface to Phillips *IPDIUSBP11* USB transceiver

AllianceCORE™ Facts

Core Specifics		
Device Family	XC4000E	
CLBs Used	600	
IOBs Used	29[1]	
System Clock f_{max}	6MHz[2]	
Device Features Used	RAM, 3-state buses, carry logic	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC4025E-4 HQ240	164[1]	304
Provided with Core		
Documentation	XC4000E/CX5215 Datasheets Core documentation Sample files for top level module in Verilog HDL	
Design File Formats	XNF Netlist Verilog Source RTL Available	
Constraint Files	Timespec, .cst, .tnm files	
Verification Tool	Verilog	
Schematic Symbols	Viewlogic, ORCAD	
Evaluation Model	Evaluation Board available extra	
Reference designs & application notes	None	
Additional Items	Firmware for microcontroller available for nominal cost	
Design Tool Requirements		
Xilinx Core Tools	XACTstep 5.2.1/6.0.1	
Entry/Verification	Verilog RTL	
Support		
Support provided by Inventra		

Notes:

1. Assuming all core signals are routed off-chip.
2. For 10-15% of design, remaining logic operates at 1/4 max clock.

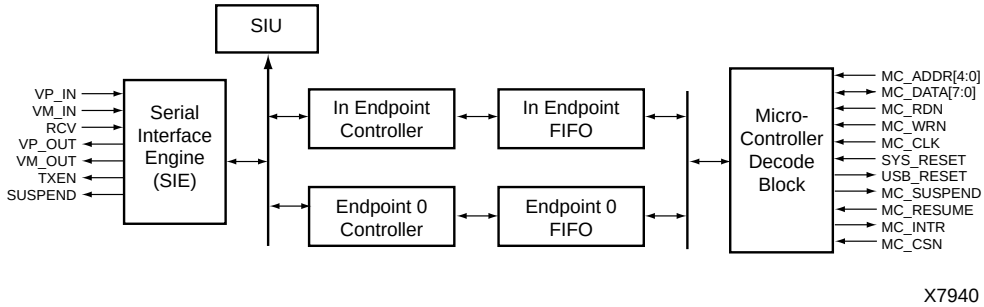


Figure 1: Low-Speed USB Function Controller Block Diagram

Applications

- Low speed (1.5 Mbps) functions requiring a microcontroller interface
- Embedded applications in telecommunication, industrial, medical or point-of-sale systems

General Description

The Low-Speed USB Function Controller is flexible, and has been optimized for a variety of low-speed, microcontroller-based applications. It includes all functionality for a complete function controller interface using one Xilinx FPGA plus an external Phillips IPDIUSBP11 USB transceiver.

Functional Description

The USB Function Controller core is partitioned into six modules as shown in Figure 1 and described below. A sample top-level Verilog RTL file is provided so the user can instantiate the netlist for synthesis and simulation environments.

Serial Interface Engine (SIE)

The SIE handles NRZI decoding/encoding, CRC generation and checking, bit-stuffing and timeout. The SIE generates the PID, deciphers packets, and transmits and receives packets from the host. The SIE uses a 6 MHz clock for data and clock extraction. It also provides the interface signals for the Phillips IPDIUSBP11 USB transceiver.

Serial Interface Unit (SIU)

This block handles endpoint address decoding for USB packets.

Endpoint 0 Controller

The Endpoint 0 controller handles control transfers.

In Endpoint Controller

The IN Endpoint controller handles TX data transfer between host and function. The IN Endpoint is configured as an interrupt endpoint.

FIFO

The core has a bi-directional 8-Byte Endpoint 0 FIFO to handle control transfers. It also has an IN Endpoint FIFO with a maximum 8-Byte capacity to handle interrupt data from the microcontroller.

Microcontroller Decode Block

The microcontroller interface is generic, with Address and Data bus interfaces, and Read and Write control signals. It generates an interrupt to the micro controller when data is ready, and when data has been successfully transmitted. The core performs hardware retries and data buffering. This improves performance by reducing the burden on the microcontroller. The microcontroller does address decoding for internal registers (i.e. FIFO Data Register).

This interface is asynchronous. All signals (MC_WRN, MC_RDN) are synchronized internally. MC_WRN and MC_RDN are active low.

Core Modifications

The Low-Speed USB Function Controller Core is modular in design, making modifications relatively simple. If you are interested in obtaining a version of the core that differs from this product description, then contact Inventra directly. Inventra can provide custom versions of the core, including support for the following:

- Changing Endpoint FIFO depths.
- Flexible interfaces for Mitsubishi 37690, Zilog Z80 and Atmel 89C51 microcontrollers.
- Modification for audio- or video-specific applications.
- Support for full-speed functions is provided by the Full-Speed Function Controller Core, also available from Inventra. A similar product description for that function is available from both Inventra and Xilinx.

Pinout

The pinout of the Low-Speed USB Function Controller has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are provided in the block diagram shown in Figure 1 and Table 1 below.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Transceiver Interface Signals		
VP_IN	Input	D+ input from XCVR
VM_IN	Input	D- input from XCVR
RCV	Input	Differential data from XCVR
VP_OUT	Output	D+ output from XCVR
VM_OUT	Output	D- output from XCVR
TXEN	Output	Enable for XCVR, active low
SUSPEND	Output	Suspend signal, puts XCVR into suspend mode
Microprocessor Interface Signals		
MC_ADDR [4:0]	Input	Microcontroller Address bus
MC_DATA [7:0]	In/Out	Microcontroller Data bus
MC_RDN	Input	Read Strobe, active low
MC_WRN	Input	Write Strobe, active low
MC_CLK	Input	6 MHz microcontroller clock input
SYS_RESET	Input	System Reset, active high
USB_RESET	Output	USB Reset, active low
MC_SUSPEND	Output	Interrupt signal generated during SUSPEND signal on USB, active high; provided as dedicated bit in Power Mgmt register
MC_RESUME	Output	Interrupt signal generated during RESUME signal on USB, active high
MC_INTR	Output	Microcontroller Interrupts, active low
MC_CSN	Input	Acts as a Block Select to microcontroller address, active low

Table 2: Microcontroller I/O Timing

Signal	Setup	Hold
MC_ADDR	3 ns	0 ns
MC_DATA	3 ns	0 ns
MC_WRN	3 ns	0 ns
MC_RDN	3 ns	0 ns

Verification Methods

The Low-Speed USB Function Controller core has been tested with products from over 30 system manufacturers at USB-IF sponsored compliance workshops. The Xilinx-based implementation of the core passed all interoperability testing with numerous host, BIOS and peripheral products.

The core has undergone extensive testing using Inventra's USB Simulation Model that includes a host controller, function controller and protocol analyzer. This model is available separately from Inventra.

Recommended Design Experience

Knowledge of the USB specification is required. The user must be familiar with HDL design methodology as well as installation of Xilinx netlists in a hierarchical design environment.

Available Support Products

Inventra supplies a complete line of hardware and software products designed to aid integration of this core into your application. These are available for additional cost. Contact Inventra for more information.

- USB Function Evaluation Board
- USB Simulation Model

Ordering Information

This product is available from the AllianceCORE™ partner listed on the first page. Please contact the partner for pricing and more information.

Related Information

Universal Serial Bus Implementor's Forum

The USB-IF publishes USB specifications and related documents:

- USB Specification, Rev. 1.0
- USB Compliance Checklist
- USB Device Class Definitions for Human, Audio, Video and Mass Storage devices

Contact:
USB Implementor's Forum
URL: www.usb.org

Philips Semiconductor

For additional information on the Philips *IPDIUSBP11* USB transceiver chip, contact:

Philips Semiconductors
URL: www.semiconductors.philips.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

The Power to Create

Inventra™

A Business Unit of Mentor Graphics
1001 Ridder Park Drive
San Jose, CA 95134-2314 USA
URL: www.mentorg.com/inventra

Features

- Fully compliant to USB 1.0 specification
- 100% programmable single-chip solution with customizable back-end functionality
- Pre-defined implementation for predictable timing in Xilinx FPGA or HardWire™
  - Push button scripts to place and route, and generate Xilinx bit files
  - Xilinx-optimized place and route constraint and guide files
- Fully verified design
  - Simulated using Inventra USB Simulation Model
  - Xilinx FPGA-proven in hardware at USB-IF Plug-Fest workshop
  - Complete synchronous design
- Provides a high level interface that shields the firmware from USB protocol details
- Complete device configuration
- Compatible with both OpenHCI and Intel UHCI standards
- Supports full-speed (12 Mbps) functions
- Support for 3 Endpoints (0, In and Out)
  - On-chip Endpoint FIFOs
  - Each can handle Bulk, Interrupt/Status Change and Isochronous data transfers
  - Endpoints interface to a microcontroller
  - Maximum packet capacity for Endpoint 0 is 8-Bytes
  - Packet capacity for In/Out Endpoints is selectable at 8-, 16- or 32-Bytes
- Flexible interface for Mitsubishi 37690, Zilog Z80 or Atmel 89C51 microcontrollers
  - Automatic Data Retry, Error recovery, and Data Toggle synchronization performed in hardware.

AllianceCORE™ Facts

Core Specifics	
Device Family	XC4000E
CLBs Used	720[1]
IOBs Used	29[2]
System Clock f_{max}	48 MHz[3]
Device Features Used	RAM, 3-state buses, carry logic

Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC4025E-3 HQ240	164[2]	304

Provided with Core	
Documentation	XC4000E Datasheet Core documentation Sample files for top level module in Verilog HDL
Design File Formats	XNF Netlist Verilog Source RTL Available
Constraint Files	Timespec, .cst and .tnm files
Verification Tool	Verilog
Schematic Symbols	Viewlogic, ORCAD
Evaluation Model	Evaluation Board available extra
Reference designs & application notes	None
Additional Items	Firmware for microcontroller available for nominal cost

Design Tool Requirements	
Xilinx Core Tools	XACTstep 5.2.1/6.0.1
Entry/Verification Tools	Verilog RTL

Support
Support provided by Inventra

Notes:
1. CLB utilization for configuration with 3 endpoints, on-chip FIFOs and 32-Bytes per in/out endpoints
2. MAX I/O with on-chip FIFOs, assuming all core signals are routed off-chip; 51 IOBs if using external DMA.
3. For 10-15% of design, remaining logic operates at 1/4 max clock rate

- Includes the following error handling capabilities:
  - CRC errors

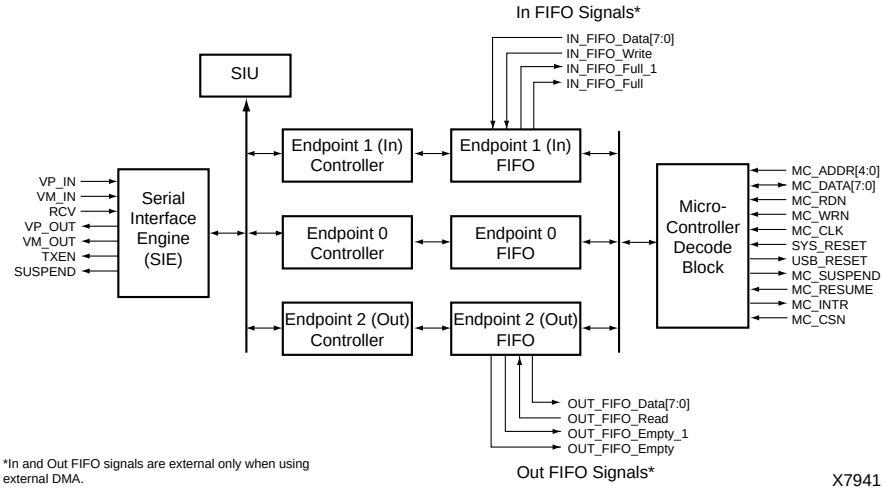


Figure 1: Full-Speed USB Function Controller Block Diagram

- Response Time Out
- ID errorNotes

Applications

- High-end computer peripheral equipment such as laser printers, plotters and high-speed telecommunications equipment.
- Embedded applications in telecommunication, industrial, medical or point-of-sale systems.

General Description

The Full-Speed USB Function Controller is flexible and can be used in a variety of applications. It includes all functionality for a complete function controller interface using one Xilinx FPGA plus an external Philips IPDIUSBP11 USB transceiver. The user can assign control of any endpoint to a microcontroller or external logic.

Functional Description

The USB Function Controller core is partitioned into modules as shown in and described below. Xilinx netlists are provided for each module.

Serial Interface Engine (SIE)

This block handles NRZI decoding/encoding, CRC generation and checking and bit-stuffing. It also provides the interface signals for an external Philips IPDIUSBP11 USB transceiver.

SIU

This block handles endpoint address decoding for USB packets.

Endpoint 0 Control

The Endpoint 0 Controller handles control transfers.

Endpoint 1 Control

The Endpoint 1 Controller is configured as an In Endpoint and handles TX data transfers between host and function. It can be configured to handle either Bulk, Isochronous or Interrupt/Status data transfers.

Endpoint 2 Control

The Endpoint 2 Controller is configured as an Out Endpoint. It can be configured to handle either Bulk, Isochronous or Interrupt/Status data transfers.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Transceiver Interface Signals		
VP_IN	Input	D+ input from XCVR
VM_IN	Input	D- input from XCVR
RCV	Input	Differential data from XCVR
VP_OUT	Output	D+ output for XCVR
VM_OUT	Output	D- output for XCVR
TXEN	Output	Enable for XCVR, active low
SUSPEND	Output	Suspend signal, puts XCVR into suspend mode
Microprocessor Interface Signals		
MC_ADDR [4:0]	Input	Microcontroller Address bus
MC_DATA [7:0]	In/Out	Microcontroller Data bus
MC_RDN	Input	Read Strobe, active low
MC_WRN	Input	Write Strobe, active low
MC_CLK	Input	48 MHz microcontroller clock input
SYS_RESET	Input	System Reset, active high
USB_RESET	Output	USB Reset, active low
MC_ SUSPEND	Output	Interrupt signal generated during SUSPEND signal on USB, active high; provided as dedicated bit in Power Mgmt register
MC_ RESUME	Output	Interrupt signal generated during RESUME signal on USB, active high
MC_INTR	Output	Microcontroller Interrupts, active low
MC_CSN	Input	Acts as a Block Select to microcontroller address, active low
FIFO Interface Signals (used only when interfacing to external DMA)		
IN_FIFO_ Data [7:0]	Input	TX data input to In FIFO
IN_FIFO_ WRT	Input	In FIFO write strobe
IN_FIFO_ FULL	Output	Indicates In FIFO is full
IN_FIFO_ FULL_1	Output	Indicates In FIFO can accept only one more Byte
OUT_FIFO_ Data [7:0]	Output	RX data received from the host
OUT_FIFO_ RD	Output	Out FIFO read strobe
OUT_FIFO_EM PTY	Output	Indicates Out FIFO is empty
OUT_FIFO_EM PTY_1	Output	Indicates only one Byte is available in Out FIFO

FIFOs

The core has a bidirectional 16 Byte Endpoint 0 FIFO that handles standard and class specific descriptors. It interfaces to the microcontroller, which handles descriptor decoding and specified descriptor actions.

The In and Out FIFOs are 32-Bytes deep and uni-directional for TX and RX data transfers, respectively. The maximum packet size for In and Out tokens is register-selectable at 8/16/32-Bytes.

The In and Out FIFOs can be interfaced either to the microcontroller or to an external DMA channel. When using external DMA, the microcontroller writes to an internal register bit that disables microcontroller access to the registers and enables external access.

Endpoint FIFOs are implemented using XC4000E Select RAM™. For applications that need larger FIFOs, the FIFO can be off-chip. Inventra will customize the core for users.

Microcontroller Decode Block

The microcontroller interface is generic, with Address and Data bus interfaces, and Read and Write control signals. It generates an interrupt to the microcontroller when data is ready, and when data has been successfully transmitted. The core performs hardware retries and data buffering. This improves performance by reducing the burden on the microcontroller. The microcontroller does address decoding for internal registers (i.e. FIFO Data Register).

This interface is asynchronous. All signals (MC_WRN, MC_RDN) are synchronized internally. MC_WRN and MC_RDN are active low.

This block is not required if the outputs of the FIFOs are connected directly to a DMA controller.

Table 2: Microcontroller I/O Timing

SIGNAL	Setup	Hold
MC_ADDR	3 ns	0 ns
MC_DATA	3 ns	0 ns
MC_WRN	3 ns	0 ns
MC_RDN	3 ns	0 ns

Core Modifications

The Full-Speed USB Function Controller Core is modular in design, making modifications relatively simple. If you are interested in obtaining a version of the core that differs from

this product description, then contact Inventra directly. Inventra can provide custom versions of the core, including support for the following:

- Changing Endpoint FIFO depths.
- Modification for audio- or video-specific applications.
- Support for low-speed functions is provided by the Low-Speed Function Controller Core, also available from Inventra. A similar product description for that function is available from both Inventra and Xilinx.

Pinout

The pinout of the Full-Speed USB Function Controller has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are shown in the block diagram in Figure 1, and in Table 1.

Verification Methods

The Full-Speed USB Function Controller core has been tested with products from over 30 system manufacturers at USB-IF sponsored compliance workshops. The Xilinx-based implementation of the core passed all interoperability testing with numerous host, BIOS and peripheral products.

The core has undergone extensive testing using Inventra USB Simulation Model that includes a host controller, function controller and protocol analyzer. This model is available separately from Inventra.

Recommended Design Experience

Knowledge of the USB specification is required. The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment.

Available Support Products

Inventra supplies a complete line of hardware and software products designed to aid integration of this core into your application. These are available for additional cost. Contact Inventra for more information.

- USB Function Evaluation Board
- USB Simulation Model

Ordering Information

This product is available from the AllianceCORE™ partner listed on the first page. Please contact the partner for pricing and more information.

Related Information

Universal Serial Bus Implementor's Forum

The USB-IF publishes USB specifications and related documents:

- USB Specification, Rev. 1.0
- USB Compliance Checklist
- USB Device Class Definitions for Human, Audio, Video and Mass Storage devices

Contact:

USB Implementor's Forum
URL: www.usb.org

Philips Semiconductor

For additional information on the Philips *IPDIUSBP11* USB transceiver chip, contact:

Philips Semiconductors
URL: www.semiconductors.philips.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

3-Port USB Hub Controller

The Power to Create

Inventra™

A Business Unit of Mentor Graphics
1001 Ridder Park Drive
San Jose, CA 95134-2314 USA
URL: www.mentorg.com/inventra

Features

- Fully compliant to USB 1.0 specification
- 100% programmable single-chip solution
- Pre-defined implementation for predictable timing in Xilinx FPGA or HardWire™
  - Push button scripts to place and route, and generate Xilinx bit files
  - Xilinx-optimized place and route constraint and guide files
- Fully verified design
  - Simulated using Inventra USB system simulation model
  - Hardware-proven in a Xilinx FPGA at USB-IF sponsored interoperability workshop
- Compatible with both OpenHCI and Intel UHCI standards
- Pre-configured with 1 up-stream and 3 down-stream ports
- Expandable down-stream ports
- Fault detection for up-stream traffic
- Detects control and status change endpoints
- Supports Suspend and Resume signaling
- Handles all Hub Class device descriptors
- Capable of handling remote wake-up
- Supports down-steam power control
- Supports a mixture of full- and low-speed ports
- Provides an ideal compound-device solution when coupled with either low- or full-speed Function Controller cores, also available from Inventra
- Xilinx-based hub evaluation board is available from Inventra
- External interfaces to Philips *IPDIUSBP11* USB transceivers

AllianceCORE™ Facts

Core Specifics		
Device Family	XC4000E	
CLBs Used	960	
IOBs Used	53[1]	
System Clock f_{max}	48Mhz[2]	
Device Features Used	RAM, Carry Logic	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC4025E-3 HQ240	140	64
Provided with Core		
Documentation	XC4000E Datasheets Core documentation Sample files for top level module in Verilog HDL	
Design File Formats	XNF Netlist Verilog Source RTL Available	
Constraint Files	Timespec, .cst, .tnm files	
Verification Tool	Verilog	
Schematic Symbols	Viewlogic, ORCAD	
Evaluation Model	Prototype board to implement Hub controller	
Reference designs & application notes	None	
Additional Items	Firmware for microcontroller available for nominal cost	
Design Tool Requirements		
Xilinx Core Tools	XACTstep 5.2.1/6.0.1	
Entry/Verification Tools	Verilog RTL	
Support		
Support provided by Inventra		

Notes:
1. Assuming all core signals are routed off-chip.
2. For 10-15% of design, remaining logic operates at 1/4 max clock rate.

Applications

- Compound Devices, with multiple functionality
- Telecommunication, PC peripheral, embedded applications

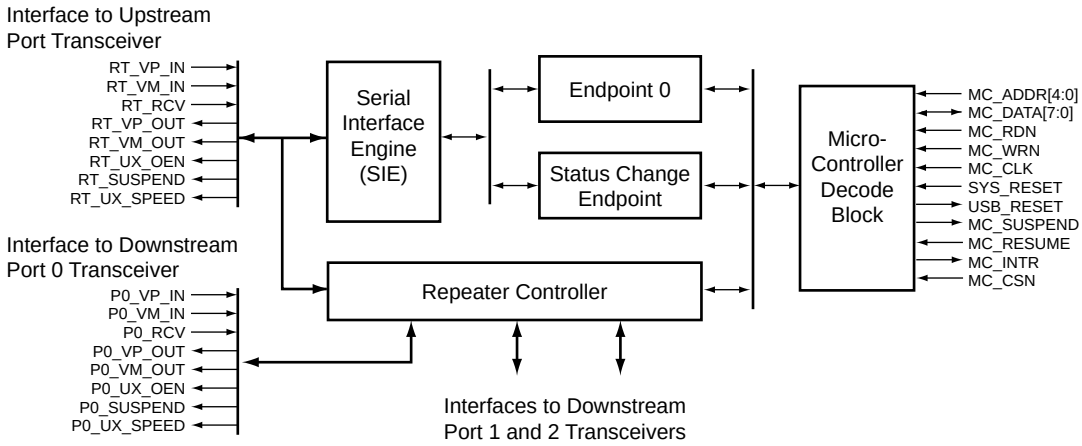


Figure 1: 3-Port USB Hub Controller Block Diagram

Functional Description

The USB Hub Controller core is partitioned into modules as shown in and described below.

Serial Interace Engine (SIE)

This block handles NRZI decoding/encoding, CRC generation and checking and bit-stuffing. It also provides the interface signals for an external Philips IPDIUSBP11 USB transceiver.

Endpoint 0

This block handles standard and hub class specific descriptors. The micro controller decodes the descriptors, and sets appropriate control bits in the repeater.

Status Change Endpoint

This block maintains the port connectivity of each down-stream port, and notifies the host controller if a device is connected or disconnected from any of the downstream ports.

Repeater Controller

This block controls traffic to and from the ports. The micro-controller sets the different states of the repeater as specified by the USB Specification. The repeater block handles:

- Detection of low-/full-speed devices connected.
- Low-speed signaling to & from down stream ports
- Suspend and Reset signaling from the root port to the down stream ports
- Resume signaling from the down stream functions to the Host.

Microcontroller Interface

This block interfaces to any standard 8 Bit micro controller. The microcontroller is used for descriptor decoding, and also adds compound device capabilities to the hub.

Core Modifications

Inventra will provide core modifications to add or remove downstream ports, or to provide a compound device.

Pinout

The pinout of the USB Hub Controller has not been fixed to specific FPGA I/O, allowing flexibility with a user's application. Signal names are provided in the block diagram shown in Table 1 and Figure 1.

Verification Methods

The USB Hub Controller core has been tested with products from over 30 system manufacturers at USB-IF sponsored compliance workshops. The Xilinx-based implementation of the core passed all interoperability testing with numerous host, BIOS and peripheral products.

The core has undergone extensive testing using Inventra's USB Simulation Model that includes a host controller, function controller and protocol analyzer. This model is available separately from Inventra.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Upstream Port Root Signals		
RT_VP_IN	Input	D+ input from the XCVR
RT_VM_IN	Input	D- input from the XCVR
RT_RCV	Input	Differential data from XCVR
RT_VP_OUT	Output	D+ output to the XCVR
RT_VM_OUT	Output	D- Output to the XCVR
RT_UX_OEN	Output	Enable for XCVR, active low
RT_SUSPEND	Output	Suspend signal, puts XCVR into suspend mode
RT_UX_SPEED	Output	Selects signaling speed on USB
Downstream Port Interface Signals		
P[0:2]_VP_IN	Input	D+ input from the XCVR
P[0:2]_VM_IN	Input	D- Input from the XCVR
P[0:2]_RCV	Input	Differential data from XCVR
P[0:2]_VP_OUT	Output	D+ output to the XCVR
P[0:2]_VM_OUT	Output	D- output to the XCVR
P[0:2]_UX_OEN	Output	XCVR Enable
P[0:2]_SUSPEND	Output	SUSPEND signal monitor
P[0:2]_UX_SPEED	Output	Selects signaling speed on USB
Microprocessor Interface Signals		
MC_ADDR [4:0]	Input	Microcontroller Address bus
MC_DATA [7:0]	In/Out	Microcontroller Data bus
MC_RDN	Input	Read Strobe, active low
MC_WRN	Input	Write Strobe, active low
MC_CLK	Input	48 MHz microcontroller clock input
SYS_RESET	Input	System Reset, active high
USB_RESET	Output	USB Reset, active low
MC_SUSPEND	Output	Interrupt signal generated during SUSPEND signal on USB, active high; provided as dedicated bit in Power Mgmt register
MC_RESUME	Output	Interrupt signal generated during RESUME signal on USB, active high
MC_INTR	Output	Microcontroller Interrupts, active low
MC_CSN	Input	Acts as a Block Select to microcontroller address, active low

Related Information

Universal Serial Bus Implementor's Forum

The USB-IF publishes USB specifications and related documents:

- USB Specification, Rev. 1.0
- USB Compliance Checklist
- USB Device Class Definitions for Human, Audio, Video and Mass Storage devices

Contact:

USB Implementor's Forum
URL: www.usb.org

Philips Semiconductor

For additional information on the Philips *IPDIUSBP11* USB transceiver chip, contact:

Philips Semiconductors
URL: www.semiconductors.philips.com

Recommended Design Experience

Knowledge of the USB specification is required. The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment.

Available Support Products

Inventra supplies a complete line of hardware and software products designed to aid integration of this core into your application. These are available for additional cost and are described below. Contact Inventra for more information.

- USB Simulation Model
- Hub Controller Evaluation Board

Ordering Information

This product is available from the AllianceCORE™ partner listed on the first page. Please contact the partner for pricing and more information.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Mentor Graphics
The Power to Create

Inventra™

A Business Unit of Mentor Graphics
1001 Ridder Park Drive
San Jose, CA 95131-2314 USA
URL: www.mentorg.com/inventra

Features

Use to evaluate and prototype a Inventra USB function core in a Xilinx XC4000E FPGA

- Flexible microcontroller interface with built-in support for
  - Atmel AT89C51/AT89C52
  - Mitsubishi M37690
  - Any other 8-bit MCU via 50 pin connector
- 64 KB socketed FLASH
- 32 KB socketed SRAM
- 4KB Fast SRAM for endpoint FIFO expansion
- Flexible power connection
- Reset and Break switches
- LEDs
- Extra 223-pin PGA socket for Xilinx XC4000E FPGA
  - Use for additional custom logic
- Wirewrap area for prototyping
- Firmware implements Chapter 9 of USB Specification 1.0

General Description

The USB Function Evaluation Board can be used to evaluate or prototype a complete USB function using any of Inventra's USB Function Controller cores and a Xilinx FPGA.

This board can be powered either by a PC disk connector or a 12 Volt DC adapter. A jumper is provided to use either micro controller's internal program space or an external PROM.

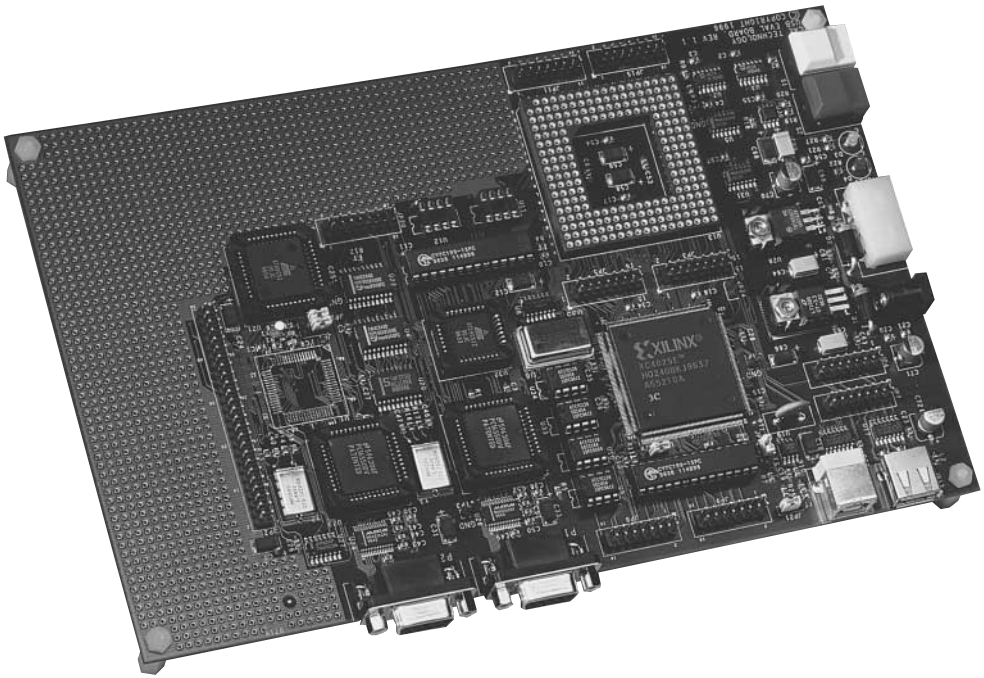


Figure 1: USB Function Evaluation Board

There are two serial ports available on the board. Port 1 is used to get debug information. The serial port can be connected to any terminal. (Usually a PC running terminal emulation software like PROCOMM or QLFAX).

An upstream port is used to connect to the host computer while the downstream port can be used to connect to another device.

Available Support Products

Inventra supplies a complete line of hardware and software products for USB system development. Contact Inventra for more information.

- USB Simulation Model
- USB Cores for host, hub and function controllers.

Ordering Information

To purchase or make further inquiries about this or other Inventra products, contact your local Mentor Graphics sales representative, or Inventra directly.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

USB Hub Evaluation Board

February 8, 1998 Product Specification

Inventra™

A Business Unit of Mentor Graphics
1001 Ridder Park Drive
San Jose, CA 95134-2314
URL: www.mentorg.com/inventra

Features

- Controls four downstream ports
- Supports full- and low-speed devices
- Individual port power control
- Individual over-current detection
- Microcontroller or hard-wired hub control support
- Activity LEDs for upstream and four downstream ports
- Firmware for microcontrolled hub
- Reset switch
- Easy power connection
- Debug header pins

General Description

The USB Hub Evaluation Board can be used to evaluate or prototype a USB hub controller core from Inventra using a Xilinx XC4000 FPGA. Presently two flavors of hub controller core are available, one with Mitsubishi M37690 microcontroller support and one hard-wired without a microcontroller interface.

Downstream ports support full/low speed control, power switching, over-current detection and activity LED. Activity LED turns on when port is first enabled and from then on it toggles after every transaction from the port. Upstream port also has one LED to indicate host activity to help in debug.

Available Support Products

Inventra supplies a complete line of hardware and software products for USB system development. Contact Inventra for additional information.

- USB Simulation Models
- USB Cores for host, hub and function controllers

Ordering Information

This product is available from the AllianceCORE partner listed on this page. Please contact the partner for pricing and additional information.

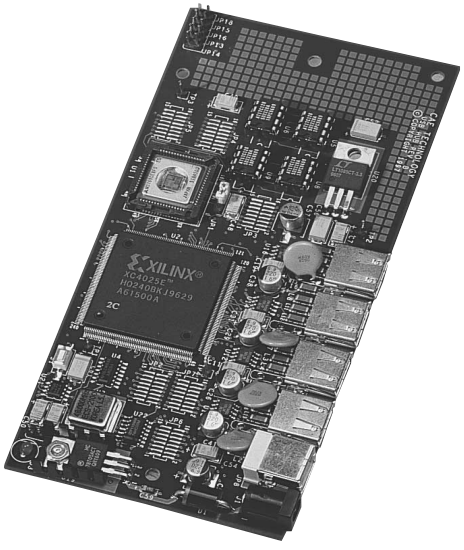


Figure 1: USB Hub Evaluation Board

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Inventra™

A Business Unit of Mentor Graphics
1001 Ridder Park Drive
San Jose, CA 95134-2314 USA
URL: www.mentorg.com/inventra

Introduction

Inventra is a leading provider of simulation and core logic macros. Xilinx and Inventra provide a comprehensive Universal Serial Bus (USB) solution. The USB Core logic saves many man months of design and integration time. Software solutions along with simulation models further shorten design schedules.

Features

- Complete functionality for USB
  - USB Host Controller
  - USB Hub Controller
  - USB Function Controller
- Complete system simulation capability
- Extensive software library available
- Includes compliance test suite
- Software debug capability
  - Provides user capability to debug system drivers

Functional Description

The USB simulation model provides full system level simulation capability. This includes the ability to generate USB traffic for multiple functions simultaneously, including an unlimited mixture of high- and low-speed devices. The user has the ability to add hubs and support features such as Connect, Disconnect, Suspend and Resume.

Host Controller Model Features

- Provides high level user interface
  - Get Status, Clear Status, Get Descriptor, Set Descriptor, etc.
  - User does not have to generate traffic on frame-by-frame basis
- Multiple traffic streams can be started simultaneously to multiple hubs and or USB functions
- Automatically generates pre-amble for low speed transactions
- USB Host Controller breaks up traffic to match device characteristics
  - Large data blocks are automatically broken up to match device requirements
- User can easily transfer large data files with one simple command
- Error Generation
  - User can generate any error from the USB Host Controller
- Automatic Retry on data Errors

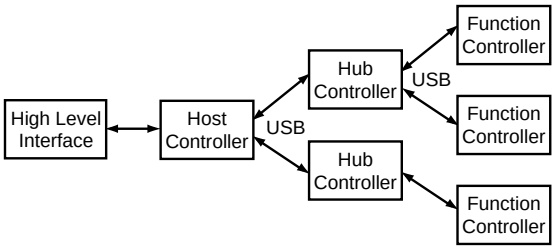


Figure 1: USB Simulation Model

HUB Function Model Features

- Connectivity Behavior
- Device Connect / Disconnect detection
- Bus fault detection and recovery
- Full / low speed device support

Function (Device) Controller Model Features

- Supported Function Controller features
  - Multiple End Points
  - Interrupt
  - Control
  - Isochronous Device
  - Bulk Device
- Multiple Device Emulation
- Error Generation Capabilities
  - CRC Errors
  - Data Retries
  - Response Time out
  - ID error
- Complete Device Configuration
- Supports all device commands
- Full speed / Low Speed Functions

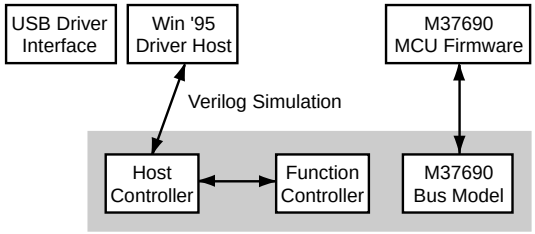


Figure 2: USB Model with Hardware/Software Co-Design

- Multiple Functions can be instantiated
- User Friendly Interface
- Complete Device Emulation Capability
- User Interface option
  - Verilog
  - “C” interface

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

CoreEl MicroSystems

CoreEl MicroSystems

46750 Fremont Blvd. #208
Fremont, CA 94538 USA
Phone: +1 510-770-2277
Fax: +1 510-770-2288
URL: www.coreel.com
E-mail: sales@coreel.com

Features

- Octet-wide operation
- Cell Rate Adaptation
  - Transmission convergence idle cells inserted when input cell not present
  - Fixed idle cell header, HEC, and payload patterns as specified in ATM UNI 3.1
- HEC Computation
  - HEC computed and inserted into the 5th byte of the cell header
  - Coset polynomial always added
- Cell Scrambling
  - Self-synchronizing scrambler as specified in ATM UNI 3.1
  - Does not scramble cell header
- Cell transmitted pulse output
- Continuous clock with cycle-by-cycle enable

Applications

The Cell Assembler core can be used in Asynchronous Transfer Mode (ATM) networking systems such as adapter cards, routers and switches.

AllianceCORE™ Facts

Core Specifics	
Device Family	XC4000XL
CLBs Used	87
IOBs Used	22[1]
System Clock f_{max}	60 MHz
Device Features Used	None

Supported Devices/Resources Remaining		
	I/O	CLBs
XC4005XL-1 PC84	39[1]	109

Provided with Core	
Documentation	Product Brief Specification Document Design Document Test Bench Design Document
Design File Formats	VHDL compiled, EDIF netlist
Constraint Files	UCF File, Exemplar scripts
Verification Tool	Behavioral VHDL Test Bench
Schematic Symbols	None
Evaluation Model	Behavioral VHDL
Reference designs & application notes	ATM-UNI Specification
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	Alliance 1.3
Entry/Verification Tool	Model Tech V-System

Support
Support provided by CoreEl Microsystems

Note:
1. Assuming all core signals are routed off-chip.

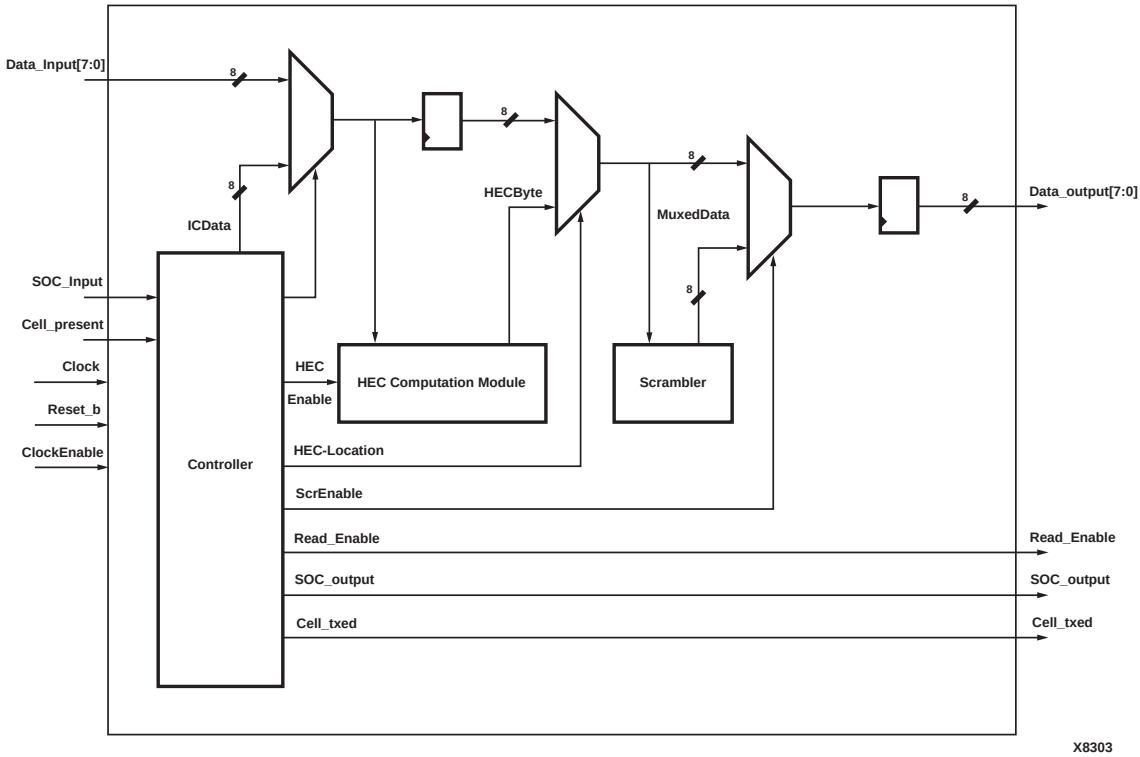


Figure 1: Cell Assembler Core Block Diagram

General Description

The Cell Assembler carries out the functions required to be performed in the receive stream of the Transmission Convergence sub-layer of an ATM Physical Layer processor. The input is a byte-aligned cell stream containing 52-byte cells with 4 bytes of header and 48 bytes of payload. The Cell Assembler computes the header Error Check sequence and inserts it in the fifth byte position of the outgoing stream; thus creating the 53 byte output cell

Functional Description

The Cell Assembler core is divided into blocks as shown in Figure 1: Controller, HEC Computation Module and Scrambler. The Clock, ClockEnable and Reset_b signals are common to all modules.

Controller Block

The Controller inputs SOC_Input and Cell_present signals and controls the data flow. It also inserts idle cells when assigned cells are not present at the input. At the end of every cell, it checks the cell_present signal. If a cell is present, the controller asserts Read_Enable and reads its header. It deasserts Read_Enable for one clock when it insertion of HEC in the stream, the controller reads payload. At the end of the cell it checks the cell present signal again for deciding whether to insert idle cells or to transmit the next valid cell.

HEC Computation Module

The HEC computation module computes HEC on first four header bytes, as indicated by the controller using the polynomial as specified in the ATM UNI 3.1 specification.

Scrambler Module

The scrambler module scrambles only payload octets using the polynomial as specified in the ATM UNI 3.1 specification.

Core Modifications

Normally, modifications are not possible by the user since the core is provided in a Xilinx netlist format. CoreEl can perform special modifications for additional charge. However source code is available for additional cost where the customer can make modifications. Contact CoreEl Microsystems for more information.

Pinout

The pinout is not fixed to any specific device I/O. Signal names for the core are provided in the block diagram shown in Figure 1, and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Data_input[7:0]	Input	Input data bus for transferring in 52-byte cells
SOC_Input	Input	Start of cell indication for input cells
Cell_present	Input	Indicates cell is available to be read in; active high.
Clock	Input	Clock of up to 80 MHz compatible with DS3, E3, STS-3c, and STS-12c applications.
Reset_b	Input	Asynchronous Reset; active low.
Clock_enable	Input	Indicates clock cycles during which module is active.
Data_output [7:0]	Output	Output data containing 53-byte cells with proper HEC.
Read_Enable	Output	Indicates that current clock cycle is being used to read in data; active high.
SOC_output	Output	Start of cell indication; active high.
Cell_txed	Output	Indicates completion of a cell transfer; active high.

Verification Methods

This core has been used in larger ASICs and is silicon proven. The FPGA verification was done by back annotating the implementation and simulating in a Model technology V-System environment.

The test bench was written in VHDL and provides fully automated verification.

Recommended Design Experience

Knowledge of ATM technology and ATM Forum specifications is needed. User should be familiar with HDL design methodology including FPGA targeting. Using the testbench requires familiarity with V-System of Model Technology.

Available Support Products

CoreEI offers a test bench for verifying the core along with a complete line of ATM Core Cells for Xilinx FPGAs that perform the following functions:

- UTOPIA Slave Interface
- Cell Delineation
- CRC-32
- CRC-10

Ordering Information

For information on this or other products mentioned in this specification, contact CoreEI Microsystems directly from the information provided on the front page.

Related Information

The ATM Forum

The ATM Forum publishes specifications regarding ATM. For more information, contact them as follows:

ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1 650-949-6700
Fax: +1 650-949-6705
E-mail: info@atmforum.com
URL: www.atmforum.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

CoreEl MicroSystems

CoreEl MicroSystems

46750 Fremont Blvd. #208
Fremont, CA 94538 USA
Phone: +1 510-770-2277
Fax: +1 510-770-2288
URL: www.coreel.com
E-mail: sales@coreel.com

Features

- Pre-defined implementation for predictable timing in Xilinx FPGA or HardWire[a]
- Octet-wide operation
- State Machine
  - Hunt, Presync, and Sync states
  - Correction and Detection states within the Sync state
- Loss of cell delineation indication
- Idle cell discard
- The header of an input cell, after any error correction, is compared with the header contents specified in ATM UNI 3.1 for an idle cell; such detected idle cells are discarded by disabling the output_enable signal.
- HEC Verification
  - Coset polynomial added before verification
  - Single bit error correction, multiple bit error detection
  - Indication upon an uncorrectable error (cell discard)
- Cell Descrambling
  - Always descrambles cell payload
  - Self synchronizing scrambler-descrambler process specified in ATM UNI 3.1
- Cell received pulse output
- Cell discarded pulse output
- Continuous clock with cycle-by-cycle enable

AllianceCORE™ Facts

Core Specifics		
Device Family	XC4000XL	
CLBs Used	270	
IOBs Used	23[1]	
CLKIOBs Used	2	
System Clock f_{max}	40 MHz	
Device Features Used	Global Clock Buffers	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC4010XL-09 PC84	38[1]	130
Provided with Core		
Documentation	Product Brief Specification Document Design Document Test Bench Design Document Test Script	
Design File Formats	VHDL compiled, EDIF netlist	
Constraint Files	UCF File, Exemplar scripts	
Verification Tool	Script Based behavioral VHDL Test Bench	
Schematic Symbols	None	
Evaluation Model	Behavioral VHDL	
Reference Designs and Application Notes	ATM-UNI Specifications	
Additional Items	ITU-T I.432 Specifications	
Design Tool Requirements		
Xilinx Core Tools	M1.3.7 or later	
Entry/Verification Tool	Model Tech V-System	
Support		
Support provided by CoreEl Microsystems		

Note:
1. Assuming all core signals are routed off-chip.

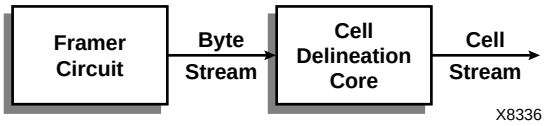


Figure 1: Typical Application of Cell Delineation Core

Applications

The Cell Delineation core can be used in Asynchronous Transfer Mode (ATM) networking systems such as adapter cards, routers and switches.

General Description

The Cell Delineation core (CC-200) carries out the functions required in the receive stream of the Transmission Convergence sub-layer of an ATM Physical Layer processor (see Figure 1). The input is a byte-aligned cell stream containing 53-byte cells with 5 bytes of header and 48 bytes of payload. CC-200 delineates cells as prescribed in ITU specification I.432 and gives out header and payload.

Functional Description

The Cell Delineation core is divided into eight blocks as shown in Figure 2: Data Buffer, HEC Computation Module, Correction Mask Generator, Cell Delineation State Machine, Byte Counters, Idle Cell Detect Logic, Controller and Descrambler.

Block Descriptions

The clock, clockenable and reset_b signals are common to all blocks. The input data is passed through a nine-byte wide buffer and data at five particular buffer positions as indicated in Figure 2, and is fed to the Idle Cell Detect Logic block, to determine whether the cell is assigned or idle.

The HEC computation module checks whether the HEC computed over the first four header bytes matches with the data in the fifth byte and indicates it using HECError signal. It uses the polynomial specified in ATM UNI 3.1.

Cell delineation is carried out by the Cell Delineation State Machine and the Controller block which controls the mechanism.

The Byte Counter keeps track of the number of bytes in a cell (53). It is reset using Reset_cntr signal as well, which is

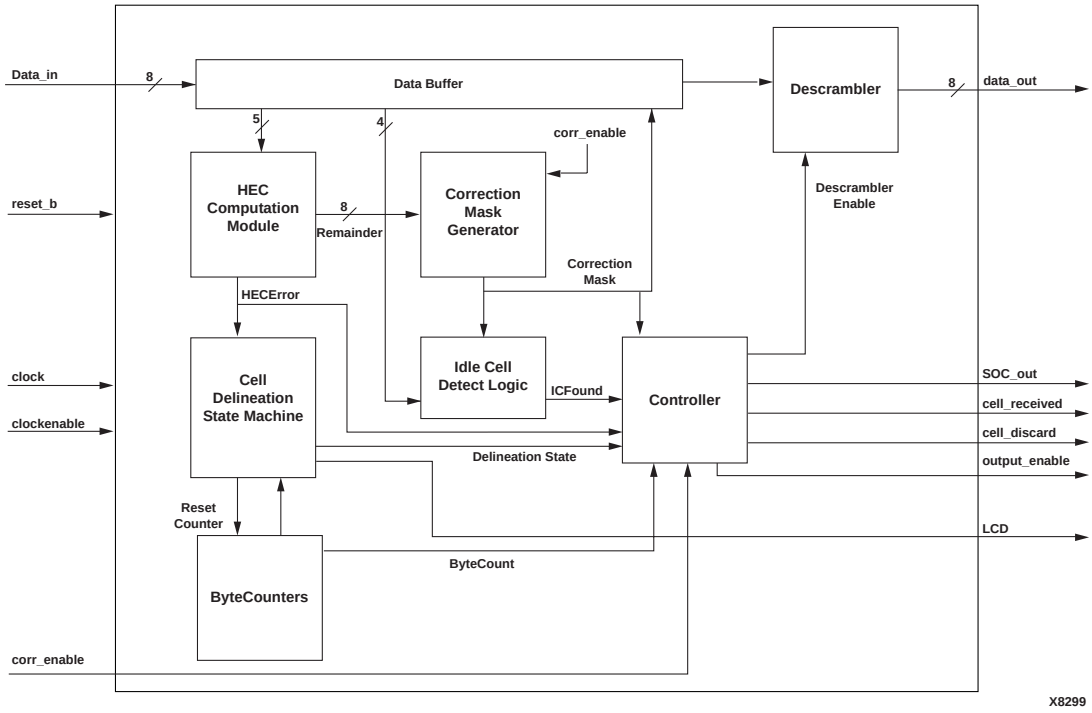


Figure 2: Cell Delineation Core Block Diagram

generated by the Cell Delineation State Machine. The Byte Counter outputs the byte count to the Controller block and the expected location of the header to the Cell Delineation State Machine.

The correction mask generator block takes the remainder from the HEC computation module and finds out the correction mask required in order to correct a single bit error in the header.

The descrambler module descrambles only payload octets using the polynomial specified in ATM UNI 3.1. It descrambles the cell payload using the self-synchronizing scrambler procedure specified in ATM UNI 3.1. When an input cell header matches the pattern specified for idle cells, the cell is discarded.

Basic Operation

In the initial Hunt state, the core performs a byte-by-byte search for possible header locations by applying the CRC check specified by the ATM Forum.

Upon detection of a possible valid header, the state machine enters the Presync state where it verifies the validity of the current cell delineation by checking the CRC Header Error Check (HEC) sequence recorded in the 5th byte of each cell. Upon a successful match for 6 consecutive cells, the state machine enters the Sync state in which cell streams are given out along with a Start-of-cell indication.

The Sync state consists of two sub-states called Correction and Detection. In the Correction state, an attempt is made to correct a cell header with a single bit error. Detection of a multiple bit error results in the cell being discarded. In either case the Detection state is entered where no attempt is made to correct any cell header.

8 consecutive cells with correct header patterns would take the state machine back to the Correction state. Seven consecutive incorrect header patterns would take the state machine back to the Hunt state.

Core Modifications

Normally, modifications are not possible by the user since the core is provided in a Xilinx netlist format. CoreEl can perform special modifications for additional charge. However source code is available for additional cost where the customer can make modifications. Contact CoreEl Microsystems for more information.

Pinout

The pinout is not fixed to any specific device I/O. Signal names for the core are provided in the block diagram shown in Figure 1, and described in Table 1.

Table 1: Signal Core Pinout

Signal	Signal Direction	Description
data_in(7:0)	Input	Byte aligned input carrying a continuous stream of 53-byte cells.
reset_b	Input	Asynchronous reset; active low.
clock	Input	Clock of up to 80 MHz compatible with E3, STS-3c, and STS-12c applications; uses 1 FPGA CLKIOB pin.
clockenable	Input	Indicates clock cycles during which module is active; uses 1 FPGA CLKIOB pin.
corr_enable	Input	Command indicating attempt to correct single bit errors in cell header; active high.
data_out(8:0)	Output	Output data containing 52-byte cells - 4 bytes header and 48 bytes payload.
SOC_out	Output	Start of cell indication; active high.
cell_received	Output	Indicates complete reception of a cell; active high. One clock wide pulse.
cell_discard	Output	Indicates cell has been discarded due to header error; active high. One clock wide pulse.
output_enable	Output	Indicates data on output bus is valid in current clock cycle; active high.
LCD	Output	Indicates loss of cell delineation; active high.

Verification Methods

This core has been used in larger ASICs and is silicon proven. The FPGA verification was done by back annotating the implementation and simulating in a Model technology V-System environment.

The test bench was written in VHDL with very powerful scripting capabilities and several scripts have been written for verifying the implementation. Additional tests can be added to the testbench by writing new scripts.

Recommended Design Experience

Knowledge of ATM technology and ATM Forum specifications is needed. User should be familiar with HDL design methodology including FPGA targeting. Using the testbench requires familiarity with V-System of Model Technology.

Available Support Products

CoreEl offers a test bench for verifying the core along with a complete line of ATM Core Cells for Xilinx FPGAs that perform the following functions:

- UTOPIA Slave Interface
- Cell Delineation
- CRC-32
- CRC-10

Ordering Information

For information on this or other products mentioned in this specification, contact CoreEl Microsystems directly from the information provided on the front page.

Related Information

The ATM Forum

The ATM Forum publishes specifications regarding ATM. For more information, contact them as follows:

ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1 650-949-6700
Fax: +1 650-949-6705
E-mail: info@atmforum.com
URL: www.atmforum.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

CRC10 Generator and Verifier (CC-130)

January 26, 1998 Product Specification

CoreEl MicroSystems

CoreEl MicroSystems

46750 Fremont Blvd. #208
Fremont, CA 94538 USA
Phone: +1 510-770-2277
Fax: +1 510-770-2288
URL: www.coreel.com
E-mail: sales@coreel.com

Features

- Fully compatible with ITU-T Recommendation I.363 for AAL3/4 and ATM OAM Cells
- Single clock operation
- Separate blocks for CRC10 Generator and CRC10 Verifier
- Fully synchronous operation
- Accepts 32 bit data per clock on which CRC10 is computed
- Performance up to 20 MHz giving a throughput of 640 Mbps
- Fully synthesizable Register Transfer Level (RTL) VHDL for FPGA available extra

Applications

The CRC10 Generator and Verifier cores can be used in telecommunications and networking equipment including ATM, SONET and Ethernet systems.

General Description

The CRC10 core is fully compliant to the ITU-T recommendation I.363 for AAL3/4 and ATM OAM Cells. The Core includes both CRC10 Generator and CRC10 Verifier modules. The CRC10 core is fully synchronous with respect to the input clock and is ideally suited to be in a Xilinx FPGA with other high level functions.

Functional Description

The CRC10 Generator and Verifier modules are divided into blocks as shown in Figure 1. Operation of each module is described below.

Note:

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000XL
CLBs - Generator: CLBs - Verifier:	83 46	83 46
IOBs - Generator: IOBs - Verifier:	46[1] 37[1]	46[1] 37[1]
CLKIOBs - Generator: CLKIOBs - Verifier:	1[1] 1[1]	1[1] 1[1]
System Clock f_{max}	20 MHz	20 MHz
Device Features Used	N/A	

Supported Devices/Resources Remaining		
	I/O	**CLBs**
XCS30 PQ208-4 (Gen) XCS30 PQ208-4 (Ver)	83 92	493 530
XC4013XL PQ160-2 (Gen) XC4013XL PQ160-2 (Ver)	83 92	493 530

Provided with Core	
Documentation	Product Brief Specification Document Test Bench Design Document Test Scripts
Design File Formats	VHDL compiled, EDIF netlist
Constraint Files	Generator: crcgen.ucf Verifier: crcver.ucf
Verification Tool	Script Based Behavioral VHDL Test Bench
Schematic Symbols	None
Evaluation Model	Behavioral VHDL
Reference Designs & Application Notes	ITU-T I.363 Specification
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	Alliance 1.3
Entry/Verification Tool	Model Tech V-System

Support
Support provided by CoreEl Microsystems

1. Assuming all core signals are routed off-chip.

Generator Operation

The CRC10 Generator computes and outputs the CRC10 for a Cell Data. Init Signal sets the CRC10 output to all 0's. The 32 Bit Trace Block computes the 32 bit intermediate

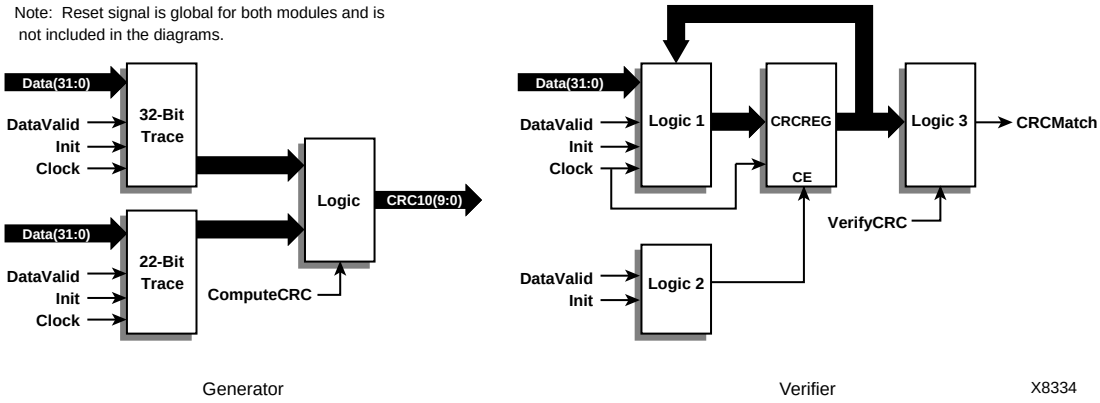


Figure 1: CRC10 Generator and Verifier Block Diagrams

CRC. The 22 Bit Trace Block computes another intermediate 32 bit CRC. The 10 Most Significant Bits of the 32 Bit Trace is selected as the CRC10 remainder when the ComputeCRC is deasserted and the 10 Most Significant Bits of the 22 Bit Trace is selected as the CRC10 remainder when the ComputeCRC is asserted.

Verifier Operation

The CRC10 Verifier computes the CRC10 for the entire cell received and generates a CRCMatch signal if the result is zero. The Init Signal resets the CRCReg Outputs to zero. Once the whole cell is fed, the VerifyCRC signal must be asserted for one clock cycle in order to check the CRCReg Value for zero. If it is zero, the CRCMatch output signal is driven high for one clock cycle.

Core Modifications

Normally, modifications are not possible by the user since the core is provided in a Xilinx netlist format. CoreEI can perform special modifications for additional charge. However source code is available for additional cost where the customer can make modifications. Contact CoreEI Microsystems for more information.

Pinout

The pinout is not fixed to any specific device I/O. Signal names for each module are provided in the block diagrams shown in Figure 1, and described in Table 1.

Verification Methods

This core has been used in larger ASICs and is silicon proven. The FPGA verification was done by back annotating the implementation and simulating in a Model technology V-System environment.

The test bench was written in VHDL with very powerful scripting capabilities and several scripts have been written for verifying the implementation. Additional tests can be added to the testbench by writing new scripts.

Recommended Design Experience

Knowledge of error correction in network systems is needed. User should be familiar with HDL design methodology including FPGA targeting. Using the testbench requires familiarity with V-System of Model Technology.

Available Support Products

CoreEI offers a test bench for verifying the core along with a complete line of ATM Core Cells for Xilinx FPGAs that perform the following functions:

- UTOPIA Interface
- Cell Delineation
- Cell Assembly
- CRC-32

Ordering Information

For information on this or other products mentioned in this specification, contact CoreEI Microsystems directly from the information provided on the front page.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
CRC10 Generator Signals		
Data(31:0)	Input	When *DataValid* is asserted, *Data* should be dword of current cell for which CRC10 is computed.
DataValid	Input	Indicates input *Data* is valid and CRC10 to be computed for value on *Data*.
Init	Input	Init Signal sets initial CRC Value to 0; asserted on start of a new cell.
Clock	Input	Used to sample all other inputs; uses FPGA CLKIOB pin.
Reset	Input	Resets outputs to 0; asserted on power-up/reset.
ComputeCRC	Input	When signal is sampled active, CRC10 output gives valid 10 bit CRC to be transmitted with data; asserted along with last dword given to CRC10 Generator.
CRC10(9:0)	Output	Indicates current CRC remainder of cell being processed.
CRC10 Verifier Signals		
Data(31:0)	Input	When *DataValid* is asserted, *Data* should be dword of current cell for which CRC10 is computed.
DataValid	Input	Indicates input *Data* is valid and CRC10 to be computed for value on *Data*.
Init	Input	Init Signal sets initial CRC Value to 0; asserted on start of a new cell.
Clock	Input	Used to sample all other inputs; uses FPGA CLKIOB pin.
Reset	Input	Resets outputs to 0; asserted on power-up/reset.
VerifyCRC	Input	When signal is sampled active, most significant 10 bits of *CRC* output is compared with zero. *VerifyCRC* should be asserted with 32 bit trailer of cell to find if cell was received without errors. *VerifyCRC* is asserted for one clock cycle if cell received is correct.
CRCMatch	Output	Asserted in response to *VerifyCRC* if current CRC remainder matches zero.

Related Information

The ATM Forum

The ATM Forum publishes specifications regarding ATM. For more information, contact them as follows:

ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1 650-949-6700
Fax: +1 650-949-6705
E-mail: info@atmforum.com
URL: www.atmforum.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

CRC32 Generator and Verifier (CC-131)

January 26, 1998 Product Specification

CoreEl MicroSystems

CoreEl MicroSystems

46750 Fremont Blvd. #208
Fremont, CA 94538 USA
Phone: +1 510-770-2277
Fax: +1 510-770-2288
URL: www.coreel.com
E-mail: sales@coreel.com

Features

- Fully compatible with ITU-T Recommendation I.363 for AAL-5
- Single clock operation
- Separate blocks for CRC32 Generator and CRC32 Verifier
- Fully synchronous operation
- Performance up to 30 MHz giving a throughput of 960 Mbps
- Accepts 32 bit data per clock on which CRC32 is computed
- Fully synthesizable Register Transfer Level (RTL) VHDL source available extra

Applications

The CRC32 Generator and Verifier cores can be used in telecommunications and networking equipment including ATM, SONET and Ethernet systems.

General Description

The CRC32 core is fully compliant to the ITU-T recommendation I.363 for AAL-5. The Core includes CRC32 Generator and CRC32 Verifier modules. The CRC32 core is ideally suited to be designed into a Xilinx FPGA with other high level functions like Segmentation and Reassembly (SAR).

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000XL
CLBs - Generator: CLBs - Verifier:	125 127	125 127
IOBs - Generator: IOBs - Verifier:	100[1] 69[1]	100[1] 69[1]
CLKIOBs - Generator: CLKIOBs - Verifier:	1 1	1 1
System Clock f_{max}	29 MHz	30 MHz
Device Features Used	None	

Supported Devices/Resources Remaining		
	I/O[1]	**CLBs**
XCS30-4 PQ208 (Gen) XCS30-4 PQ208 (Ver)	29 60	451 449
XC4013XL-2 PQ160 (Gen) XC4013XL-2 PQ160 (Ver)	29 60	451 449

Provided with Core	
Documentation	Product Brief Specification Document Test Bench Design Document Test Scripts
Design File Formats	VHDL compiled, EDIF netlist
Constraint Files	crcver.ucf crcgen.ucf
Verification Tool	Script Based Behavioral VHDL Test Bench
Schematic Symbols	None
Evaluation Model	Behavioral VHDL
Reference Designs & Application Notes	ITU-T I.363 Specification
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	Alliance 1.3
Entry/Verification Tool	Model Tech V-System

Support
Support provided by CoreEl Microsystems

Note:
1. Assuming all core signals are routed off-chip.

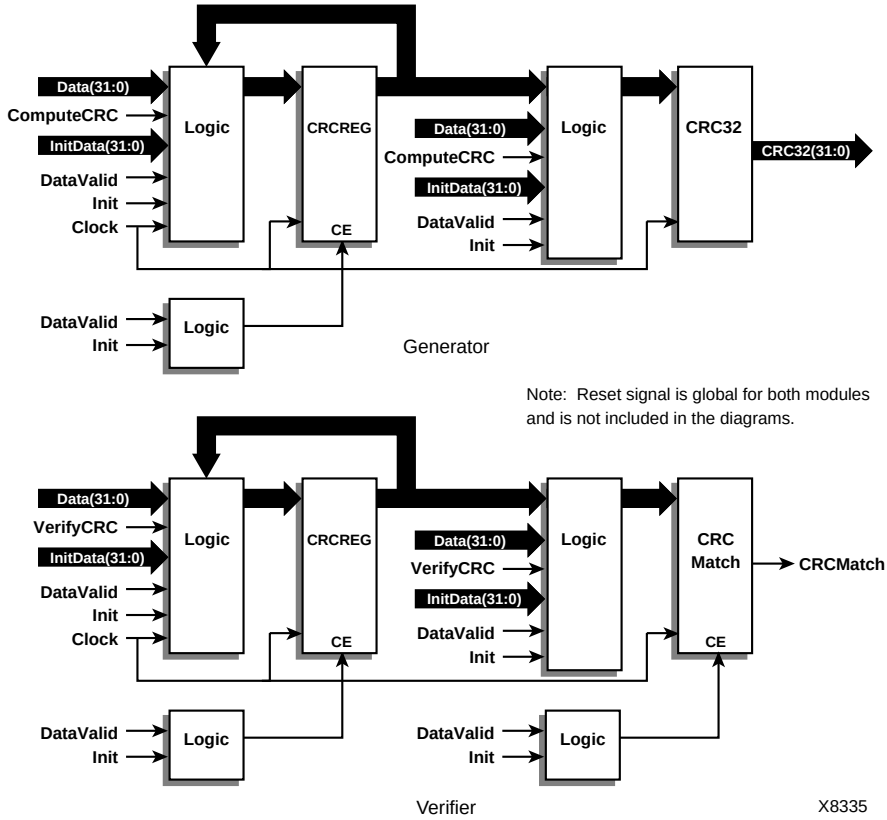


Figure 1: CRC32 Generator and Verifier Block Diagrams

The CRC32 core generates CRC32 remainder over incoming 32-bit data. The current CRC32 remainder value is always driven as an output from the CRC32 core. The CRC32 core supports resetting and preloading of the current CRC32 remainder. It enables the CRC32 core to be used to handle incoming data either as cell data or as packet data. It also supports verification of current CRC32 with a constant remainder as defined in ITU-T recommendation I.363.

Functional Description

The CRC32 Generator and Verifier modules are divided into blocks as shown in Figure 1. Operation of each module is described below.

Generator Operation

The CRC32 Generator computes and outputs the CRC32 for a Data packet. The Reset signal sets the CRC32 outputs 0.

Before starting CRC32 computation over a packet, the CRC register is programmed with the Initial value. According to the ITU I.363 specification the initial value is FFFFFFFF (hex). Assert Init for one clock cycle and drive InitData with FFFFFFFF (hex). From the next clock onwards, the data can be fed along with the qualifier DataValid. DataValid is asserted before the data is fed.

The current value of CRC32 is always available at the CRC32 output bus. Once the whole packet is fed, the ComputeCRC signal must be asserted for one clock cycle in order to get the final CRC32 which conforms to ITU I.363 specification, and which will be transmitted with the packet.

Verifier Operation

The CRC32 Verifier computes the CRC32 for the entire packet received and compares that with a constant value defined in the ITU I.363 specification. It generates a CRCMatch signal if there is a match. Reset signal deasserts the CRCMatch output.

Before starting CRC32 verification over a packet, the CRC register must be programmed with the Initial value FFFFFFFF (hex) as per ITU specification. Assert Init signal for one clock cycle and drive InitData with FFFFFFFF (hex). From the next clock onwards, the data can be fed along with the qualifier DataValid. DataValid is the signal which is to be asserted before the Data is fed.

Once the whole packet is fed, VerifyCRC must be asserted for one clock cycle in order to check for a CRC32 match. If it matches the constant value specified in I.363, the CRC-Match output signal will be driven high for one clock cycle.

Core Modifications

Normally, modifications are not possible by the user since the core is provided in a Xilinx netlist format. CoreEl can perform special modifications for additional charge. However source code is available for additional cost where the customer can make modifications. Contact CoreEl Microsystems for more information.

Pinout

The pinout is not fixed to any specific device I/O. Signal names for each module are provided in the block diagrams shown in Figure 1, and described in Table 1.

Verification Methods

This core has been used in larger ASICs and is silicon proven. The FPGA verification was done by back annotating the implementation and simulating in a Model technology V-System environment.

The test bench was written in VHDL with very powerful scripting capabilities and several scripts have been written for verifying the implementation. Additional tests can be added to the testbench by writing new scripts.

Recommended Design Experience

Knowledge of error correction in network systems is needed. User should be familiar with HDL design methodology including FPGA targeting. Using the testbench requires familiarity with V-System of Model Technology.

Available Support Products

CoreEl offers a test bench for verifying the core along with a complete line of ATM Core Cells for Xilinx FPGAs that perform the following functions:

- UTOPIA Interface
- Cell Delineation
- Cell Assembly
- CRC-10

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
CR32 Generator Signals		
Data(31:0)	Input	When *DataValid* is asserted, *Data* should be next dword of AAL5 packet over which CRC32 is computed.
ComputeCRC	Input	When sampled active, *CRC* output is inverted; should be asserted along with last payload to find CRC remainder to be transmitted as last dword of AAL5 trailer.
InitData(31:0)	Input	When *Init* is asserted, *InitData* should be partial CRC result for AAL5 packet processed so far.
DataValid	Input	Indicates input *Data* is valid and CRC32 to be computed for value on *Data*.
Init	Input	Loads *CRC* output with input *InitData*; asserted at start of new cell.
Clock	Input	Used to sample all other inputs; uses FPGA CLKIOB pin.
Reset	Input	Resets outputs to 0; asserted on power-up/reset.
CRC32(31:0)	Output	Indicates current CRC remainder of AAL5 packet being processed. During AAL5 packet processing, *CRC32* should be sampled and value stored in memory at each packet change. When same AAL5 packet processing resumes, stored CRC value should be driven onto *InitData*, and *Init* pulse should be generated to maintain CRC consistency over AAL5 packet.
CRC32 Verifier Signals		
Data(31:0)	Input	When *DataValid* is asserted, *Data* should be next dword of AAL5 packet over which CRC32 is computed.
VerifyCRC	Input	When sampled active, *CRC* output is compared with an ITU I.363 defined constant; asserted with AAL5 trailer to find if AAL5 packet was received without errors.

Signal	Signal Direction	Description
InitData(31:0)	Input	When *Init* is asserted, *InitData* should be partial CRC result for AAL5 packet processed so far.
DataValid	Input	Indicates input *Data* is valid and CRC32 to be computed for value on *Data*.
Init	Input	Loads *CRC* output with input *InitData*; asserted at start of new cell.
Clock	Input	Used to sample all other inputs; uses FPGA CLKIOB pin.
Reset	Input	Resets outputs to 0; asserted on power-up/reset.
CRCMatch	Output	Asserted in response to *VerifyCRC* if current CRC remainder matches an ITU I.363 defined constant.

Ordering Information

For information on this or other products mentioned in this specification, contact CoreEl Microsystems directly from the information provided on the front page.

Related Information

The ATM Forum

The ATM Forum publishes specifications regarding ATM. For more information, contact them as follows:

ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1 650-949-6700
Fax: +1 650-949-6705
E-mail: info@atmforum.com
URL: www.atmforum.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Integrated Silicon Systems, Ltd.

50 Malone Rd
Belfast BT9 5BS
Northern Ireland
Phone: +44 1232 664664
Fax: +44 1232 669664
E-Mail: info@iss-dsp.com
URL: www.iss-dsp.com

Features

- Single and double byte address recognition
- 16 Bit (CRC-16) and 32 Bit (CRC-32) frame check sequence
- Asynchronous 8 bit input data interface, suitable for a wide range of FIFOs
- Compatible with ITU recommendation Q.921
- Serial interface with external clocking for interfacing to the PCM-highway
- Transmission is synchronous to network interface with back pressure mechanism. Buffering at the network interface not required.
- Supports transparent mode
- Supports modular scaling of multiple HDLC channels through parallel cores or core multiplexing
- Supports data rate up to 50 Mbits/s

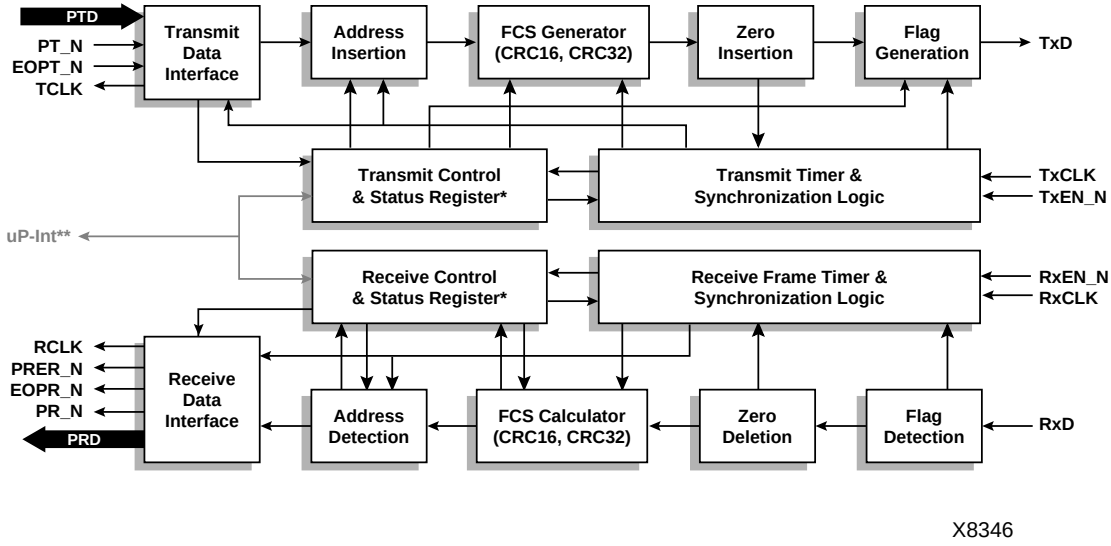
Applications

The ISS HDLC Protocol core is a high performance implementation suitable for a range of applications, including X.25, Frame Relay and ISDN B-channel and D-channel.

AllianceCORE™ Facts

Core Specifics		
Device Family	XC4000XL	
CLBs Used	Area optimized: 129[1] Speed optimized: 151[1]	
IOBs Used	27[2]	
CLKIOBs Used	2	
System Clock fmax	Area optimized: 20 MHz Speed optimized: 42 MHz	
Device Features Used	N/A	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC4013XL-3 (area) XC4013XL-2 (speed)	167[1] 167[1]	447 425
Provided with Core		
Documentation	Core Documentation	
Design File Formats	.xnf or.ngd netlist VHDL source available extra	
Constraint Files	.cst	
Verification Tool	Test Vectors	
Schematic Symbols	VHDL, Verilog	
Evaluation Model	None	
Reference designs & application notes	None	
Additional Items	None	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
Entry/Verification Tool	Synopsys VSS	
Support		
Support provided by ISS.		

Notes:
1. CLB counts are for the example implementation described in Table 1.
2. Assuming all core signals are routed off-chip.



*** Status Register Blocks are typically customized by ISS for each application.**
**** Microprocessor Interface does not come standard with core, but can be customized and added by ISS.**

Figure 1: HDLC Protocol Core Block Diagram

Table 1: Specification for Example Implementation

Parameter	Value
# of channels	1 duplex
Data Rate	Area optimized: 20 Mbps Speed optimized: 42 Mbps
Max frame size	Unlimited
FIFO Size	External
Protocols	HDLC
Additional	8-bit broadcast only address, not inserted16-bit FCS

General Description

The ISS HDLC Protocol Core is a high performance, third-level soft core module for bit-oriented packet transmission mode systems. It is suitable for Frame-Relay, X.25, ISDN B-Channel (64 KBit/s) and D-Channel (16 Kbit/s). The core fulfills the specification according to ITU Q.921, X.25 Level 2 recommendation.

The data stream and transmission rate is controlled from the network node (PCM highway clock) with a back pressure mechanism. This eliminates additional synchronization and buffering of the data at the network interface.

The data interface is 8 bit wide and asynchronous, and includes an 8 bit synchronization buffer. It provides basic adaptation for a wide range of FIFOs. The core can be used as a single channel HDLC protocol controller or switched parallel cores used to implement an N multiple channel controller. The high throughput and modular structure also enables multiplexing of the core between channels for low data rate applications.

Functional Description

The HDLC Protocol core is divided into blocks as shown in Figure 1. Operation of the core is described below.

Transmit Operation

The Transmit Data Interface provides a byte wide interface between the transmission buffer and the HDLC Protocol core. Transmit data is latched on the rising edge of TCLK when the packet transmit (PT) input is asserted. The last

byte of a packet is identified by the end of packet (EOPT) input control signal.

Subsequent blocks in the HDLC Protocol transmit controller insert the address, which may be either single or double byte, into the frame, calculate the Frame Check Sequence (FCS), which may be either a 16 or 32 bit CRC, perform zero insertion on the bit stream to ensure that the flag sequence does not occur in the data stream and add the start and end flags to the frame. The transmit bit stream is clocked out on the rising edges of TxCLK under the control of the enable signal TxEN.

The required configuration of the transmit controller may be defined using the control registers which are accessed via a microprocessor interface. The status register enables the transmit section operation to be monitored

Receive Operation

The HDLC Protocol core receiver accepts a bit stream on port RxD. The data is latched on the rising edges of RxCLK under the control of the enable input RxEN. The Flag Detection block searches the bit stream for the flag sequence in order to determine the frame boundaries. Any stuffed zeros are detected and removed and the FCS is calculated and checked. The core can be set-up, via the receiver control register, to either discard or pass errored frames. Address detection is performed and the relevant frames are made available to the system on the byte wide Received Data Interface.

Microprocessor Interface

The required microprocessor interface can be implemented as part of the provision of any HDLC core. It is not included in the standard core since it is likely to be different for every system in which the core is used.

Core Modifications

The information contained in this datasheet describes the basic HDLC Protocol core provided by ISS. Additional parts of the users system may be developed by ISS and integrated with this core to provide an application specific solution matched directly to the requirements. Typical additional features include microprocessor interface functions, buffers, multichannel capability and channel multiplexers. Contact ISS directly to discuss any specific requirements. See also the HDLC Protocol Core Implementation Request Form at the end of this data sheet.

Pinout

The pinout is not fixed to any specific device I/O. Signal names are provided in the block diagram shown in Figure 1, and described in Table 1. Unless otherwise stated all signals are active high and bit(0) is the least significant bit.

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
Transmitter Interface		
PTD	Input	Packet Transmit Data (7: 0) - for data transfer between transmission buffer and HDLC Protocol core.
PT_N	Input	Packet Transmit - initiates packet transmission sequence by core. Falling edge of signal synchronizes transmission sequence as start of packet; must stay low for duration of packet, if not transmission sequence will be aborted with an abort flag pattern; active low.
EOPT_N	Input	End of Packet Transmit - indicates end of current transmitted packet; must be asserted during last byte period; active low.
TCLK	Output	Transmit Data Clock - byte clock with variable pulse width; loads (on rising edge) 8 bit transmission data into transmission shift register. Rising edge generated when last bit of previous data has been shifted. Shifting is not synchronous and can vary, depending on zero insertion.
TxD	Output	Transmission data - serial data out line, driven by tri-state buffer; may be directly connected to PCM high-way.
TxCLK	Input	Transmission Clock, - 16 kHz to 50 MHz (2.048, 4.096 MHz typical); uses 1 FPGA CLKIOB pin.
TxEN_N	Input	Transmission Enable - enables transmission data and tri-state driver. May be generated from channel multiplexer; active low.

Signal	Signal Direction	Description
Receiver Interface		
RCLK	Output	Receiver Data Clock - byte clock with variable pulse width; pushes, on rising edge, 8 bit received data out of receiver shift register. Rising edge generated when first bit of next byte is received.
PRER_N	Output	Packet Receive Error - indicates frame error. It is CRC results of received packet; active low.
EOPR_N	Output	End of Packet Received - indicates end of received packet, set during last received byte; active low.
PR_N	Output	Packet Received - indicates receipt of valid correct packet; stays low, until received packet is transferred; active low.
PRD	Output	Packet Receive Data (7:0) - provides data transfer between receiver of HDLC Protocol Core and receive buffer.
RxEN_N	Input	Receive Enable - enables masking of received data, may be generated by channel multiplexer; active low.
RxCLK	Input	Receiver Clock, - 16 kHz up to 50 MHz (2.048, 4.096 MHz typical)
RxD	Input	Receive Data - is serial data in, latched with rising edge of RxCLK if RxEN is low.

Verification Methods

The core has been verified in Xilinx devices using post layout simulation across a wide range of configurations by using input vector sequences compatible with many international standards.

Available Support Products

Also available from ISS is a bit parallel (byte stuffed) version of this core for PPP applications.

Ordering Information

For information on this or other products mentioned in this specification, contact Integrated Silicon Systems directly from the information provided on the front page.

Related Information

International Telecommunications Union

International Telecommunications Union
Place des Nations
CH-1211Geneve 20
Switzerland
Phone: +41 22 730 51 11
Fax: +41 22 733 72 56

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95214
Phone: 408-559-7778
Fax: 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: 800-231-3386 (inside the US)
408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE specific information, contact:

Phone: 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

HDLC Protocol Core Implementation Request Form

To: Integrated Silicon Systems, Inc.
FAX: +44 1232 669664
E-mail: info@iss-dsp.com

From: _____

Company: _____

Name: _____

Address: _____

Country: _____

Phone: _____

Fax: _____

E-mail: _____

ISS configures and ships Xilinx netlist versions of the HDLC Protocol core customized to your specification. Please fill out and fax this form so ISS can respond with an appropriate quotation that includes performance and density metrics for the target Xilinx FPGA

1. Number of required channels: _____
   If more than one channel, storage is required for inactive channel state. Is this storage to be on-chip or off-chip?

2. Please indicate:
   ____ Serial data stream output with Zero-bit stuffed transparency
   ____ Byte parallel data output with Octet stuffed transparency?

3. Data rate (per channel): _____

4. Available clocks: _____

5. The bare core provides separate input, output and enable ports for each configurable register. If a specific host bus interface is required, please specify the interface below.

6. The bare core provides byte-wide data ports with ByteTaken and ByteAvailable signals for use with external data FIFOs. If on-chip FIFOs are required with the core, please indicate which and specify their required size. Appropriate control and error condition signals will then be available.

   - Tx FIFO Size: _____

   - Rx FIFO Size: _____

7. If there are any specific protocol requirements or deviations from the HDLC specification, please detail them below.

8. Indicate required HDLC frame address field size:
   ____ 8 bits fixed
   ____ 16 bits fixed
   ____ Programmable by control register bit

9. Indicate how address should be inserted into transmitted HDLC frames:
   ____ Always
   ____ Never
   ____ Programmable by control register bit

10. Indicate if a programmable address is to be matched against incoming frames:
    ____ Always
    ____ Never
    ____ Programmable by control register bit

11. Indicate if the broadcast (all ones) address is recognized:
    ____ Always
    ____ Never
    ____ Programmable by control register bit

12. Indicate if the broadcast address is recognized
    ____ Yes
    ____ No

13. Indicate if an all addresses mode (where all frames are accepted and passed to the host
    ____ Always (Note: this effectively disables all other forms of address recognition.)
    ____ Never
    ____ Programmable by control register bit

14. Indicate if incoming address is stripped out of the packet before the data is passed through to the host:
    ____ Always
    ____ Never
    ____ Programmable by control register bit

15. Indicate HDLC Frame Check Sequence (CRC) size:
    ____ 16 bits fixed
    ____ 32 bits fixed
    ____ Programmable by control register bit

16. The default FCS polynomials are as follows. If alternative polynomials are required, specify them in the space below.
    16 bit: $X^{16}+X1^{2}+X^{5}+1$
    32 bit: $X^{32}+X^{26}+X2^{3}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X+1$

17. Specify any additional requirements:

Integrated Silicon Systems, Ltd.

50 Malone Rd
Belfast BT9 5BS
Northern Ireland
Phone: +44 1232 664664
Fax: +44 1232 669664
E-Mail: info@iss-dsp.com
URL: www.iss-dsp.com

Features

- Programmable solution for high data rate Reed Solomon decoding
- Can be configured to support a range of standards involving Reed Solomon encoding e.g. European Telecommunication Standard 300 421 and 300 429
- Paramaterized architecture can be customized using the following parameters:
  - Symbol size
  - Number of errors corrected
  - Number of erasures corrected
  - Field polynomial
  - Generator polynomial
  - Statistics gathering
  - Interfaces
- Single decoder implementation supports any valid data block length
- Systematic code structure where each code word can be partitioned into original data block and appended parity symbols
- Decoder can process continuous or burst data
- Supports high speed (>300 Mbps) applications
- Symbol-wide inputs and outputs, clocked by a single symbol-rate clock
  - Design can be further optimized if higher rate clock (e.g. bit rate clock) is available
- Simple core interface allows easy integration into larger systems

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000XL
CLBs Used	See Table 1	
IOBs Used		
System Clock fmax		
Device Features Used	RAM, 3-state buffers, Carry Logic, RPM	
Supported Devices/Resources Remaining		
See Table 1		
Provided with Core		
Documentation	Core Documentation	
Design File Formats	.ngd or XNF netlist VHDL Source RTL available extra	
Constraint Files	Timespec, .cst and.tnm files	
Verification Tool	Synopsys VSS	
Schematic Symbols	None	
Evaluation Model	None	
Reference designs & application notes	None	
Additional Items	Bit-accurate C Model is available	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
Entry/Verification Tool	Synopsys Design Compiler Synopsys VSS	
Support		
Support provided by ISS.		

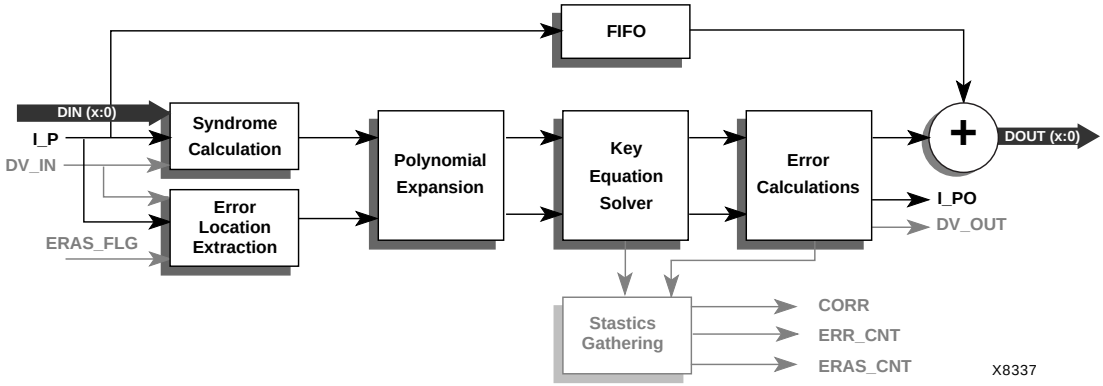


Figure 1: Reed-Solomon Decoder Block Diagram - clock and reset signals omitted for clarity

Table 1: Example Encoder Implementations

	Example #1	Example #2	Example #3
Correctable Errors	8	4	4
Bits/Symbol	8	8	8
Erasure Support?	No	No	No
Statistics Gathering?	No	Yes	Yes
Xilinx Device	XC4036XL-1	XCS40-3	XC4020E-2
CLBs Used	941	740	740
IOBs Used	19	24	24
CLKIOBs Used	1	1	1
System Clock fmax	40 MHz	20 MHz	20 MHz
Data Rate	320 Mbps	160 Mbps	160 Mbps
CLBs Remaining	355	44	44
I/O Remaining	268	200	200

Applications

- Digital video and audio broadcast.
- Digital satellite broadcast.
- Data storage and retrieval systems (e.g. hard disks, CD-ROM etc.).

General Description

The Reed Solomon decoder is a Xilinx FPGA-based core for systems where data error detection/correction is required. The core implements the full functionality of a general Reed Solomon decoder and all necessary interface circuitry using a single Xilinx FPGA.

The source code version of the core is extremely flexible due to its parameterized design, and can be rapidly configured for a wide variety of applications. The AllianceCORE facts table shows density and performance metrics for a specific implementation example. ISS will customize and provide a Xilinx-specific implementation of the core tailored to the needs of your application, (see Ordering Information).

Polynomials

A typical Code Generator Polynomial for the decoder is:

$g(x) = (x + \alpha^0)(x + (^1)(x + \alpha^2) \ldots..(x + \alpha^{15})$, where $(=\alpha 02_{HEX}$ and the number of parity symbols is 16.

A typical Field Generator Polynomial is:

$p(x) = x^8 + x^4 + x^3 + x^2 + 1$, where the Galois Field used is $GF(2^8)$.

Functional Description

The Reed-Solomon decoder core is partitioned into modules as shown in Figure 1, and described below.

Syndrome Calculation

The syndrome calculator accepts the received symbols. It considers the symbol values as polynomial coefficients and determines if the series of symbols contained in a data block form a valid code word for the particular Reed-Solomon code chosen. It evaluates the polynomial for 2t syndrome values (where t is the number of correctable random errors) and detects whether the evaluations are zero (the data block is a code word) or non-zero (the data block is not a code word). Any block that is not a code word has been affected by errors.

Erasure Location Extraction

Erasures are errors in known locations in the received data block, and are flagged by other system functions such as the receiver demodulator. Since the location of an erasure is known, the Reed-Solomon decoder only needs to determine the magnitude of the error. This is unlike a random error where both error location and magnitude are calculated. Therefore, for a given correction power, twice as many erasures as random errors can be corrected.

A combination of erasures and random errors can be corrected provided the combination does not exceed the correction power of the code, which is directly related to the number of parity symbols added to a data block.

Erasure symbols are flagged using the ERAS_FLG input, and the erasure location extraction block produces a polynomial representing erasure locations.

Polynomial Expansion

The syndrome and erasure polynomials are combined in this block to produce a single unified polynomial that can be processed by the key equation solver.

Key Equation Solver

The key equation relates the combined syndrome/erasure polynomial with two error polynomials. The first is the error locator polynomial, which gives information about which symbols in the code word are in error. The second is the error evaluator polynomial which, when further processed by the error calculation circuit, yields information regarding the magnitude of each error. The key equation solver is responsible for the derivation of the locator and evaluator polynomials.

Error Calculation

This component uses the error locator and evaluator polynomials to compute the error values for each symbol in the received data block. As each error value is computed, the appropriate received symbol is fetched from a code word FIFO, which buffers the received symbols during the decoding process. Each error value is simply added to the received symbol to produce the corrected symbol. (Note that addition and subtraction are identical in the Finite Field arithmetic used in Reed-Solomon decoding.)

Statistics Gathering (Optional)

This block receives information from the key equation solver and the error calculator. It then computes statistics regarding the degree of data corruption caused by the transmission channel and the success of the decoder in correcting this corruption. Typical statistical functions include:

- Random error and erasure counts
- Flags to indicate uncorrectable data blocks
- Computation of bit error rates

This block along with the signals DV_IN, DV_OUT, CORR, ERR_CNT, and ERAS_CNT are optional. ISS can customize and provide this for specific applications depending upon your need.

Clock

All flipflops in the R-S decoder core operate on the rising edge of the input symbol rate clock CLK. Data inputs are latched and outputs generated on the rising edge of the clock. CLB count of the core can be reduced if a higher clock rate, such as the bit rate clock, is available. In this case some of the hardware functions within the decoder may be clocked at the higher rate and reused multiple times per symbol period, thus reducing the overall hardware requirements.

Latency

The latency of the Reed-Solomon decoder is heavily dependent on:

- Data block length
- Number of correctable errors
- Whether uncorrectable data blocks are to be flagged
- Degree by which the clock rate exceeds the symbol rate

The syndrome calculation and determination of block correctability usually takes one data block length each since all symbols in a block must be examined. Therefore, an uncorrectable signal is valid once the entire block has passed through the statistics gathering circuit.

To ensure that the uncorrectable signal emerges from the decoder simultaneous to the first data block symbol, the data must be delayed by one block length. Hence, the Syndrome Calculator and Statistics Gathering blocks account for two data block lengths of latency.

The remaining components contribute variable latency based on the clock rate and the number of correctable errors. It should be noted that these are only general guidelines for decoder latency and, if low latency is a priority, it can be achieved by trading off other design characteristics e.g. area. The latency of the Reed Solomon decoder example designs are shown in Table 1.

Pinout

The pinout of the Reed-Solomon decoder has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are given in Table 2.

Input Interface

The input interface to the decoder consists of a bit parallel symbol data input, DIN, information/parity flag, I_P, input data valid flag, DV_IN, erasure indicator flag, ERAS_FLG, the symbol rate clock, CLK and an asynchronous reset, RES. All inputs are sampled on the CLK rising edge and the control inputs refer to the current input symbol.

I_P together with DV_IN identify valid input information and parity symbols. At the start of a block, both I_P and DV_IN are asserted to indicate valid information symbols at the data input. After the information symbols have been received the I_P input is deasserted, while DV_IN is held asserted, to indicate valid parity symbols on the input.

During periods when no valid input symbols are being input, DV_IN is deasserted. The erasure indicator input flag, ERAS_FLG, is asserted to indicate erasures in the input data stream that have been identified by, for example, a Viterbi decoder function in the system.

Output Interface

The output interface of the decoder consists of a bit parallel symbol data output, DOUT, information/parity flag, I_PO, output data valid, DV_OUT, correctable indicator, CORR, errors corrected count, ERR_CNT and erasures corrected count, ERAS_CNT. All outputs are clocked out on the rising edge of the symbol clock, CLK.

I_PO and DV_OUT are asserted when the first symbol of a block is output to indicate valid information symbols. I_PO is deasserted when parity symbols are output, while DV_OUT remains asserted. DV_OUT is deasserted at the end of the block if the start of the next block is not available, otherwise it is remains asserted.

If a block can be corrected, then CORR is also asserted at the first symbol of the block and remains in this state for the duration of the block output, and is deasserted with DV_OUT. If CORR is not asserted then the decoder was unable to correct the block.

ERR_CNT indicates the number of errors that were corrected for a block, while ERAS_CNT indicates the number of erasures that were corrected for a block. These signals go active at the first output symbol and are maintained for the duration of the block output from the decoder.

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
Input Interface Signals		
DIN[x:0]	Input	Input data symbols
I_P	Input	Information / parity input indicator, which identifies the current input symbol as information symbol or parity symbol. 1 = information symbol 0 = parity symbol
ERAS_FLG*	Input	Optional input flag that marks the position of erasures in the input data stream.
CLK	Input	Symbol clock; uses 1 FPGA CLKIOB pin
RES	Input	Asynchronous reset, active high
Output Interface Signals		
DOUT[x:0]	Output	Data output symbols
I_PO	Output	Output data information /parity flag 1= information 0 = parity
Statistics Gathering Signals (Optional)		
DV_IN	Input	Data valid input. When 1 indicates valid input data.
DV_OUT	Output	Output data valid flag, active high
CORR	Output	Correctable output flag, which is output concurrent with the first symbol of a block and indicates that the block was correctable. 1 = correctable 0 = non-correctable
ERR_CNT	Output	Number of errors corrected in block
ERAS_CNT	Output	Number of erasures corrected in blocks

* Only relevant if erasure support is required.

Verification Methods

The core has been fully tested across a wide range of parameter settings, using input sequences compatible with many international digital video, audio and satellite standards.

Recommended Design Experience

Familiarity with system standards relevant to the particular application is assumed. A basic understanding of Reed Solomon decoding is useful, but not essential.

Available Support Products

A bit accurate C model of the Reed-Solomon decoder is available to assist in functional verification and system integration.

ISS also supplies a complete line of peripheral cores that can be integrated with the Reed Solomon Encoder and Decoder cores for a complete forward error correction system using Xilinx FPGAs. For more information, contact ISS directly regarding:

- Block and convolution interleavers and deinterleavers
- Scramblers and descramblers
- Sync detection and insertion
- Convolution encoders
- Viterbi decoders

Ordering Information

The Reed Solomon Decoder is available for purchase directly from ISS. The implementation will vary depending upon your application. To determine what is required for your system, fill out and fax the attached Implementation Request Form to ISS at +44 1232 669664.

Related Information

European Telecommunications Standards Institute

For information on European digital broadcasting systems standards contact:

European Telecommunications Standards Institute
6921 Sophia Antipolis Cedex
France
Phone: +33 92 94 42 00
Fax: +33 93 65 47 16

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95214
Phone: 408-559-7778
Fax: 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: 800-231-3386 (inside the US)
408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE specific information, contact:

Phone: 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Reed-Solomon Implementation Request Form

To: Integrated Silicon Systems, Inc.
FAX: +44 1232 669664
E-mail: info@iss-dsp.com

From: _____

Company: _____

Name: _____

Address: _____

Country: _____

Phone: _____

Fax: _____

E-mail: _____

ISS configures and ships Xilinx netlist versions of the Reed Solomon cores customized to your specification. Please fill out and fax this form so ISS can respond with an appropriate quotation that includes performance and density metrics for the target Xilinx FPGA.

Implementation Issues

1. Indicate your exact requirement:
   _____ Reed-Solomon encoder
   _____ Reed-Solomon decoder
   _____ Interleaver
   _____ Deinterleaver

2. Maximum symbol rate: _____

3. Number of bits per symbol: _____

4. Indicate clock availability:
   _____ Bit clock
   _____ Symbol Clock
   _____ Both

5. Code generator polynomial (e.g. $(x+a^1)(x+a^2)...$)

6. Field generator polynomial (e.g. $x^8+x^4+x^3+x^2+1$)

7. Number of errors: _____

8. Codeword length/format (e.g. 204, 188):

9. Erasure support required:
   Yes _____ No _____

10. Indicate error statistics reporting requirement(s):
    _____ Signal to show position of corrected errors
    _____ Signal to show position of uncorrected blocks
    _____ Other (please specify)

11. Indicate required control signals:
    _____ low to high transition at start of codeword, high to low transition at end of codeword
    _____ low to high transitional start of codeword, high to low transition at start of parity symbols
    _____ delayed control signals available at output
    _____ other (please specify)

12. Please specify any additional requirements:

13. If interleaver and/or deinterleaver functions are required, specify maximum depth of interleaving:

14. If the interleaver and/or deinterleaver are required is the depth of interleaving to be programmable and if so, to what degree?

Business Issues

15. Indicate timescales of requirement:
    a. date for decision _____
    b. date for placing order _____
    c. required delivery date _____

16. Indicate your area of responsibility:
    _____ decision maker
    _____ budget holder
    _____ recommender

17. Has a budget been allocated for the purchase?
    Yes _____ No _____

18. What volume do you expect to ship of the product that will use this core? _____

19. What major factors will influence your decision?
    _____ cost
    _____ customization
    _____ testing
    _____ implementation size

20. Are you considering any other solutions?

Integrated Silicon Systems, Ltd.

50 Malone Rd
Belfast BT9 5BS
Northern Ireland
Phone: +44 1232 664664
Fax: +44 1232 669664
E-Mail: info@iss-dsp.com
URL: www.iss-dsp.com

Features

- Programmable solution for high data rate Reed-Solomon encoding
- ISS can configure to support a range of standards involving Reed-Solomon encoding, e.g. European Telecommunication Standard 300 421 and 300 429.
- Architecture can be customized using the following parameters:
  - Symbol size
  - Number of parity symbols
  - Field polynomial
  - Generator polynomial
- Single encoder implementation supports any valid data block length
- Systematic code structure where each code word can be partitioned into original data block and appended parity symbols
- Encoder can process continuous or burst data
- Supports high speed (>570 Mbps) applications
- Symbol-wide inputs and outputs, clocked by a single symbol-rate clock
- Simple interface allows easy integration into larger system

Applications

- Digital video and audio broadcast
- Digital satellite broadcast
- Data storage and retrieval systems (e.g. hard disks, CD-ROM etc)

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000E/XL
CLBs Used	See Table 1	
IOBs Used		
System Clock fmax		
Device Features Used	Carry Logic, RPM	
Supported Devices/Resources Remaining		
See Table 1		
Provided with Core		
Documentation	Core Documentation	
Design File Formats	XNF or NGD netlist	
Constraint Files	Timespec, .cst and .tnm files	
Verification Tool	Test Vectors	
Schematic Symbols	None	
Evaluation Model	None	
Reference designs & application notes	None	
Additional Items	Bit-accurate C Model is available	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
Entry/Verification Tool	Synopsys Design Compiler Synopsys VSS	
Support		
Support provided by ISS.		

General Description

The Reed-Solomon encoder is a Xilinx FPGA-based core for systems where data error detection/correction is required. The core implements the full functionality of a general Reed-Solomon encoder and all necessary interface circuitry using a single Xilinx FPGA.

The source code version of the core is extremely flexible due to its parameterized design, and can be rapidly configured for a wide variety of applications. Table 1 shows density and performance metrics for specific implementation examples. ISS will customize and provide a Xilinx-specific implementation of the core tailored to the needs of your application, (see Ordering Information).

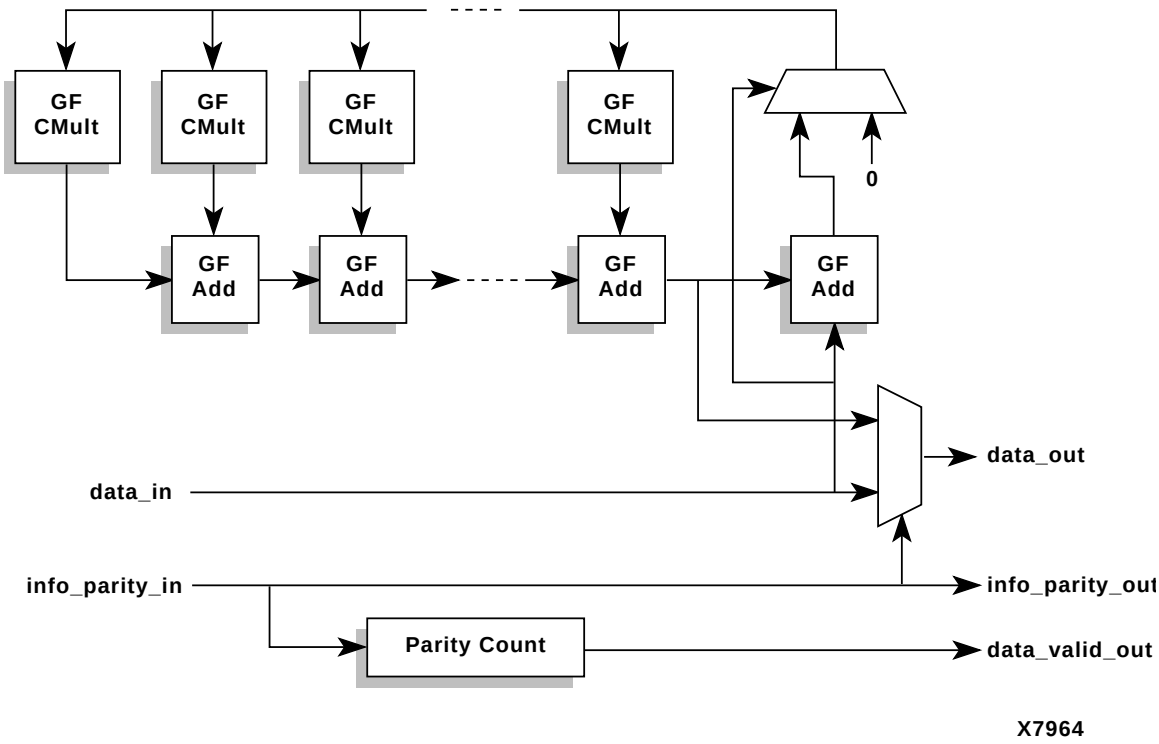


Figure 1: Reed Solomon Encoder Block Diagram

Table 1: Example Encoder Implementations

	Example #1	Example #2	Example #3	Example #4
Parity Symbols	16	16	16	8
Bits/Symbol	8	8	8	4
Device	XC4005XL-1 PC84	XC4005E-3 PC84	XCS10-3 PC84	XC4003E-3 PC84
CLBs Used	105	98	106	24
IOBs Used	20	21	19	13
System Clock f_{max}	72.2 MHz	42 MHz	49.3 MHz	48 MHz
CLBs Remaining	91	98	90	172
IOBs Remaining	41	40	42	48

Functional Description

The Reed-Solomon encoder core is partitioned into modules as shown in Figure 1 and described below.

GF CMult

This block performs the multiplication of two symbol values over a Galois Field that is defined by the choice of the number of bits per symbol. For example, choosing 8 bits per symbol, the block performs multiplication over GF(2^8). One of the multiplier inputs is a constant, corresponding to a coefficient value from the generator polynomial, which is also customizable by the user. Multiplication is carried out using a bit-parallel, polynomial basis architecture. One GF CMult block is required for each parity symbol to be generated.

GF Add

The GF Add block performs the addition of two symbol values over the appropriate Galois Field.

Parity Count

The parity count block counts the number of parity symbols

produced by the encoder for each block of data. The number of valid parity symbols is twice the maximum number of errors that are to be corrected. Any additional symbols produced by the encoder are caused by spurious symbol values that may exist between successive data blocks when burst data is being processed. These spurious symbols are marked in the output data stream by changing the value of the “data_valid_out” signal from high to low.

Pinout

The pinout of the Reed-Solomon Encoder has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are shown in the block diagram of Figure 1 and in Table 2.

Table 2: Core Signal Pinout

Signal	Signal Direction	Description
DATA_IN [x:0]	Input	Input data symbols
INFO_PARITY_IN	Input	Marks valid input information symbols
CLK	Input	Symbol Clock
RESET	Input	System Reset, active high
DATA_OUT [x:0]	Output	Output data symbols
INFO_PARITY_OUT	Output	Marks valid output information symbols
DATA_VALID_OUT	Output	Marks valid output information and parity symbols.

Verification Methods

The core has been fully tested across a wide range of parameter settings, using input sequences compatible with many international digital video, audio and satellite standards.

Recommended Design Experience

Familiarity with system standards relevant to the particular application is assumed. A basic understanding of Reed-Solomon encoding is useful, but not essential.

Available Support Products

A C model of the Reed-Solomon encoder is available to assist in functional verification and system integration.

ISS also supplies a complete line of peripheral cores that can be integrated with the Reed Solomon Encoder and Decoder cores for a complete forward error correction system using Xilinx FPGAs. For more information, contact ISS directly regarding:

- Block and convolution interleavers and deinterleavers
- Scramblers and descramblers
- Sync detection and insertion
- Convolution encoders
- Viterbi decoders

Ordering Information

The Reed Solomon Encoder is available for purchase directly from ISS. The implementation will vary depending upon your application. To determine what is required for your system, fill out and fax the attached Implementation Request Form to ISS at +44 1232 669664.

Related Information

European Telecommunications Standards Institute

For information on European digital broadcasting systems standards contact:

European Telecommunications Standards Institute
06921 Sophia Antipolis Cedex
France
Phone : +33 92 94 42 00
Fax : +33 93 65 47 16

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Reed-Solomon Implementation Request Form

To: Integrated Silicon Systems, Inc.
FAX: +44 1232 669664
E-mail: info@iss-dsp.com

From: _____

Company: _____

Name: _____

Address: _____

Country: _____

Phone: _____

Fax: _____

E-mail: _____

ISS configures and ships Xilinx netlist versions of the Reed Solomon cores customized to your specification. Please fill out and fax this form so ISS can respond with an appropriate quotation that includes performance and density metrics for the target Xilinx FPGA.

Implementation Issues

1. Indicate your exact requirement:
   _____ Reed-Solomon encoder
   _____ Reed-Solomon decoder
   _____ Interleaver
   _____ Deinterleaver

2. Maximum symbol rate: _____

3. Number of bits per symbol: _____

4. Indicate clock availability:
   __ ____ Bit clock
   _____ Symbol Clock
   _____ Both

5. Code generator polynomial (e.g. $(x+a^1)(x+a^2)...$)

6. Field generator polynomial (e.g. $x^8+x^4+x^3+x^2+1$)

7. Number of errors: _____

8. Codeword length/format (e.g. 204, 188):

9. Erasure support required:
   Yes _____ No _____

10. Indicate error statistics reporting requirement(s):
    _____ Signal to show position of corrected errors
    _____ Signal to show position of uncorrected blocks
    _____ Other (please specify)

11. Indicate required control signals:
    _____ low to high transition at start of codeword,high to low transition at end of codeword
    _____ low to high transitional start of codeword, high to low transition at start of parity symbols
    _____ delayed control signals available at output
    _____ other (please specify)

12. Please specify any additional requirements:

13. If interleaver and/or deinterleaver functions are required, specify maximum depth of interleaving:

14. If the interleaver and/or deinterleaver are required is the depth of interleaving to be programmable and if so, to what degree?

Business Issues

15. Indicate timescales of requirement:
    a. date for decision _____
    b. date for placing order _____
    c. required delivery date _____

16. Indicate your area of responsibility:
    _____ decision maker
    _____ budget holder
    _____ recommender

17. Has a budget been allocated for the purchase?
    Yes _____ No _____

18. What volume do you expect to ship of the product that will use this core? _____

19. What major factors will influence your decision?
    _____ cost
    _____ customization
    _____ testing
    _____ implementation size

20. Are you considering any other solutions?

MT1F T1 Framer

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408-733-3344
Fax: +1 408-733-9922
E-mail: sales@virtualipgroup.com
URL: www.virtualipgroup.com

Features

- DS1/ISDN - Primary Rate Interface Framing transceiver
- Frames to D4, ESF, and SLC-96 formats
- Control through Microprocessor interface
- Extracts and inserts robbed - bit signaling
- Programmable output clocks
- FDL support logic circuitry
- Fully independent Transmitter and Receiver block
- Pay load and Local loopback capability
- Loop up/down code generation and detection capability
- Extensive status indication

Applications

- DS1/ESF digital trunk interfaces
- Computer to PBX interfaces
- High speed computer to computer data links
- Digital cross-connect interface

AllianceCORE™ Facts

Core Specifics		
Device Family	XC4000EX	
CLBs Used	1296	
IOBs Used	37[1]	
System Clock fmax	1.544 MHz	
Device Features Used	N/A	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC4036EX-3 HQ240	156[1]	0
Provided with Core		
Documentation	Core Specification Document	
Design File Format	.ncd and .nga files Verilog Source RTL avail extra	
Constraint Files	.ucf	
Verification Tool	Test Vectors	
Schematic Symbols	None	
Evaluation Model	None	
Reference Designs and Application Notes	None	
Additional Items	None	
Provided with Core		
Xilinx Core Tools	Alliance 1.3	
Verification Tool	Verilog XL Simulator	
Support		
Support provided by Virtual IP Group.		

Note:

1. Assuming all core signals are routed off-chip.

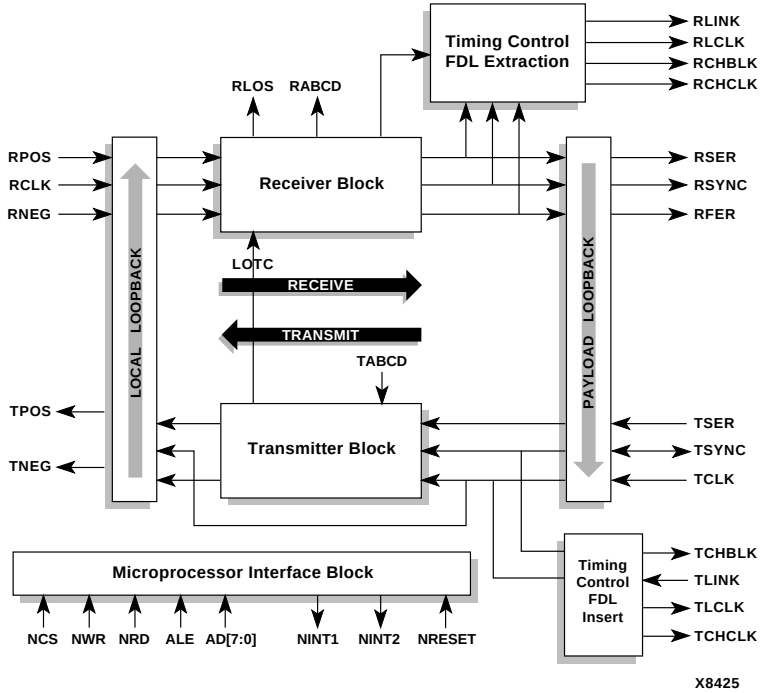


Figure 1: MT1F T1 Framer Block Diagram

General Description

The MT1F is a comprehensive, software-driven T1 framer core. It can be interfaced to a standard microcontroller or microprocessor data bus. The MT1F is very flexible and can be configured into numerous orientations via software. The core provides a set of 60 8-bit internal registers which the user can access to configure the core and obtain information from the T1 link. The core fully meets all of the latest T1 specifications including ANSI T1.403-1989, AT&T TR 62411 (12-90), and CCITT G.704 and G.706.

The MT1F core is compatible with the existing D4 framing standard described in AT&T PUB 43801 and the new extended superframe format (ESF) as described in AT&T C.B# 142. The salient differences between the D4 framing and ESF framing formats are the number of frames per superframe and use of the F-bit position (refer to tables 4&5). In the D4 framing 12 frames make up a superframe; in ESF, 24. A frame consists of 24 channels (times slots) of 8-bit data preceded by an F-bit. Channel data is transmitted and received MSB first.

Functional Description

Functionally the MT1F T1 Framer core can be divided into three main sections: the Receiver block, the Transmitter block, and the Microprocessor interface block. The block diagram is shown in Figure1.

Receiver Block

On the receive side, the device will clock in the serial T1 stream via the RPOS and RNEG signal lines. The synchronizer will locate the frame and multiframe patterns and establish their respective positions. This information is used by the rest of the receive side circuitry.

The MT1F is an “off-line” framer, which means that all of the T1 serial stream that goes into the device will come out unchanged. Once the T1 data has been framed, the robbed-bit signaling data and FDL can be extracted. The receiver block provides the logic for B8ZS decoder, Bipolar violation detection and counting, synchronizer, alarm detection, loop code detection, CRC generation and detection, error counting for CRC and Frame error, signaling extraction and channel marking.

Transmitter Block

The transmit side clocks in the unframed T1 stream at TSER and adds in the framing pattern, the robbed-bit signaling, and the FDL. The Transmit block includes logic required for functions like F bit insertion, idle code insertion, clear channel, signaling insertion, loop code generation, Bit 7 stuffing, CRC generation, FDL insertion, Yellow alarm generation and insertion, AIS generation and B8ZS encoder.

Microprocessor Interface Block

The Microprocessor interface contains a multiplexed address and data bus input and output data bus with output enable which can be connected to either a Microcontroller or Microprocessor.

Core Modifications

Virtual IP Group can perform modifications on the Xilinx netlist version of this core for additional cost. This includes adding or removing blocks, or creating a different controller interface.

An Elastic Store function (in Verilog source format only) is available for additional cost. Contact Virtual IP Group for more information.

Pinout

Signal names for the core are shown in Figure 1 and described in Table 1.

Verification Methods

The FPGA core was tested through simulation.

Recommended Design Experience

Users should be familiar with T1 and related standards as well as Xilinx design flows.

Ordering Information

This product is available directly from Virtual IP Group, Inc. Please contact them for further information or pricing.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Transmitter Interface Signals		
TPOS	Output	Transmit Bipolar Positive Data: Updated on rising edge of TCLK. NRZ data is output when TCR1.7 = 1.
TNEG	Output	Transmit Bipolar Negative Data: Updated on rising edge of TCLK. Zero is output when TCR1.7 = 1.
TSER	Input	Transmit Serial Data: Transmit NRZ serial data, sampled on falling edge of TCLK.
TSYNC	I/O	Transmit Sync: A pulse on TSYNC will set either frame or multiframe boundaries. Can be programmed to output either a frame or multiframe pulse; can also be set to output double-wide pulses at signaling frames.
TCLK	Input	Transmit Clock: 1.544 MHz Primary clock.
TCHBLK	Output	Transmit Channel Block: output; can be forced high or low during any of 24 T1 channels.
TLINK	Input	Transmit Link Data: If enabled, TLINK will be sampled during F-bit time on falling edge of TCLK for data insertion into either FDL stream (ESF) or Fs bit position (D4).
TLCLK	Output	Transmit Link Clock: 4 KHz demand clock for TLINK input.
TCHCLK	Output	Transmit Channel Clock: 192 KHz clock pulses high during LSB of each channel.
TABCD	Input	Transmit ABCD signaling: When enabled, data on this line is sampled on negedge of TCLK and inserted into the transmitted NRZ stream at signaling frames.
Receiver Interface Signals		
RPOS	Input	Receive Bipolar Positive Data Input: Sampled on falling edge of RCLK. Tie together to receive NRZ data and disable bipolar violation monitoring circuitry.

Signal	Signal Direction	Description
RCLK	Input	Receive Clock: 1.544 MHz primary clock.
RNEG	Input	Receive Bipolar Negative Data Input: Sampled on falling edge of RCLK.
RLINK	Output	Receive Link Data: Updated with FDL data (ESF) or Fs bits (D4) one RCLK before start of frame.
RLCLK	Output	Receive Link Clock: 4 KHz demand clock for RLINK.
RCHBLK	Output	Receive Channel Block: programmable, can be forced high or low during any of 24 T1 channels.
RCHCLK	Output	Receive Channel Clock: 192 KHz clock, pulses high during LSB of each channel.
RSER	Output	Received NRZ Serial Data; updated on rising edges of RCLK.
RSYNC	Output	Receive Sync: An extracted pulse, (one RCLK wide) identifies either frame or multiframe boundaries. Can also be set to output double-wide pulses on signaling frames.
RFER	Output	Receive Frame Error: High during F-bit time when Ft or Fs errors occur in D4 mode or when FPS or CRC errors occur in ESF mode. Low during resync.
RABCD	Output	Receive ABCD signaling: Outputs received signaling data.
LOTC/RLDS	Output	Receive Loss of Sync/Loss of Transmit Clock: dual function. If CCR1.6=0, signal will toggle high when synchronizer is searching for T1 frame and multiframe. If CCR1.6=1, signal will toggle high when TCLK has not been toggled for 5µs.
Microprocessor Interface Signals		
NCS	Input	Chip Select: must be low to access internal registers (for read or write).
NWR	Input	Microprocessor Write signal.
NRD	Input	Microprocessor Read signal.
ALE	Input	Address Latch Enable: serves to demultiplex bus.
AD(7:0)	In/Out	Multiplexed Address/Data Bus: 8-bit multiplexed address/data input bus. For addressing internal registers, only AD(5:0) are decoded.
NINT1	Output	Receive Alarm Interrupt 1: Flags host controller during alarm conditions defined in Status Register 1.
NINT2	Output	Receive Alarm Interrupt 2: Flags host controller during conditions defined in Status Register 2.
NRESET	Input	Reset input: Active low reset. Minimum pulse width should be one clock period (either TCLK/RCLK) duration.

UTOPIA Slave (CC-141)

CoreEl MicroSystems

CoreEl MicroSystems

46750 Fremont Blvd. #208
Fremont, CA 94538 USA
Phone: +1 510-770-2277
Fax: +1 510-770-2288
URL: www.coreel.com
E-mail: sales@coreel.com

Features

- Pre-defined implementation for predictable timing in Xilinx FPGA or HardWire[a]
- Conforms to ATM Forums UTOPIA Level 2 specifications, Version 1.0
  - Support for 25 MHz, 33 MHz and 50 MHz operation
- Conforms to cell transfer procedure required in ATM UNI devices
- 8 bit / 16 bit UTOPIA operation
- SPHY operation in both octet-level handshake and cell-level handshake
- MPHY operation in cell level handshake supports up to 31 PHY devices
- Detection and dropping Runt cells on transmit side
- Dropping excess bytes of a cell on transmit side
- Parity verification on transmit side
- Parity indication on receive side
- Optional cell drop on the occurrence of parity error(s) on transmit side
- It supports cell of 52 bytes as well as cell of 53 bytes in case of 8 bit UTOPIA operation and cell of 52 bytes as well as 54 bytes in case of 16 bit UTOPIA operation

AllianceCORE™ Facts

Core Specifics	
Device Family	XC4000XL
CLBs - Transmitter: CLBs - Receiver:	381 - 265
IOBs - Transmitter: IOBs - Receiver:	51[1] 46[1]
CLKIOBs - Transmitter: CLKIOBs - Receiver:	2 2
System Clock f_{max}	50 MHz
Device Features Used	SelectRAM™, Global Clocks

Supported Devices/Resources Remaining	I/O[1]	CLBs
XC4020XL-09 BG256 (Tx)	154	403
XC4020XL-09 BG256 (Rx)	159	519

Provided with Core	
Documentation	Product Brief Specification Document Design Document Test Bench Design Document Test Scripts
Design File Formats	VHDL Compiled, XNF netlist
Constraint Files	Transmitter: txut.ucf Receiver: rxut.ucf
Verification Tool	Script Based behavioral VHDL Test Bench
Schematic Symbols	None
Evaluation Model	Behavioral
Reference Designs and Application Notes	UTOPIA Level 2 Specification V1.0.
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	Alliance 1.3
Entry/Verification Tool	Model Tech V-System

Support
Support provided by CoreEl Microsystems

Note:
1. Assuming all core signals are routed off-chip.

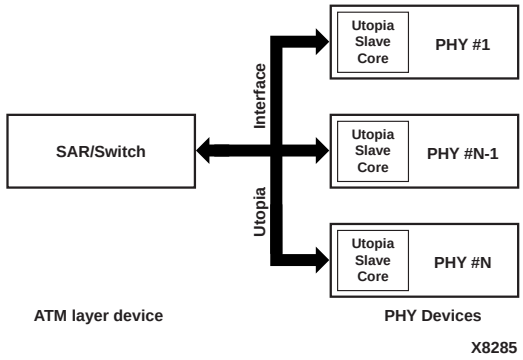


Figure 1: Data transfer between UNI device and a UTOPIA compatible ATM layer device

Applications

The UTOPIA core can be used in Asynchronous Transfer Mode (ATM) networking systems such as adapter cards, routers and switches.

General Description

The Slave UTOPIA Core Cell (SUC) is a UTOPIA Slave that can be used in any Physical Layer Device (PHY) that performs the functions of the Transmission Convergence sublayer. The SUC facilitates data transfer between the UNI (User Network Interface device) and a UTOPIA compatible ATM layer device as shown in Figure 1. In SPHY operation, it supports octet level handshake and cell level handshake. In MPHY operation, it supports up to 31 PHY devices.

Functional Description

The Slave UTOPIA core is architecturally divided into Transmit and Receive blocks as shown in Figure 2 and Figure 3, respectively. Operation of each are described below.

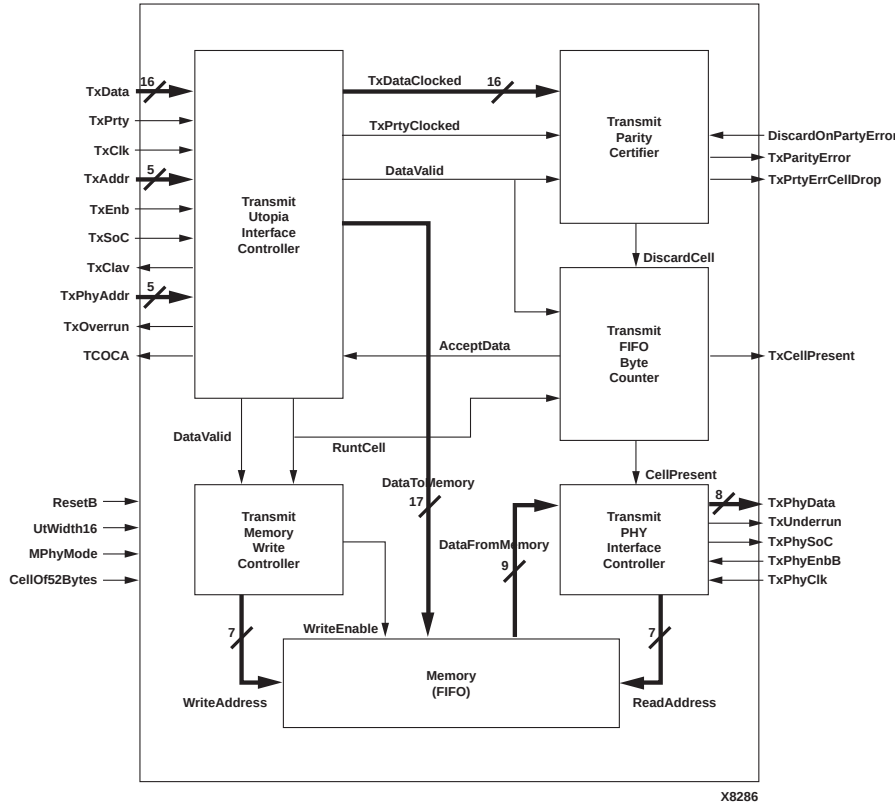


Figure 2: Transmit UTOPIA Block Diagram

Transmitter Block Operation

Transmit UTOPIA provides an industry standard interface between the ATM layer and a PHY device through a 216-Byte (4 cell) deep rate-matching buffer (FIFO). It supports both SPHY and MPHY modes of operation.

The Transmit interface is controlled by the ATM layer. The ATM layer provides an interface clock to the Transmit UTOPIA core for synchronizing all interface transfers. Data flow in the transmit interface is in the same direction as the ATM enable. Signals TxEnb, TxData, TxSoC, and TxPrty are sampled on the rising edge of the TxClk.

It accepts 54, 53, or 52 byte cells from the ATM layer, writes them into an internal FIFO with TxClk. Cells are read from the FIFO with TxPhyClk and sent to a cell processing device (cell processor) within the PHY. It detects and discards runt cells and excess bytes. It also performs parity check and gives the user the option of discarding the cells on the occurrence of parity errors.

The UTOPIA core indicates it can accept data using the TxClav signal, then the ATM layer drives data onto TxData and asserts TxEnb. The UTOPIA core controls the flow of data via the TxClav signal.

Receive Operation

The Receive operation is controlled by the ATM layer. The ATM layer provides an interface clock to the UTOPIA core to synchronize all transfers. The receive interface has data flowing in the opposite direction as the ATM enable.

The ATM receive block generates all output signals on the rising edge of RxClk. Signals RxData, RxSoC, RxPrty and RxEnb are sampled by the UTOPIA core on the rising edge of RxClk.

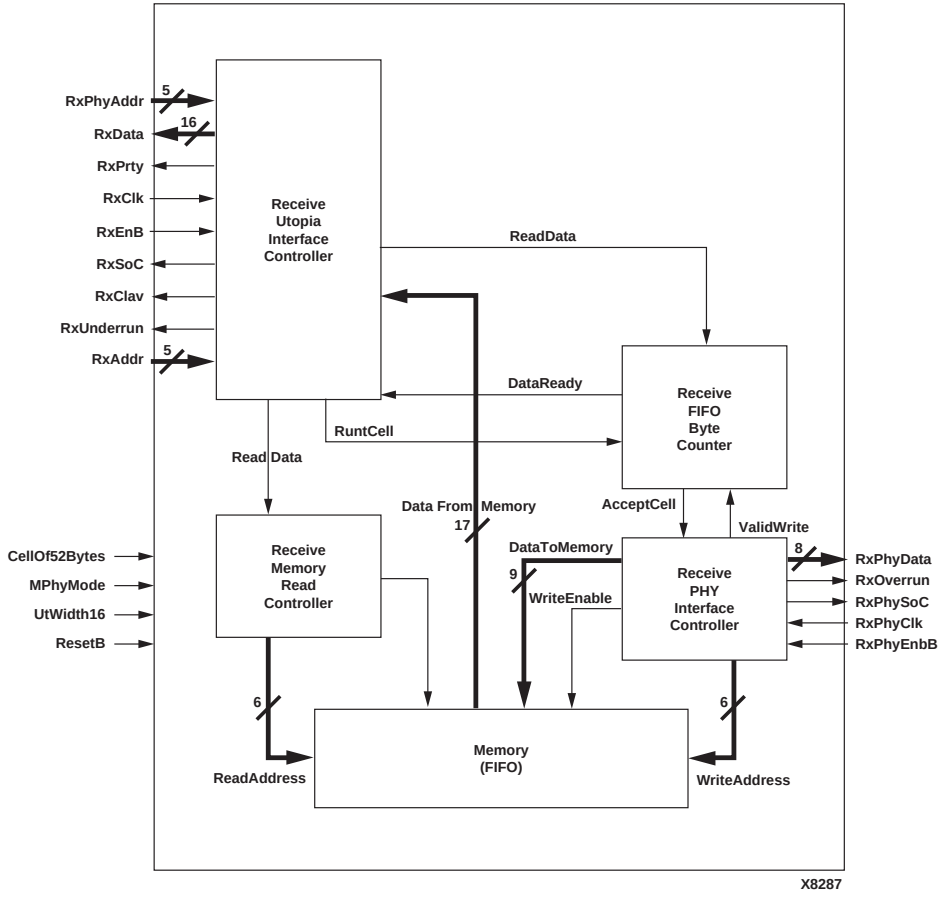


Figure 3: Receive UTOPIA Block Diagram

Receive data is transferred from the UTOPIA core to the ATM layer through the following procedure. The UTOPIA core indicates it has valid data; then the ATM layer asserts RxEnb to read this data from the UTOPIA core. The UTOPIA core indicates valid data via the RxClav signal.

Receive UTOPIA supports both SPHY (Cell-Level handshake or Octet-Level handshake) and MPHY modes of operation. It can accept 53, or 52 byte cells from the cell processing device.

For 16 bit UTOPIA operation, it converts cell of 53 bytes into cell of 54 bytes by adding one junk byte as the 6th byte of the cell, and directly writes them into an internal 108 byte (2 cell) deep rate matching buffer (FIFO) using RxPhyClk. It then reads the cells from the FIFO with RxClk, and sends them to the ATM layer.

It gives 52, 53 or 54 byte cells to the ATM layer depending on Cellof52Bytes and Utwidth16 signals. It generates Odd parity for data on the UTOPIA Interface.

Core Modifications

Normally, modifications are not possible by the user since the core is provided in a Xilinx netlist format. CoreEl can perform special modifications for additional charge. However source code is available for additional cost where the customer can make modifications. Contact CoreEl Microsystems for more information.

Pinout

The pinout is not fixed to any specific device I/O. Signal names for the transmit and receive UTOPIA blocks are provided in the block diagrams shown in Figures 2 and 3, and described in Tables 1 and 2, respectively.

Verification Methods

This core has been used in larger ASICs and is silicon proven. The FPGA verification was done by back annotating the implementation and simulating in a Model technology V-System environment.

The test bench was written in VHDL with very powerful scripting capabilities and several scripts have been written for verifying the implementation. Additional tests can be added to the testbench by writing new scripts.

Recommended Design Experience

Knowledge of ATM technology and UTOPIA Level 2 specifications is needed. User should be familiar with HDL design methodology including FPGA targeting. Using the testbench requires familiarity with V-System of Model Technology.

Table 1: Transmit UTOPIA Block Signal Pinout

Signal	Signal Direction	Description
UTOPIA Transmit Interface		
TxData(7:0)	Input	Least significant octet of transmit data, driven from ATM to PHY. Bit 7 is MSB, bit 0 is LSB in 8-bit data path.
TxData(15:8)	Input	Most significant octet of transmit data, driven from ATM to PHY. Bit 15 is MSB, bit 0 is LSB in 16-bit data path.
TxPrty	Input	Data path parity; odd parity bit over TxData(7:0) in 8 bit mode or over TxData(15:0) in 16 bit mode.
TxClk	Input	Transmit Master Clock; ATM layer provides interface clock to PHY layer to synchronize all information transfers. Uses 1 FPGA CLKIOB pin.
TxAddr[4:0]	Input	Address of MPHY device. True data driven from ATM layer to poll and select appropriate MPHY device (port in presence of multiple TxClav signals); Bit 4 is MSB. Value of address for Tx and Rx UTOPIA of MPHY devices must be identical.
TxEnb	Input	Enable, validates data on TxData lines.
TxSoC	Input	Start of cell indicator.
TxClav	Output	Cell Available; active high signal from SPHY device to ATM layer in SPHY mode; active high tri-state signal from MPHY device to ATM layer in MPHY mode. A polled MPHY device (port) drives TxClav only during each cycle following its address on TxAddr.
TxOverrun	Output	Indicates FIFO full; active high.
TCoca	Output	Indicates change of cell alignment (from either runt cell or a cell with excess bytes).
TxParityError	Output	Indicates parity errors
TxPrtyErrCellDrop	Output	High signal if cell is dropped due to TxPrty error.

Signal	Signal Direction	Description
PHY Layer Transmit Interface		
TxCellPresent	Output	Indication to cell processor of presence of a complete cell in FIFO.
TxPhyData(7:0)	Output	Data output to cell processor; bit 7 is MSB, bit 0 is LSB.
TxUnderrun	Output	Indicates FIFO empty; active high.
TxPhySoC	Output	Start of cell indication to cell processor.
TxPhyEnbB	Input	Read enable; active low. Indicates valid data present on TxPhyData.
TxPhyClk	Input	Cell Clock, from cell processor; synchronizes all information transfers with UTOPIA core. Uses 1 FPGA CLKIOB pin.
Configuration Signals		
TxPhyAddr(4:0)	Input	PHY device address; provided by UNI device.
ResetB	Input	Power-on reset; active low.
UtWidth16	Input	Indicates 16 bit data path; active high.
MPhyMode	Input	MPHY mode indicator; active high.
CellOf52Bytes	Input	Indicates cells of 52 bytes; active high. Low value = cell of 53 bytes when UtWidth16 is low, and = 54 bytes when UtWidth16 is high.
DiscardOnPrty-Error	Input	Indicates cells are discarded on occurrence of parity error; active high

Table 2: Receive UTOPIA Block Signal Pinout

Signal	Signal Direction	Description
UTOPIA Receive Interface		
RxData(7:0)	Output	Low octet of receive data from PHY to ATM; tristated when device (port) not selected by UTOPIA core. Bit 7 is MSB, bit 0 is LSB for 8-bit data path.
RxData(15:8)	Output	High octet of receive data from PHY to ATM; tristated when device (port) not selected by UTOPIA core. Bit 15 is MSB, bit 0 is LSB for 16-bit data path.
RxPrty	Output Tristated	Data path parity; odd parity bit over RxData(7:0) in 8 bit mode or over RxData(15:0) in 16 bit mode. Tristated when device (port) not selected by UTOPIA core.
RxClk	Input	Clock from ATM to PHY; synchronizes all transfers Uses 1 FPGA CLKIOB pin.
RxEnb	Input	Receive Enable; validates data on RxData.
RxSoC	Output Tristated	Start of cell. Tristated for multiple PHYs, and enabled only in cycles following those with RxEnb asserted.
RxClav	Output Tristated	Cell Available. Active high signal from SPHY device to ATM layer in SPHY mode. In MPHY mode, it is active high tri-state signal from MPHY device to ATM layer. Polled MPHY device (port) drives RxClav only during cycle following one with its address on RxAddr lines.
RxUnderrun	Output	FIFO empty indicator; active high.
RxAddr[4:0]	Input	MPHY device address. True data from ATM to MPHY layer to poll and select appropriate MPHY device (port in presence of multiple RxClav signals). Bit 4 is MSB. Value of address for both Tx and Rx UTOPIA of MPHY devices must be identical.

Signal	Signal Direction	Description
PHY Layer Receive Interface		
RxPhyData(7:0)	Input	Data input from cell processor. Bit 7 is MSB, bit 0 is LSB.
RxOverrun	Output	FIFO full indicator; active high.
RxPhySoC	Input	Start of cell indication from cell processor.
RxPhyClk	Input	Cell Clock from cell processor; synchronizes all transfers with UTOPIA core. Uses 1 FPGA CLKIOB pin.
RxPhyEnbB	Input	Indicates valid data on RxPhyData; active low.
Configuration Signals		
RxPhyAddr (4:0)	Input	PHY device address; provided by UNI device.
CellOf52Bytes	Input	Indicates cells of 52 bytes; active high. Low value = cell of 53 bytes when UtWidth16 is low, and = 54 bytes when UtWidth16 is high.
MPhyMode	Input	MPHY mode indicator; active high.
UtWidth16	Input	Indicates 16 bit data path; active high.
ResetB	Input	Power-on reset; active low.
UTOPIA Receive Interface		
RxData(7:0)	Output	Low octet of receive data from PHY to ATM; tristated when device (port) not selected by UTOPIA core. Bit 7 is MSB, bit 0 is LSB for 8-bit data path.
RxData(15:8)	Output	High octet of receive data from PHY to ATM; tristated when device (port) not selected by UTOPIA core. Bit 15 is MSB, bit 0 is LSB for 16-bit data path.
RxPrty	Output Tristated	Data path parity; odd parity bit over RxData(7:0) in 8 bit mode or over RxData(15:0) in 16 bit mode. Tristated when device (port) not selected by UTOPIA core.
RxClk	Input	Clock from ATM to PHY; synchronizes all transfers Uses 1 FPGA CLKIOB pin.

Available Support Products

CoreEl offers a test bench for verifying the SUC along with a complete line of ATM Core Cells for Xilinx FPGAs that perform the following functions:

- Cell Delineation
- Cell Stream Assembly
- CRC-32
- CRC-10

Ordering Information

For information on this or other products mentioned in this datasheet, contact CoreEl Microsystems directly from the information provided on the front page.

Related Information

The ATM Forum

The ATM Forum publishes specifications regarding ATM. For more information, contact them as follows:

ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1 650-949-6700
Fax: +1 650-949-6705
E-mail: info@atmforum.com
URL: www.atmforum.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Viterbi Decoder

February 8, 1998 Product Specification

CAST, Inc.

24 White Birch Drive
Pomona, New York 10907 USA
Phone: +1 914-354-4945
Fax: +1 914-354-0325
E-Mail: info@cast-inc.com
URL : www.cast-inc.com

Features

- Hard Decision Decoder
- Trace-back method for survivor memory
- Branch Metrics computations can be added by CAST for different applications
- Parameterized architecture allows for customization of the following functions:
  - Number of states in the trellis
  - Number of bits to represent transition values
  - Add-Compare-Select (ACS) cells
  - The length of the trace-back
  - The length of the received burst
  - The initial path metric for state 0
  - The survivor memory (RAM) word length

Applications

- Wireless telecommunication (satellites, digital cellular phones, etc.)
- Consumer electronics (CD players, etc.)

AllianceCORE™ Facts

Core Specifics[1]		
Device Family	Spartan	XC4000XL
CLBs Used	425	425
IOBs Used	109[2]	109[2]
System Clock f_{max}	11.55 Mhz	19.98 Mhz
Device Features Used	Global Buffers	

Supported Devices/Resources Remaining[1]		
	I/O	**CLBs**
XC4013XL-09	82[2]	151
XCS30-3	82[2]	151

Provided with Core	
Documentation	Core Documentation Sample Decoder
Design File Formats	.ngo or XNF Netlist or VHDL Source RTL
Constraint Files	NCF
Verification Tool	VHDL
Schematic Symbols	Viewlogic
Evaluation Model	None
Reference designs & application notes	Sample Design
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	M1.3
Entry/Verification Tool	VHDL RTL

Support
Support provided by CAST, Inc.

Notes:
1. Results obtained for the parameters in the table below.
2. Assuming all core signals are routed off-chip.

Parameter	Value
Number of States in trellis	16
Number of bits to represent transition values	8
Add-Compare-Select (ACS) cells	4
Length of the trace-back	30
Length of the received burst	5
Initial path metric for state 0	-4
Survivor memory (RAM) word length	8

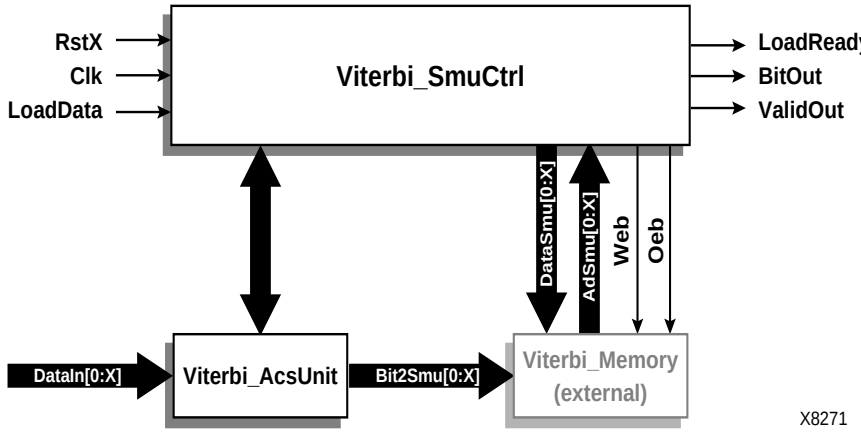


Figure 1: Viterbi Block Diagram

General Description

A Viterbi decoder is used for decoding convolutional codes. It can also be used to produce the maximum likelihood estimate of the transmitted sequence over a channel with inter-symbol interference (ISI).

The source code version of the core is parameterizable and therefore easily adapted to a wide variety of applications. The data given in this fact sheet is for an example only. The Viterbi Decoder can be delivered in a specific format (see Ordering Information).

Functional Description

The Viterbi Decoder is partitioned into modules as shown in Figure 1 and described below.

Viterbi_AcsUnit

ACS (add-compare-select) unit calculates the path metrics to find the minimum path. The number of AcsUnits is parameterizable.

Viterbi_SmuCtrl

SmuCtrl block handles the survivor memory management. The state-machine controls the alternation of reading new branch metrics and trace-back. During trace-back it reads the decision values from the memory and outputs the decoded bits. It reconstructs the actions of the encoder in the reversed order by updating the State-register with a decision value pointed by the former State-value. As a result of it, the decoded bits are also output in reverse order.

Viterbi_Memory

Viterbi memory is a RAM memory which stores the trace-back values during calculation. This block is external to the core and is added by the user or can be included by CAST for additional cost.

Core Modifications

The Viterbi Decoder core can be customized to include:

- Number of states in the trellis
- Number of bits to represent transition values
- Add-Compare-Select (ACS) cells
- The length of the trace-back
- The length of the received burst
- The initial path metric for state 0
- The survivor memory (RAM) word length
- Addition of Viterbi Memory block

Please contact CAST directly for any required modifications. Use the fax-in checklist at the end of this specification.

Pinout

The pinout of the Viterbi Decoder core has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are shown in the block diagram in Figure 1, and in Table 1.

Verification Methods

The core model has been extensively tested using various settings of the parameters. A special encoder circuit is used as a testbench for full testing of the decoder.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Input Interface Signals		
RstX	Input	Asynchronous reset, active low
Clk	Input	Common Clock
LoadData	Input	Enables Data Loading
DataIn[0:X]	Input	Transition values input, the width depends on the number of ACS-units
Output Interface Signals		
LoadReady	Output	Indicates Ready for Data
BitOut	Output	Decoded Output Bit
ValidOut	Output	Indicates the Valid Output Data
External Memory Interface Signals		
Bits2Smu[0:X]	Output	Output data to external Memory
DataSmu[0:X]	Input	Input data from external Memory
Web	Output	Write Enable to external Memory
Oeb	Output	Read Enable to external Memory
AdSmu[0:X]	Output	Address to external Memory

Recommended Design Experience

The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment.

Ordering Information

The Viterbi Decoder is available for purchase directly from CAST, Inc. If the XNF Netlist format is required, a setting for the parameters will have to submitted before the netlist can be created. Ask CAST, Inc. for details.

Related Information

A.B. Carlson, Communications systems, McGraw-Hill, 1986.

E.A. Lee, D.G. Messerschmitt, Digital Communication, Boston, MA Kluwer Academic Publishers, 1988.

G.D. Forney, Jr., “The Viterbi algorithm:, Proc. IEEE, vol. 61, pp.268-277, March 1973.

G. Fettweis, H. Meyr, “High-Speed Parallel Decoding: Algorithm and VLSI-Architecture”, IEEE Comm. Magazine, pp. 46-55, May 1991.

G. Feygin, P.G.Gulak, “Architectural Tradeoffs for Survivor Sequence Memory Management in Viterbi decoders”, IEEE Trans. Commun. Tech., vol. 41, no. 3, March 1993.

R. Cypher, C.B.Shung, “Generalized Trace-Back techniques for Survivor Memory Management in the Viterbi Algorithm”, Journal of VLSI Signal Processing, 5, 85-94 (1993).

Motorola, “Convolutional Encoding and Viterbi Decoding Using the DSP56001 with a V.32 Modem Trellis Example, Motorola Inc. 1989.

B. Sklar, “Digital Communications, fundamentals and applications”, Prentice Hall International Inc. 1988.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Viterbi Implementation Request Form

To: **CAST, Inc.**
FAX: **+1 914-354-0325**
E-mail: **info@cast-inc.com**

From: _____

Company: _____

Name: _____

Address: _____

Country: _____

Phone: _____

Fax: _____

E-mail: _____

CAST, Inc. will customize and deliver a Xilinx optimized netlist for the specific Viterbi Decoder implementation you require. Please fill out and fax or email this information to CAST so they can provide you with an accurate response to, and quote for your requirements.

Implementation Issues

1. Number of states in trellis: _____

2. Number of bits to represent transition values: _____

3. Add-compare-select cells: _____

4. Length of trace-back: _____

5. Length of receiveR burst:_____

6. Initial path metric for state 0: _____

7. Survivor memory word length: _____

Business Issues

8. Indicate timescales of requirement:
   a. date for decision _____
   b. date for placing order _____
   c. required delivery date _____

9. Indicate your area of responsibility:
   _____ decision maker
   _____ budget holder
   _____ recommender

10. Has a budget been allocated for the purchase?
    Yes _____ No _____

11. What volume do you expect to ship of the product that will use this core? _____

12. What major factors will influence your decision?
    _____ cost
    _____ customization
    _____ testing
    _____ implementation size

13. Are you considering any other solutions?

T7L Technology, Inc.

220 Duncanmill Road
Suite 307
North York, Ontario
Canada, M3B 3J5
Phone: +1 416-445-4783
Fax: +1 416-445-5741
E-mail: t7linfo@t7l.com
URL: www.t7l.com

Features

- Scalable RISC CPU cores for embedded controller applications
- Compact RISC-oriented opcode design makes the core very small and requires less program size
- 26 different configurations, depending on application requirements
  - 8 - 32 bit data width
  - 8 - 32 bit data address width
  - 10 - 24 bit instruction length
  - 10 - 24 bit program address width
  - 1K - 16M bit maximum program size
- Function generator based logic structure and fully optimized for Xilinx FPGA physical architecture
- Up to 33Mhz operation
- Three stage, fully-balanced pipeline machine for prefetch/branch, instruction decode and execution
- Built-in hardware stack unit for instant program switching and low latency interrupt dispatching
- Debug version with snoop capability provided with each production core
- Ships with integrated development toolkit for CPU core embedding and run-time debugging
  - Support for high-level programming languages
  - Symbolic Assembler/Debugger
  - C Compiler

AllianceCORE™ Facts

Core Specifics	
Device Family	XC4000E, EX, XL
CLBs Used	See Table 1
IOBs Used	
System clock fmax	
Device Features Used	Select RAM
Supported Devices/Resources Remaining	
See Table 1	
Provided with Core	
Documentation	Design Manual H/W & S/W, Compiler, Assembly & C
Design File Formats	EDIF, .xnf, .ngo netlists
Constraint Files	N/A
Verification Tool	Development Software
Schematic Symbols	Foundation
Evaluation Model	Design base board
Reference designs & application notes	Music Panel sample design in VHDL and Schematics
Additional Items	PC Windows-based Debugger, assembly compiler and vector generator; Debug Version of core
Design Tool Requirements	
Xilinx Core Tools	Alliance/Foundation 1.3
Entry/Verification Tool	VHDL or Schematic
Support	
Support provided by T7L Technology, Inc.	

Applications

- Microcontroller Unit (MCU)
- Real-time Digital Signal Processor
- Dedicated Communication Processor
- Integrated Embedded System
- Programmable Peripheral Controller

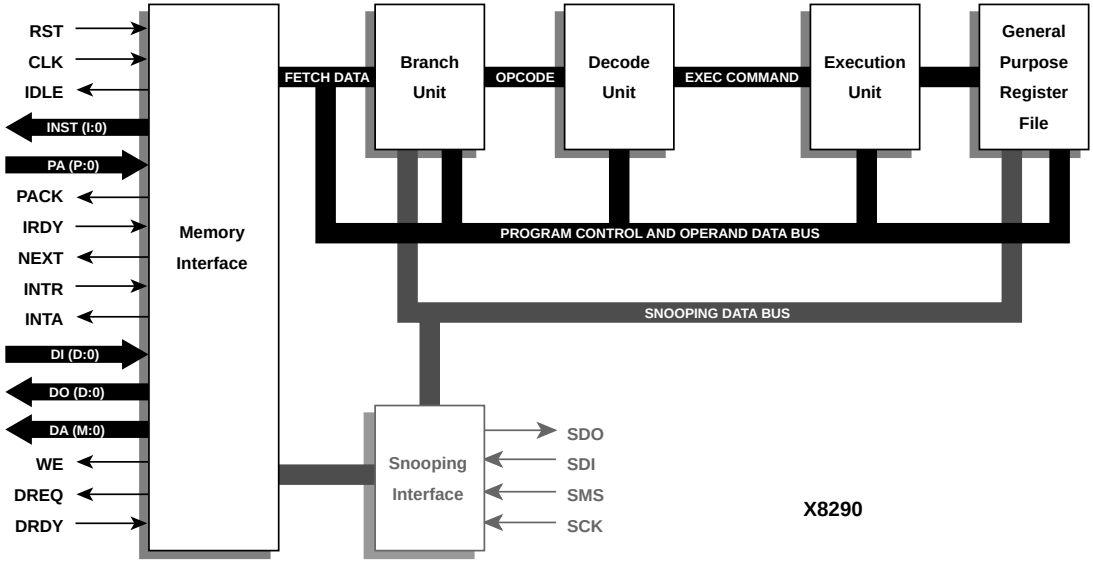


Figure 1: TX400 Block Diagram

General Description

The TX400 series RISC CPU core are a compact designs, optimized for Xilinx XC4000 FPGAs. T7L provides a full hardware development platform and software development tools that allow the user to integrate their unique logic with the core.

Functional Description

The TX400 core is based on a scalable 3-stage pipeline RISC CPU architecture. Separate instruction and data bus structures allow concurrent operations between program fetching and data transfer. A three-stage pipeline mechanism speeds up the system clock rate and simplifies the CPU state machine design.

The TX400 is partitioned into blocks as shown in Figure 1, and described below.

Memory Interface

This block interfaces with external system devices through the CPU bus. It transfers data to and from Data Memory Space, fetches instructions from Program Memory Space Convey and acknowledges external interrupt requests.

Snooping Interface

This block contains a serial data interface to an external snooping controller. It transfers data to and from CPU internal registers and modifies data memory contents.

Branch Unit

This performs program flow and interrupt control functions and generates next instruction prefetch address. It includes a built-in hardware stack for call and interrupt services.

Decode Unit

This decodes instructions and generates execution commands; compiles and exports source and target register addresses; provides data flow control command for execution.

Execution Unit

This executes data path commands and updates result to target operand; communicates with data bus for load/store operations.

General-purpose Register File

This is a dual-port register file for storing operand data. The register file supports simultaneous read and write operations.

Core Modifications

T7L offers 26 versions of the TX400 CPU core as fixed function Xilinx netlists. This broad selection should meet the vast majority of customer requirements. If you need a configuration that is not listed in Table 2, then contact T7L to discuss the possibility of supplying either a custom netlist or source code to meet your need.

Pinout

A simple handshake protocol is employed in the CPU bus interface. Signals can be selected as either active high or active low and are fully synchronous to either the rising or falling edge of the CPU clock input.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
System Signals		
RST	Input	CPU Reset; force CPU into initial known state for bootstrapping when reset is released.
CLK	Input	CPU Clock; synchronizes all internal logic and function units.
IDLE	Output	CPU Idle; indicates CPU is waiting for new instruction.
Instruction Bus		
INST (I:0)	Input	Instruction Input; fetches instruction words from program memory space, qualified by IRDY during normal operation and by INTA for importing interrupt address during interrupt acknowledgment.
PA (P:0)	Output	Program Address; fetches instructions from program memory space, aligned with instruction word length 'I'.
PACK	Output	Prefetch Acknowledge that instruction word has been latched.
IRDY	Input	Instruction Ready for fetching.
NEXT	Output	Next Program Location; used for continuous burst access in program memory space for instruction fetching with location next to previous cycle.
INTR	Input	Interrupt Request for any asynchronous event. signal is.
INTA	Output	Interrupt Acknowledge; informs external logic to present interrupt address on instruction input bus; ends acknowledgment cycle.
Data Bus		
DI (D:0)	Input	Data Input bus for load data from data memory space, qualified by DRDY.
DO (D:0)	Output	Data Output bus for store data; output data lines held until CPU receives DRDY from external memory controller. No endian rule required.
DA (M:0)	Output	Data Address bus for data memory operations, qualified by DREQ; address lines held until CPU receives DRDY from external memory controller. M = data memory size.
WE	Output	Write Enable control for current data bus transaction once exported.
DREQ	Output	Data Request; initiates bus cycle in Data Bus and waits for DRDY.
DRDY	Input	Data Ready; acknowledgment signal from external memory controller for both read and write operations.
Snooping Bus (On Debug Versions Only)		
SDO	Output	Snoop Data Out; serial data out signal during CPU snooping cycle, qualified by SCK.
SDI	Input	Snoop Data In; serial data input signal during CPU snooping cycle, qualified by SCK.
SMS	Input	Snoop Mode Select; controls snoop mode; 0 for command mode, 1 for snooping data transfer.
SCK	Input	Snoop Data Clock to perform serial data shift in and out from CPU core.

There are four groups of signals (All scalable variables arezero basis).

- System Signals - for overall CPU control and status information
- Instruction Bus - for fetching instruction words from Program Memory Space
- Data Bus - for load or store of data into Data Memory & IO Space as requested by program

Snooping Bus - signals for reading or modifying CPU internal registers and data memory contents. These signals are on the debug versions of the cores only.

The pinout is not fixed to any specific device I/O. Signal names are provided in the block diagram shown in Figure 1, and described in Table 1.

Verification Methods

Test vectors generated by the V-System V4.4J from Model Technology have been used to test every instruction, instruction combination and interrupt event for these cores. They have also been silicon proven in Xilinx XC4000 FPGAs using the T7L Design Base Board, which is available separately from T7L.

Selecting the Right Core

There are 26 pre-implemented versions of the TX400, divided into seven groups. The characteristics of each are shown in Table 2. Each version is purchased and shipped separately. Each core comes with Xilinx netlists for a debug version that includes snooping circuitry and I/O, and a production version without snoop capability. The implementation statistics shown in Table 3 are for the production version of each core.

- ***Group 1 TX401*** - Microcontroller type CPU cores for embedded applications, and requires a small instruction set. Sizes range from 8 to 10-bit data and 10 to 12-bit instruction length.
- ***Group 2 TX402*** - Standard RISC CPU cores for embedded controller applications. Its size ranges from 8 to 10-bit data and 12 to 16-bit instruction length.
- ***Group 3 TX403*** - Compact RISC CPU cores for embedded system applications and requires a small instruction set. Its size ranges from 12 to 16-bit data and 12 to 16-bit instruction length.
- ***Group 4 TX404*** - Mid-size RISC CPU cores for embedded system applications. Its size ranges from 12 to 16-bit data and 12 to 16-bit instruction length.
- ***Group 5 TX405*** - Mid-size RISC CPU cores for DSP system applications. Its size ranges from 12 to 16-bit data and 12 to 16-bit instruction length.
- ***Group 6 TX406*** - Powerful RISC CPU cores for embedded system applications. Its size ranges from 24 to 32-bit data and 16 to 24-bit instruction length.
- ***Group 7 TX407*** - Powerful RISC CPU cores for DSP system applications. Its size ranges from 24 to 32-bit data and 16 to 24-bit instruction length.

Recommended Design Experience

To integrate a TX-400 core in an embedded system with a Xilinx XC4000 FPGA requires a simple design flow. The core is supplied in the form of Xilinx .ngo, EDIF, .xnf netlists and schematic symbol. Users include the core by either capturing into the top-level schematic or declaring/instantiating as a component in the top-level VHDL model.

Users should be familiar with RISC processors, processor code development environments and embedded system design.

Available Support Products

T7L offers 26 versions of the TX400 RISC CPU core along with hardware tools for integrating them into Xilinx FPGAs. T7L also supplies a complete software development environment. Products include:

- SDP Design Base Board (with Xilinx XC4000 FPGA) - $295
- Windows95 based assembler, debugger, monitor and vector generator (provided with core)
- C-Compiler (provided with core)

Ordering Information

There are 26 different cores in the TX400 family. Specifications for each are detailed in Table 2. Each comes with Xilinx netlists for production and debug versions. VHDL source is available for additional cost. For more information on these or other T7L products, contact T7L Technology directly.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Table 2: TX400 Series RISC CPU Selector Guide

Core	Data Bits (D)	Data Address Bits (M)	Max Data Bytes	Instruction Length Bits (I)	Program Address Bits (P)	Max Program Bits	General Registers	Special Registers	Stack Unit Level	No. of Instruction
Group 1										
TX401A	8	8	256	10	10	1K	4	3	16	47 +
TX401B	8	8	256	12	12	4K	8	3	16	47 +
TX401C	10	10	1K	10	10	1K	4	3	16	47 +
TX401D	10	10	1K	12	12	4K	8	3	16	47 +
Group 2										
TX402E	8	10	1K	12	12	4K	4	3	16	80 +
TX402F	8	12	4K	16	16	64K	16	3	32	80 +
TX402G	10	12	4K	12	14	16K	4	3	32	80 +
TX402H	10	14	16K	16	16	64K	16	3	32	80 +
Group 3										
TX403I	12	12	4K	12	14	16K	8	3	16	47 +
TX403J	12	14	16K	16	16	64K	16	3	32	47 +
TX403K	16	14	16K	12	14	16K	8	3	32	47 +
TX403L	16	16	64K	16	18	1M	16	3	32	47 +
Group 4										
TX404M	12	12	4K	12	14	16K	4	3	32	80 +
TX404N	12	16	64K	16	16	64K	16	3	32	80 +
TX404O	16	16	64K	12	14	16K	4	3	32	80 +
TX404P	16	16	64K	16	18	1M	16	3	32	80 +
Group 5										
TX405Q	12	12	4K	12	14	16K	4	3	32	140 +
TX405R	12	14	16K	16	16	64K	16	3	32	140 +
TX405S	16	16	64K	12	14	16K	4	3	32	140 +
TX405T	16	16	64K	16	18	1M	16	3	32	140 +
Group 6										
TX406U	24	24	16M	16	18	1M	16	3	32	80 +
TX406V	32	32	4G	16	18	1M	16	3	32	80 +
TX406W	32	32	4G	24	24	16M	256	3	32	80 +
Group 7										
TX407X	24	24	16M	16	18	1M	16	3	32	140 +
TX407Y	32	32	4G	16	18	1M	16	3	32	140 +
TX407Z	32	32	4G	24	24	16M	256	3	32	140 +

Table 3: Example Xilinx Device Utilization Statistics

Device	Statistic	TX400 Series Core			
TX402		**E**	**F**	**G**	**H**
XC4000E	CLBs	150	200	200	250
	IOBs	61	71	69	77
TX403		**I**	**J**	**K**	**L**
XC4000E	CLBs	130	180	180	220
	IOBs	73	81	83	93
TX404		**M**	**N**	**O**	**P**
XC4000E	CLBs	200	230	260	300
	IOBs	73	83	85	93

RISC CPU Core Design Base Board

December 15, 1997 | Product Specification

T7L Technology, Inc.

220 Duncanmill Road
Suite 307
North York, Ontario
Canada, M3B 3J5
Phone: +1 416-445-4783
Fax: +1 416-445-5741
E-mail: t7linfo@t7l.com
URL: www.t7l.com

Features

- Xilinx FPGA-based development board for use with T7L RISC processor cores
  - Supplied with XC4010EPC84
  - Higher pin count packages accommodated through extra daughter card
  - Supports Xilinx X-Checker cable
- Use in conjunction with T7L Scalable Development Platform software
  - Develop and debug code for working FPGA version of processor
- Built in 2 Mbits FLASH ROM for programming firmware
- Built in 1 Mbits FLASH ROM for configuring Xilinx FPGA
- Sockets for up to 2 Mbits of data memory in asynchronous SRAM
- On-board 24-key input device with dedicated scan decode circuitry
- On-board standard parallel port interface for debugging

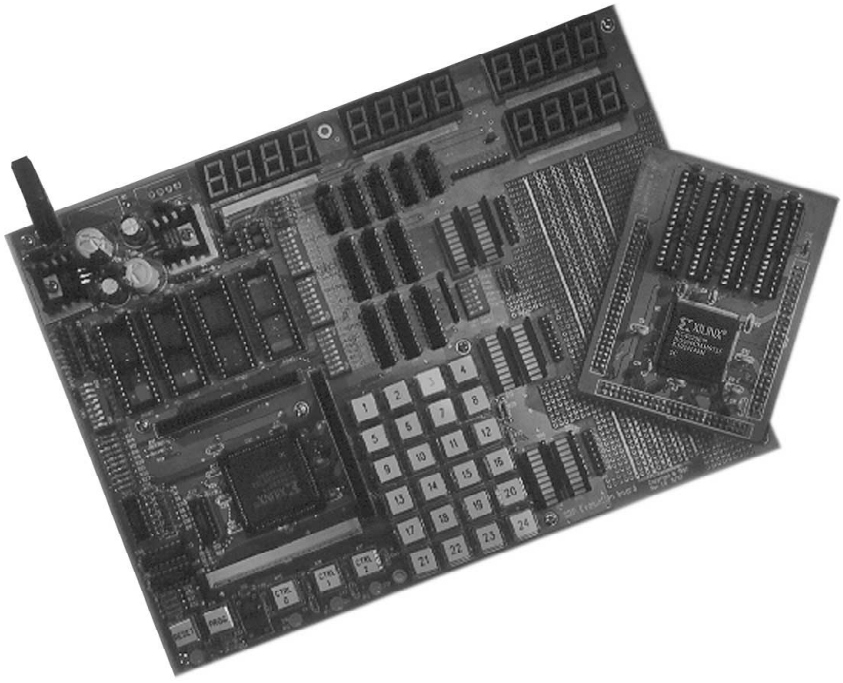


Figure 1: T7L RISC CPU Design Base Board

and downloading
- 48 user definable, general purpose I/O lines with 48 individual status LEDs
- 16 seven-segment LED display with dedicated decode circuitry

General Description

The T7L Scalable Development Platform (SDP) design base board is offered as part of the design kit for building prototypes of embedded systems using TX400 RISC CPU cores. It comes with a Xilinx XC4010EPC84 chip and some basic I/O devices.

The user is only required to glue together the CPU core, on board peripherals and application-specific I/O logic through either schematic capture or by creating a top-level VHDL model. With help of Xilinx place-and-route software, the system will then quickly be converted into physical configuration bitstream. Once the data file is downloaded into the on-board FPGA, the development board becomes a hardware prototype of the system running at speed for real-time software development, evaluation and debug.

Functional Description

The RISC CPU Core Design Base Board contains everything you need to develop and debug a RISC processor based FPGA design.

FPGA Chip

Basic on-board FPGA device is an XC4000E chip in a PLCC-84 package, upgradable to any XC4000E/EX/XL device in larger packages by means of a daughter board. Configuration data is downloaded from a PC system via standard parallel port.

Program Memory

There are two Flash ROMs for user's firmware program and initial data items. It supports 2 Mbits of program memory space. On-board programming utility for firmware codes is provided in the T7L SDP software tools.

Data Memory

There are two IC sockets for program data in asynchronous SRAM. It supports up to 1 Mbits of data memory space. Each IC socket is for any 32-pin DIP package static memory device, such 61 series fast SRAM devices.

Configuration Memory

There is one Flash ROM for user's Xilinx FPGA configuration. It supports 1 Mbits of FPGA configuration data, upgradable for 2 Mbits. On-board programming utility for FPGA configuration bit image is provided in the T7L SDP software tools.

Control Keys

There are seven input keys designed for control functions:

- *Reset*: User-defined reset input for FPGA and/or on board devices
- *Program*: To initialize the FPGA and wait for re-load of configuration data
- *Init*: To hold the FPGA and wait for re-load of configuration data via JTAG port.
- *User Control*: Four user-defined control input keys with external latching functions.

Keypad Matrix

A 24-key input device is provided on board. It is organized in a 3 x 8 matrix array. Scan decoding is supported externally by a TTL chip that saves a lot of I/O pins to the FPGA device.

LED Display

Total 16 seven-segment LED display units on board. They are time multiplexed and shared data lines. It is supported by an external TTL decoder in order to save I/O pins to and from an FPGA device.

User I/O Ports

There are 48 general-purpose I/O lines for user-defined logic functions, grouped into three 16-bit I/O ports. These can be connected to off-board peripheral device using cables. Each I/O line is supported by a status LED for real time logic debugging.

Download Port

An interface is provided to download configuration data and perform JTAG boundary scan testing. It supports the Xilinx X-Checker cable and software utility.

Parallel Port

An on-board 25-pin connector is built for CPU real-time hardware debugging, configuration data downloading and JTAG boundary scan testing. Hardware debug facilities for T7L CPU core is provided in the T7L SDP software tools.

Available Support Products

T7L also offers 26 versions of the TX400 RISC CPU core along with a development software. When used with the design base board, the customer has a complete development environment for integrating and testing T7L RISC cores in Xilinx FPGAs.

Ordering Information

The Design Base Board lists for $295 directly from T7L Technology. For more information on this or associated products, contact T7L directly.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Scalable Development Platform Integrated Software

T7L Technology, Inc.

220 Duncanmill Road
Suite 307
North York, Ontario
Canada, M3B 3J5
Phone: +1 416-445-4783
Fax: +1 416-445-5741
E-mail: t7linfo@t7l.com
URL: www.t7l.com

Features

- Software development platform for T7L RISC CPU cores
  - Supplied free with CPU core license
- Complete integrated software tool set
  - Project manager
  - Source code editor
  - C compiler
  - Assembler
  - Object file linker
  - Test vector generator
  - Software debugger
  - Hardware debugger
- Runs under Windows 95

General Description

The T7L Scalable Development Platform (T7L SDP) is a software development platform for T7L Scalable CPU products and is free for all T7L CPU core license users. It provides project management, source code editor, C compiler, assembler, object file linker, test vector generator and facilities to help in the program debugging process during soft-

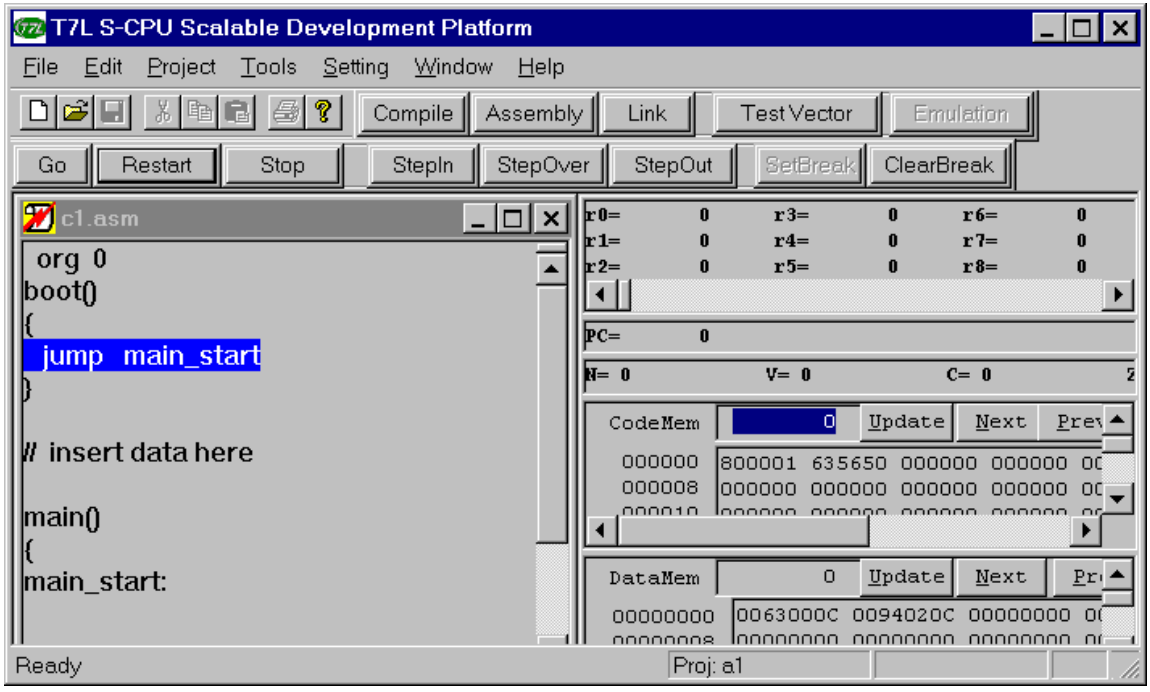


Figure 1: SDP Integrated Software Debugger Screen

ware and hardware development. SDP is designed to run under Windows 95 or upward compatibles. It does not work for Windows 3.1 or older versions.

Functional Description

The T7L SDP toolkit includes a complete set of software tools as described below.

Project Manager

It provides a GUI based project development environment for use in embedding T7L CPU core and in software programming. Various file house-keeping functions are supported for user's design data base.

Source Code Editor

This is a built-in text editor for source code and test vector editing. It provides an integrated design entry tool on the same development platform.

C Compiler

A C Compiler is provided for high-level language software programming. It is an embedded version of standard C language processor and generates assembly source code for further optimization.

Assembler

This is a super symbolic assembler for object code generation according to the user's target CPU configuration. The user can modify mnemonic and operand syntax strings as required. A new style of assembly programming was created to cope with the T7L CPU scalability.

Object File Linker

This helps the user to combine object codes and common library elements together and generates system firmware codes.

Test Vector Generator

This generates an input stimulus file in ASCII form for system simulation. The file can be merged with the user's system test vectors for chip level testing. The vectors also support memory file format (.mem) for Xilinx LogiBLOX RAM/ROM generator. With the Test Vector Generator, T7L SDP is perfectly integrated with Xilinx place and route tools in internal ROM-base program development.

Software Debugger

This is a virtual CPU simulator for the T7L scalable RISC processor. Software debugging can be carried out in parallel with the target hardware prototype. It supports debugging commands, Restart, Stop, Go, Step, Set Break and Clear Break, and displays all register and data memory contents.

Hardware Debugger

A hardware debugging tool is supplied for real time at-speed debugging. Using a standard parallel interface, the debugger can remotely read and modify internal CPU register and data memory contents in the target hardware system. Other than software debugging commands, the debugger also supports Xilinx FPGA configuration data downloading. A T7L SDP snooping bus port is defined for debugging during technical development and for on-site troubleshooting. An optional T7L SDP debugger cable is provided for on-site re-configuring and debugging.

Available Support Products

T7L offers 26 versions of the TX400 RISC CPU core each bundled with the SDP Integrated Software. When used with the SDP Design Base Board, customers have a complete development environment for integrating and testing T7L RISC cores in Xilinx FPGAs.

The software manual and all other documentation can be downloaded from the T7L web site at www.t7l.com.

Ordering Information

The SDP Integrated Software is bundled free with all T7L RISC CPU cores. For more information on this or associated products, contact T7L directly.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

V8-uRISC 8-bit RISC Microprocessor

February 8, 1998 | Product Specification

VAutomation, Inc.

20 Trafalgar Square
Nashua, NH 03063
Phone: +1 603-882-2282
Fax: +1 603-882-1587
E-mail: sales@vautomation.com
URL: www.vautomation.com

Features

- RISC architecture for low gate count and high performance
- Single Cycle instruction execution for all register to register opcodes
- High Code density, many opcodes are a single byte
- 33 opcodes, 4 addressing modes, 2 user defined opcodes
- 8-bit ALU
- 64K byte addressing capability
- Two banks of 8-bit general purpose registers
- 16-bit program counter and stack pointer
- Seven maskable interrupts, one non-maskable
- Assembler, debugger and simulator available now, C Compiler (May '98)
- IntelliCore™ Prototyping System available for evaluation and rapid product development
- Includes VHDL/Verilog source code for the following peripherals:
  - UART
  - DMA controller
  - I2C EEPROM interface
  - P1284 parallel interface
  - Page register

AllianceCORE™ Facts

Core Specifics[2]	
Device Family	XC4000E
Macrocells Used	268
IOBs Used	54[1]
CLKIOBs Used	1
System Clock fmax	15.9 MHz
Device Features Used	SelectRAM

Supported Devices/Resources Remaining[2]		
	I/O	**CLBs**
XC4013E-1 PQ160	74[1]	308

Provided with Core	
Documentation	Reference Manual
Design File Formats	XNF, EDIF, VHDL or Verilog netlist
Constraint Files	Example .ucf
Verification Tool	Test Bench
Schematic Symbols	None
Evaluation Model	None
Reference designs & application notes	Includes VHDL/Verilog source for UART, DMA controller, I2C EE-PROM I/F, P1284 I/F, page register
Additional Items	IntelliCore™ Prototype system available extra

Design Tool Requirements	
Xilinx Core Tools	Alliance 1.3
Entry/Verification Tool	FPGA Synthesis tool required for supplied peripherals; Model Tech for simulation

Support
Support provided by VAutomation.

Notes:
1. Assuming all core signals are routed off-chip.
2. For CPU and register file only. Figures do not include peripherals.

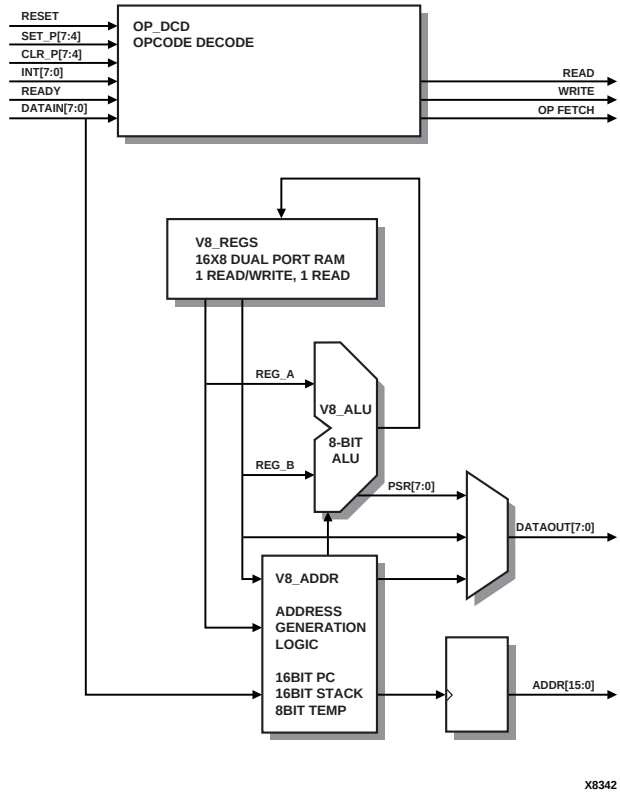


Figure 1: V8-uRISC 8-bit RISC Microprocessor Block Diagram

Applications

- 8-bit processing applications which need a combination of small size, good performance and low power
- Custom microcontroller for any embedded application

General Description

The V8-uRISC 8-bit RISC microprocessor is a general purpose processor core designed and optimized specifically for programmable logic. It combines a small gate count with single clock cycle execution for many instructions to deliver a high performance 8-bit microprocessor with a very small footprint.

The V8-uRISC is delivered with a set of standard peripherals including a timer, UART, DMA controller, I2C EEPROM interface, P1284 parallel interface, and page register design to expand the address space to 8 megabytes. The peripherals are delivered in non-optimized VHDL/Verilog source code. The V8-uRISC and its' peripherals allow high performance custom microcontrollers to be implemented within Xilinx FPGAs.

Functional Description

The V8-uRISC core is partitioned into modules as shown in Figure 1, and described below.

Instruction Decode (OP_DCD)

The Instruction Decode block controls the operation of the V8-uRISC. Opcodes are fetched from memory and loaded into the Opcode Register. The opcode is then decoded and the appropriate control signals are output. The control signals depend on the opcode and the number of clock cycles that have been executed for the particular opcode.

Most of the control outputs are decodes and thus may glitch and have a significant amount of delay before settling to their final state. However, WRITE comes directly from a Flip-Flop to insure there are no glitches.

The READY signal will hold the current state of the CPU and deactivates many of the register load control signals. The INT and RESET lines force an INT opcode to be loaded into the opcode register. Special processing is performed depending on the type of interrupt.

Arithmetic Logic Unit (V8_ALU)

The ALU block performs the 8 bit arithmetic and logical operations. The inputs to the ALU come directly from the Register File or from the TEMP register in the V8-uRISC_ADDR module. Certain constants such as 0 and 1 can also be selected as inputs to the ALU for increment and decrement operations.

The ALU includes bit-manipulation logic to allow testing, setting and clearing of a single bit of a byte. This is a common function in embedded microcontrollers.

The Processor Status Register (PSR) shown in Table 1, is also located within the V8_ALU block. Most opcodes pass data through the ALU so that the appropriate PSR flags can be updated. A logic 1 on the SET_P signals will cause the corresponding bit in the PSR register to be set; a logic 1 on the CLR_P pin will clear the corresponding PSR bit to zero. These signals are typically connected to IO device status lines that allow the V8-uRISC to quickly test and branch on these bits.

Table 1: Program Status Register (PSR)

Bit	Name	Operation
7	P7	FLAG bit, Set by SET_P[7], Cleared by CLR_P[7]
6	P6	FLAG bit, Set by SET_P[6], Cleared by CLR_P[6]
5	P5	FLAG bit, Set by SET_P[5], Cleared by CLR_P[5]
4	P4	FLAG bit, Set by SET_P[4], Cleared by CLR_P[4]
3	I	Interrupt MASK bit (1=ignore interrupts, 0=enable interrupts)
2	N	Negative bit (1 when ALU result has bit 7=1)
1	C	Carry bit (NOT borrow for subtract)
0	Z	Zero bit (1 when ALU result is zero)

Register File (V8_REGS)

The Register File is a 16x8, dual read, single write register file with two 8x8 banks. It is implemented as a dual port RAM with one port being read/write and the other port being read only.

The registers in the first bank are referred to as R0 through R7, as shown in Table 2. R0 can be thought of as the accumulator as it is the default source/destination for most opcodes. Typically R4 through R7 are used as index registers into RAM although they can be used for any purpose.

Table 2: Register File Use

Register	Typical Purpose
R0 [7:0]	Accumulator
R1 [7:0]	General Purpose Registers
R2 [7:0]	
R3 [7:0]	
R4 [7:0]	Register pairs form 16-bit addresses for indexed and offset addressing modes
R5 [7:0]	
R6 [7:0]	
R7 [7:0]	

The Interrupt Mask (I) bit of the PSR is used to select the second bank of registers which are used only during interrupt processing. This allows interrupts to be serviced without having to save and restore all of the registers, saving several microseconds for each interrupt.

Additional banks of registers can be added to allow rapid changes in context or for any purpose where the current state of the CPU needs to be retained. Typically the bank select bits are sourced from the FLAG bits in the PSR.

Address Generation Logic (V8_ADDR)

The address generation logic module consists of the 16 bit program counter, the stack pointer and an 8 bit temporary register. The output of the V8_ALU block is actually the address for the next clock (ADDR_D). The ADDR_D bus is then registered external to the V8_ALU block so that the address can come directly from a Flip-Flop.

The performance limiting paths in a V8-uRISC microcontroller design are typically from the ADDR bus, through address decodes to external IO, RAM or ROM and then into the DATAIN bus of the V8-uRISC. Providing the ADDR bus directly from Flip-Flops saves time in this critical path. Critical address decodes can also be done a cycle early by decoding the ADDR_D bus and then registering the address decode signal.

The 16 bit Program Counter (PC) provides the address from which opcodes are fetched. The counter is incremented whenever an opcode is fetched or the data bytes of an opcode are read. The counter is reloaded whenever a Jump or Branch opcode is executed or when processing an interrupt.

The stack pointer (SP) is a 16 bit up/down counter. It provides the address when data is pushed or popped from the stack. The SP points to the byte that was most recently pushed onto the stack. Typically the upper 8 bits of the SP are set to 0x01, but they can easily be set to any desired location in RAM. With the source-code version of the core, the counter can be reduced to less than 16 bits by simply changing a few constants in the HDL code to reduce the gate count of the V8-uRISC CPU.

DATAOUT Multiplexor

The Data Out Mux is an 8-bit 3 to 1 mux. It routes data from the register file or the high and low halves of the PC. The PC is required to be output when executing an interrupt or jumping to a subroutine.

Core Modifications

An optional HDL source license to the V8-uRISC allows the user to make unlimited modifications to the core. The core has been specifically designed to allow the implementation of two user defined opcodes.

Analysis of your application code can typically identify one or two functions that require most of the CPU bandwidth. By coding these functions as a single opcode, performance can be significantly increased without increasing the clock rate. This in turn can keep the power requirements of your product low.

Applications for custom opcodes include: 512 or 1024 bit math for high speed Public Key Encryption algorithms, multiply/accumulate for DSP applications, table lookups for decoding, block moves, and trigonometric functions commonly used in GPS applications.

Other modifications include adding register files, or reducing the size of the program counter. Consult VAutomation for any of these modifications.

Pinout

Signal names for the core are shown in Figure 1, and described in Table 3.

Verification Methods

VAutomation prototypes and verifies all of its cores using Xilinx devices. The V8-uRISC core has been implemented within the Xilinx-based IntelliCore™ Prototyping System (IPS) and extensively tested in real-world applications (e.g. command processing for USB audio). The IPS is available from VAutomation in support of a V8-uRISC evaluation and early software testing while hardware design is underway. Please consult the IntelliCore™ Prototyping System datasheet for more details.

The V8-uRISC has also been tested in simulation using a test bench which implements the core as a XC4013-based microcontroller. This microcontroller is identical to one implemented and tested within the IPS. The test bench also consists of a self-checking assembly program (Intel hex format object file) that is meant to be run on the test bench. The test bench is supplied with the core.

Table 3: Core Signal Pinout

Signal	Signal Direction	Description
ADDR[15:0]	Output	Address Bus
DATAIN[7:0]	Input	Data Input Bus
DATAOUT[7:0]	Output	Data Output Bus
READ	Output	Read Cycle - High when data is being loaded from the DATAIN bus.
WRITE	Output	Write Cycle - High when data is valid on the DATAOUT bus.
CLK	Input	Clock - All Flip-Flops clock on the rising edge of the clock; uses 1 FPGA CLKIOB pin.
RESET	Input	Reset - Must be asserted for at least 3 rising edges of CLK.
INT[7:0]	Input	Interrupt Request - INT[0] is Non-maskable and is generally reserved for JTAG debugging purposes.
READY	Input	Ready - Freezes the current state when low, operation completes when high. Must be high when RESET is asserted.
OP_FETCH	Output	Opcode Fetch Cycle. Typically used for single step debugging.
CLR_P[7:4]	Input	Clear PSR bit
SET_P[7:4]	Input	SET PSR bit
PSR	Output	Program Status Register

Recommended Design Experience

Users should be familiar with Xilinx design flows and tools, have experience with microprocessor systems and be skilled in assembly and/or C language programming.

Available Support Products

The V8-uRISC is supplied with an assembler (DOS application). An Instruction Level Simulator and debugger are available now. The debugger operates seamlessly with the Instruction Level Simulator or the IPS. A C Compiler will be available in May '98. The V8-uRISC is also supported by the IntelliCore™ Prototyping Systems (IPS) which is available separately. Please consult the IPS datasheet for more details.

Ordering Information

The V8-uRISC core is provided under license from VAutomation for use in Xilinx devices in either netlist or HDL source form. Please contact VAutomation at the location listed on the first page of this datasheet for ordering details.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

VAutomation, Inc.

20 Trafalgar Square
Suite 443 (4th Floor)
Nashua, NH 03063 USA
Phone: +1 603-882-2282
Fax: +1 603-882-1587
E-mail: sales@vautomation.com
URL: www.vautomation.com

Features

- Customizable modular design supports 13,000 to 260,000 system-gates
  - Up to four Xilinx XC4036EX-HQ240 FPGAs
  - Serial down load or PROM configuration, switch selectable on a per-FPGA basis
- Supports VAutomation microprocessor, USB Serial Interface Engine or FireWire cores
- Software
  - V8-uRISC assembler
  - V8-uRISC Monitor/debugger
  - JTAG master debugger
  - Windows98 USB audio class code for generic speaker application
- 128K x 16 SRAM
- Two 8K x 8 serial EEPROMs
- 2M x 8 DRAM
- Audio Codec (AD1845) and analog amplifiers for digital sound playback or recording
- External Interfaces
  - USB Serial Port with 1 uplink "A" and 3 downlink "B" connectors with transceivers and power
  - FETs
  - IEEE-1394 "FireWire" Serial Port
  - 100mbs three port PHY via TI TSB11C01
  - RS-232 Serial Port
  - IEEE-1284 Parallel Port
  - Analog Audio (MIC, Line in/out, Headphone)
  - Ethernet interface (MII connector)
  - IEEE-1149 JTAG controller port
  - IrDA Serial Port General Purpose Custom interfaces (via 40-pin MII connector)
  - 192 board-to-board interconnects for debugging with a logic analyzer

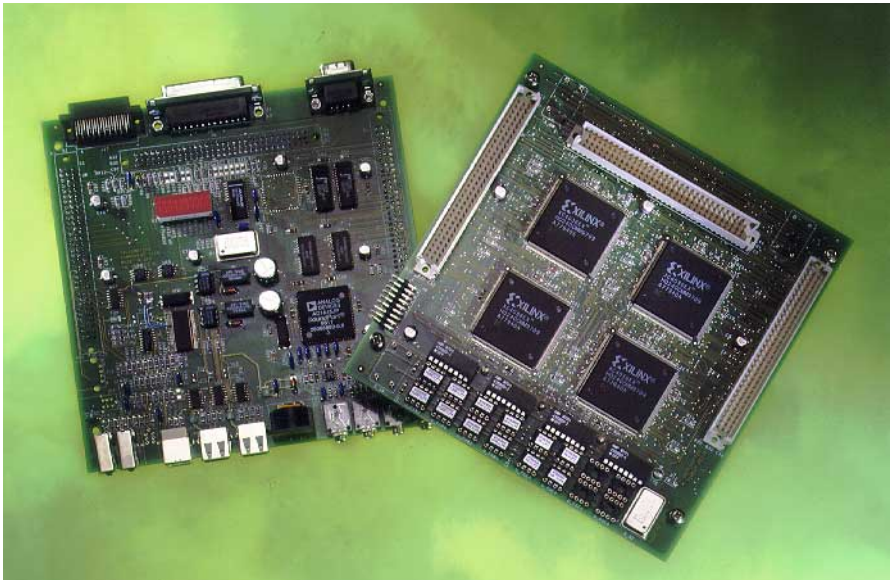


Figure 1: IntelliCore™ Prototyping System

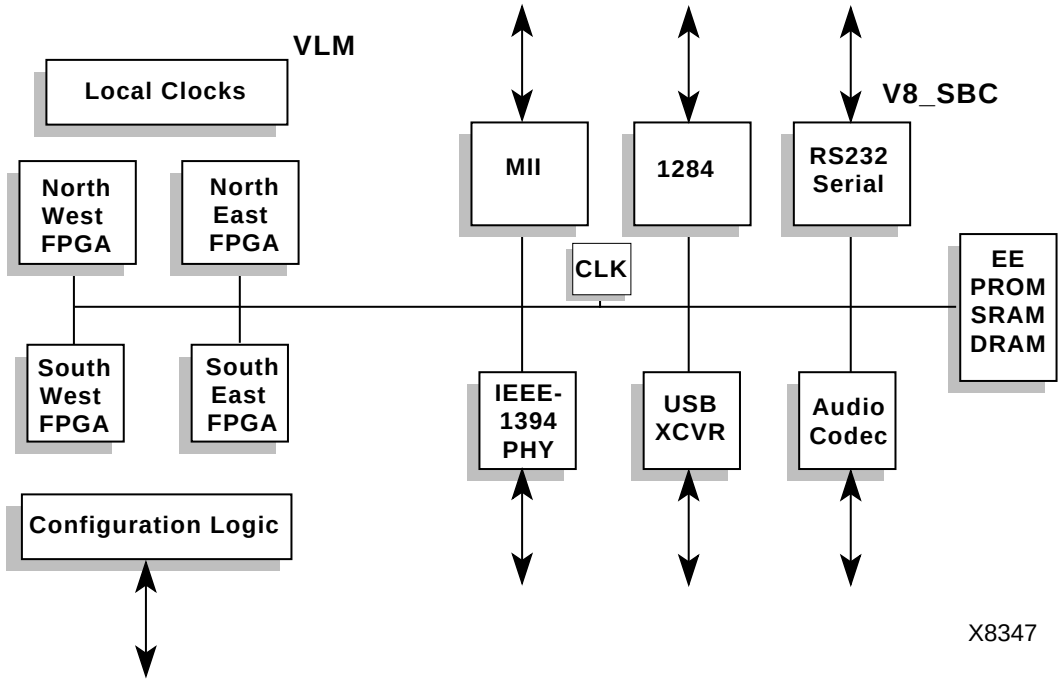


Figure 2: IntelliCore™ Prototyping System Block Diagram

- Universal Power Supply
  - Input: 95-250V, 47-63Hz
  - Output: 27 Watts DC, +5V @ 4A, +12V @ 0.5A
- UL, CE, CSA certified

General Description

The VAutomation IntelliCore™ Prototyping System (IPS) is an ideal platform for evaluating VAutomation's microprocessor and serial interface engine cores using Xilinx FPGAs. VAutomation uses the IPS for in-house development of the V8, VUSB and other cores. The IPS is an integral part of any V8-uRISC or USB development effort. The IPS enables:

- V8-uRISC/VUSB Evaluation
- Rapid software testing and integration
- JTAG debugging capabilities
- Rapid hardware design verification via FPGA emulation of your design

V8-uRISC/VUSB Evaluation

The IPS comes preconfigured as a Universal Serial Bus (USB) audio speaker application. Just plug it into any Windows98 PC and you'll hear USB in action with no drivers required! The IPS audio application is plug-and-play.

Hardware/Software Co-Design

The IPS enables software coding and integration to begin before the hardware design is completed. The FPGA hardware phase is no longer in the critical path and is overlapped with the software development.

JTAG Debugger

The IPS functions as a JTAG master controller and works in concert with VAutomation's V8-uRISC debugger. The FPGA can be debugged with only the four JTAG pins required for complete debug support.

Generic Hardware Emulation System

There is no need to simulate for weeks, no difficult system level models to code, and no test vectors to write when you can verify your design in a real world environment. The IPS can be used as an emulation system for design, debug and early software integration of your FPGAs. Utilize the rapid design turnarounds enabled by the Xilinx SRAM-based FPGAs to cut you hardware simulations down to size. Integrate software while the hardware is still being debugged.

Available Support Products

VAutomation supplies a complete line of cores that can be integrated and tested on the IPS. Contact VAutomation for more information. These cores include:

- V8-uRISC 8-bit microcontroller
  - UART and Baud Rate generator
  - 16-bit Timer/counter

  - IEEE 1284 parallel port
  - JTAG master/slave controller
  - DMA controller
  - DRAM controller
- VUSB USB target Serial Interface Engine

Ordering Information

The VAutomation IntelliCore Prototyping System lists for $995.00 USD (limit two per customer) and comes complete with the following deliverables:

- IPS electronics package
  - VLM Logic Module PCB with one Xilinx XC4036HQ240-3 FPGA
    - V8/VUSB requires all of the resources of the FPGA
    - Two and Four FPGA configurations also available for application specific logic
  - V8_SBC PCB
  - Pretested and Verified
- V8 assembler.EXE file for DOS/Windows
- V8 and VUSB Xilinx Configuration PROMs
- V8 Monitor/Debugger burned into the EEPROM and on a DOS format floppy
- Windows98 compatible USB Audio Class code for generic speakers
- Universal Power Supply
- IPS Users Manual
- V8-uRISC Users Manual
- VUSB Users Manual

To purchase or make further inquiries about this or other VAutomation products, contact VAutomation directly.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

M16450 Universal Asynchronous Receiver/Transmitter

January 12, 1998 Product Specification

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408-733-3344
Fax: +1 408-733-9922
E-mail: sales@virtualipgroup.com
URL: www.virtualipgroup.com

Features

- Functionally compatible to NS16450
- Complete asynchronous communication protocol including:
  - 5, 6, 7 or 8 bit data transmission
  - Even/odd or no parity bit generation and detection
  - Start and stop bit generation and detection
  - Line break detection and generation
  - Receiver overrun and framing errors detection
  - Communications rates of upto 56K baud
- Internal programmable baud rate generator
- Buffered transmit and receive registers
- Exception handling using interrupt/polled modes
- Internal diagnostic capabilities with loopback
- Modem handshake capability using CTS, RTS, DSR, DTR, RI and DCD signals
- Complete status reporting capabilities
- Line break generation and detection

Applications

- Serial Communication Port
- Modem Interface port

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000E
CLBs Used	188	188
IOBs Used	34[1]	34[1]
CLKIOBs Used	4	4
System Clock fmax	18.4 MHz	18.4 MHz
Device Features Used	Global Buffers	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XCS40PQ240-3	155[1]	596
XC4020EHQ240-2	155[1]	596
Provided with Core		
Documentation	Core Design Document Designers application note	
Design File Formats	.ngd, XNF netlist Verilog Source RTL available extra	
Constraint Files	.cst file, xactinit.dat.	
Verification Tool	Test Vectors	
Schematic Symbols	None	
Evaluation Model	None	
Reference designs & application notes	FPGA Design Document	
Additional Items	Evaluation board available extra	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
Entry/Verification Tool	Verilog RTL/Verilog XL simulator	
Support		
Support provided by Virtual IP Group, Inc.		

Note:
1. Assuming all core signals are routed off chip.

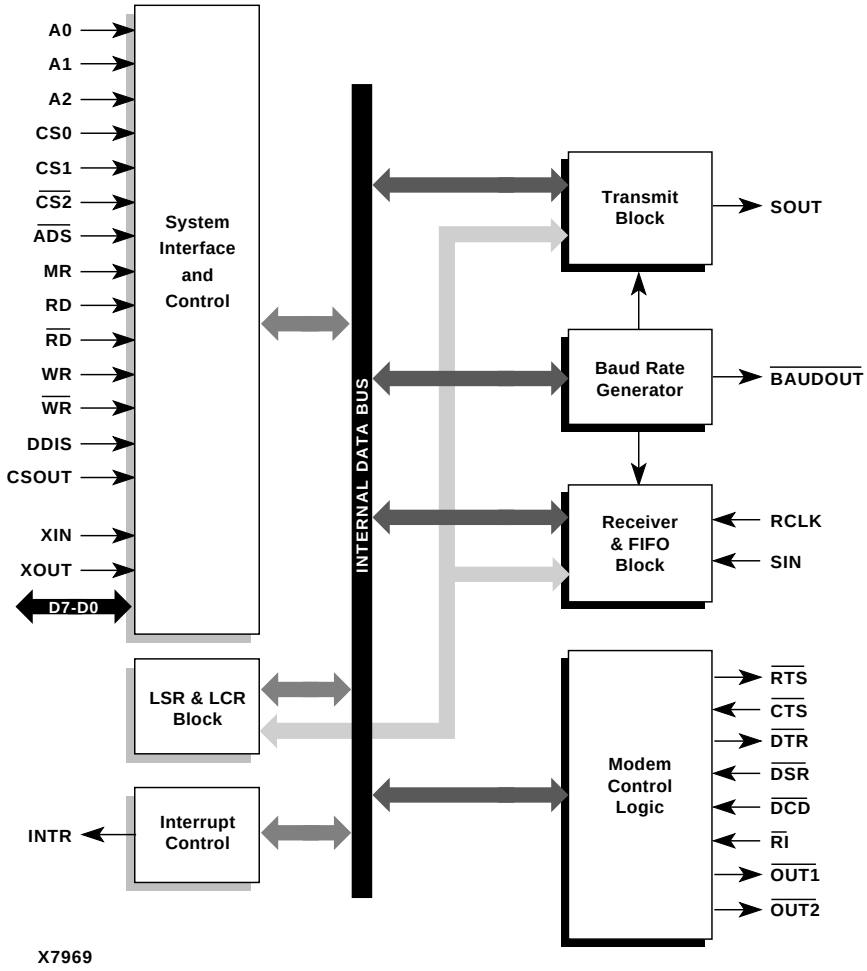


Figure 1: M16450 Block Diagram

General Description

M16C450 interfaces with a microcontroller or microprocessor on one side and serial communications equipment on the other. It provides full modem control through input and output signals for easy handshaking with modems during communication. Internal registers provide full programmability of serial asynchronous communication parameters. This core is functionally compatible to the National Semiconductor NS16450.

Functional Description

The functional Block Diagram is shown in Figure 1. The internal modules are explained below.

System Interface and Control Block

This block supports the Processor interface and generates the internal system level signals for proper functioning.

LSR and LCR Block

This block holds the Line Status and Line Control Registers. These two registers control communication for the core.

Interrupt Control Block

This block handles all interrupt capabilities for the core.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
System Interface Signals		
A0, A1, A2	Input	Address signals to select an internal register for read/ write operations.
CS0-2	Input	Chip Select. CS0 and CS1 are active high, CS2 is active low.
ADS	Input	Address Strobe. Chip Select and address signals are latched internally on rising edge of $\overline{ADS}$. Pulled low if unused.
MR	Input	Master Reset, active high.
RD	Input	Read Control, active high. Pulled low if unused.
$\overline{RD}$	Input	Read Control, active low. Pulled high if unused.
WR	Input	Write Control, active high. Pulled low if unused.
$\overline{WR}$	Input	Write Control, active low. Pulled high if unused.
DDIS	Output	**D**river **DIS**able signal, driven low, when core outputs data. Used to control data flow direction in transceiver or, tristate buffers enable control.
CSOUT	Output	Indicates read/write selection of UART. Active high and remains high when UART is selected through chip select inputs.
XIN	Input	Master clock input.
XOUT	Input	Master clock output, inverted from XIN.
D7 - D0	In/Out	Bidirectional databus carries data to be written to internal registers; also reports status of registers during read cycle.
Modem Interface Signals		
$\overline{RTS}$	Output	Active low **R**EQUEST **T**O **S**END indicates UART is ready to exchange data. System controls this pin bit in modem control register.
$\overline{CTS}$	Input	Active low **C**LEAR **T**O **S**END indicates modem is ready to exchange data. Present state monitored by reading MSR.
$\overline{DTR}$	Output	Active low **D**ATA **T**ERMINAL **R**EADY tells modem that UART is ready to establish communication link. System controls this pin through bit in MSR.r.
$\overline{DSR}$	Input	Active low **D**ATA **S**ET **R**EADY indicates modem is ready to handshake with core. Present state monitored by reading MSR.
$\overline{DCD}$	Input	Active low **D**ATA **C**ARRIER **D**ETECT indicates modem has detected carrier on communications line. Present state monitored by reading MSR.
$\overline{RI}$	Input	Active low **R**ING **I**NDICATOR indicates modem has detected ring signal. Present state monitored by reading MSR.
OUT1-2	Output	General purpose outputs. System controls these pins by through bit in MSR.
Transmit/Receive Signals		
SOUT	Output	Serial data output from transmitter block.
RCLK	Input	Input receive clock, should be 16 times communications baud rate.
SIN	Input	Serial data input for receiver block.
Other Signals		
INTR	Output	Active high interrupt signal.
BAUDOUT	Output	Baud rate generator output clock. 16 times programmed communication baud rate.

Transmit Block

This block controls serial data transmission as per programmed parameters.

Baud Rate Generator Block

This block generates the Baud Rate Clock for the transmitter section of the core. This clock can also be used by the receiver block connecting $\overline{\text{BAUDOUT}}$ to RCLK.

Receiver Block

This block handles reception for the core. The clock for this block is provided by RCLK. This clock should be 16 times the baud rate.

Modem Control Logic Block

This block handles modem handshaking for the core. These signals can be used for communication or as general purpose signals. The modem control register resides in this block, providing internal diagnostic capability.

Core Modifications

Modifications can be done to remove the internal baud rate generator, or to strip off either transmitter or receiver. These modifications can be performed by Virtual IP Group, Inc. for additional cost.

Pinout

The pinout has not been fixed to a specific FPGA/IO allowing flexibility with the user application. A pinout is suggested for use with a user-constructible evaluation board. Information for this is in the FPGA Design Document included with the core. Signal names are provided in the block diagram shown in Figure 1 and Table 1.

Verification Methods

The core has been tested with in-house developed simulation test vectors that are provided with the core. Assembly level 80x86 programs were used to test the functionality of the FPGA using a hardware evaluation board.

Recommended Design Experience

Knowledge of interface with Microprocessor based systems is required. The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment. Usage of Alliance or Foundation tools is required.

Available Support Products

The FPGA Design Document included with the core gives directions of constructing a general purpose FPGA evaluation daughter board that can be plugged in to a standard port socket on the target system through a flat cable.

Ordering Information

This product is available from Virtual IP Group, Inc. Please contact them for pricing and additional information.

Related Information

Refer to Specification Document for programming of this core for a typical application in a system. The user is required to refer to Designer's application note for integrating this core with other cores.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

M16550A - Universal Asynchronous Receiver/Transmitter With FIFOs

January 12, 1998 | Product Specification

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408-733-3344
Fax: +1 408-733-9922
E-mail: sales@virtualipgroup.com
URL : www.virtualipgroup.com

Features

- Complete asynchronous communication protocol includes:
  - 5, 6, 7 or 8 bit data transmission
  - Even/odd or no parity bit generation and detection.
  - Start and stop bit generation and detection.
  - Line break detection and generation.
  - Receiver overrun and framing errors detection
- Communications rates of up to 56K baud
- Internal programmable baud rate generator
- Buffered transmit and receive registers
- Exception handling using interrupt/polled modes
- Two Modes of operation - NS16450 and FIFO mode
- Transmitter is buffered with 16 Byte FIFO
- Receiver is buffered with 16 Byte FIFO plus 3 error bits per data byte
- Internal diagnostic capabilities with loopback
- Modem handshake capability using CTS, RTS, DSR, DTR, RI and DCD signals
- Complete status reporting Capabilities
- Line break generation and detection

Applications

- Serial Communication Port
- Modem Interface Port

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000E
CLBs Used	359	359
IOBs Used	38[1]	38[1]
System Clock fmax	16 MHz	16 MHz
Device Features Used	Global Buffers	

Supported Devices/Resources Remaining		
	I/O	**CLBs**
XCS40 PQ240-3	155	425
XC4020E HQ240-2	155	425

Provided with Core	
Documentation	Core Design Document FPGA Design Document
Design File Formats	NGD or XNF netlist Verilog Source RTL (available extra)
Constraint Files	.cst file, xactinit.dat.
Verification Tool	Test Vectors
Schematic Symbols	None
Evaluation Model	None
Reference designs & application notes	FPGA Design Document
Additional Items	Evaluation Board available extra

Design Tool Requirements	
Xilinx Core Tools	Alliance 1.3
Entry/Verification Tool	Verilog RTL/Verilog XL simulator

Support
Support provided by Virtual IP Group, Inc.

Note:
1. Assuming all core signals are routed off-chip.

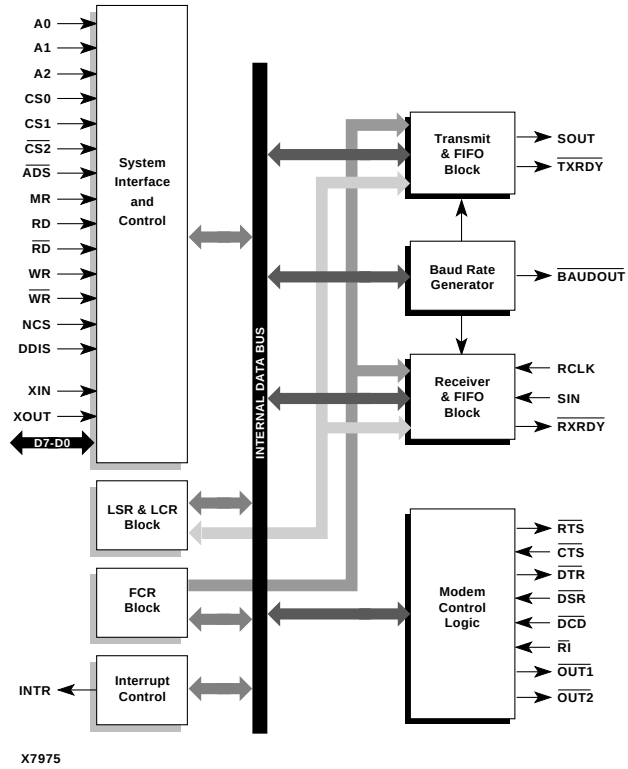


Figure 1: M16550A Block Diagram

General Description

The M16550A interfaces with a microcontroller or microprocessor on one side and serial communications equipment on the other. It provides full modem control through easy handshaking with modems during communication. Internal registers provide full programmability of serial asynchronous communication parameters. It also implements exception detection and reporting to the processor using interrupt or polled modes. The communication line and modem status can be monitored at any time by the processor by reading appropriate registers in the core.

Functional Description

This core emulates the functionality of National Semiconductor's NS16550A. The block diagram with internal structure is shown in Figure 1.

System Interface and Control Block

This block supports the processor interface and generates the internal system level signals for proper functioning.

LSR and LCR Block

This block holds the Line Status and Line Control Registers. These two registers control serial communication capabilities of the core.

FCR Block

This block has the FIFO control register which controls the parameters for the transmit and receive FIFOs.

Interrupt Control Block

This block handles all interrupt capabilities for the core.

Transmit and FIFO Block

This block holds the transmitter section and a 16-byte transmit FIFO.

Baud Rate Generator Block

This block generates the Baud Rate Clock for the transmitter section of the core. This clock can also be used by the receiver block by connecting $\overline{\text{BAUDOUT}}$ to RCLK

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
System Interface Signals		
A0, A1, A2	Input	Address signals to select an internal register for read/ write operations.
CS0-2	Input	Chip Select. CS0 and CS1 are active high, CS2 is active low.
ADS	Input	Address Strobe. Chip Select, and address signals are latched internally on rising edge of $\overline{ADS}$. Pulled low if unused.
MR	Input	Master Reset Signal, active high.
RD	Input	Control, active high. Pulled low if unused.
$\overline{RD}$	Input	Control, active low. Pulled high if unused.
WR	Input	Control, active high. Pulled low if unused.
$\overline{WR}$	Input	Write Control, active low. Pulled high if unused.
DDIS	Output	**D**river **DIS**able signal, driven low, when core outputs data. Used to control data flow direction intransceiver, or tristate buffers enable control.
CSOUT	Output	Indicates read/write selection of UART, active high and remains high when UART is selected through chip select inputs.
XIN	Input	Master clock input.
XOUT	Input	Master clock output, inverted from XIN.
D7 - D0	In/Out	Bidirectional databus carries data to be written to internal registers; also reports status of registers during read cycle.
Modem Interface Signals		
$\overline{RTS}$	Output	Active low **R**EQUEST **T**O **S**END indicates UART is ready to exchange data. System controls this pin through bit in modem control register.
$\overline{CTS}$	Input	Active low **C**LEAR **T**O **S**END indicates modem is ready to exchange data. Present state monitored by reading MSR.
$\overline{DTR}$	Output	Active low **D**ATA **T**ERMINAL **R**EADY tells modem that UART is ready to establish communication link. System controls this pin by programming a bit in MSR.
$\overline{DSR}$	Input	Active low **D**ATA **S**ET **R**EADY indicates modem is ready to handshake with core. Present state monitored by reading MSR.
$\overline{DCD}$	Input	Active low **D**ATA **C**ARRIER **D**ETECT indicates modem has detected carrier on communications line. Present state monitored by reading MSR.
$\overline{RI}$	Input	Active low **R**ING **I**NDICATOR indicates modem has detected ring signal. Present state monitored by reading MSR.
OUT1-2	Output	General purpose outputs. System controls these pins by through bit in MSR.
Transmit/Receive Signals		
SOUT	Output	Serial data output from transmitter block.
RCLK	Input	Input receive clock, should be 16 times communications baud rate.
SIN	Input	Serial data input for receiver block.
Other Signals		
INTR	Output	Active high interrupt signal.
BAUDOUT	Output	Baud rate generator output clock.16 times the programmed communication baud rate.

Receiver and FIFO Block

This block handles reception for the core. The clock for this block is provided by RCLK. This clock should be 16X the Baud Rate. The receive data FIFO is included in this block which stores 16 bytes at a time. The 16 bit FIFO's for status parameters (parity, framing and break) for the corresponding bytes in the receiver data FIFO is also included in this block.

Modem Control Logic Block

This block handles the modem control capabilities for the megacell core. These signals can be used for the communication purpose also some of the signals can be used as general purpose signals ($\overline{OUT1}$ and $\overline{OUT2}$). The modem control register resides in this block which provides the internal diagnostic capability for the core.

Core Modifications

Modifications can be done to remove the internal baud rate generator, or strip off either transmitter or receiver or size the FIFO of Transmitter and Receiver separately. These modifications can be performed by Virtual IP Group, Inc. for additional cost.

Pinout

The pinout has not been fixed to specific FPGA I/O allowing flexibility with the users application. The evaluation board pinout, however has been fixed. This information is included in the design documentation. Signal names are provided in the block diagram shown in Figure 1 and Table 1.

Verification Methods

The core has been tested with in-house developed simulation test vectors that are provided with the core. Assembly level 80x86 programs were used to test the functionality of the FPGA in hardware.

Recommended Design Experience

Knowledge of interface with Microprocessor based systems is required. The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment. Usage of Alliance or Foundation tools is required.

Available Support Products

The FPGA Design Document included with the core gives directions of constructing a general purpose FPGA evaluation daughter board that can be plugged in to a standard port socket on the target system through a flat cable.

Ordering Information

This product is available from Virtual IP Group, Inc. Please contact them for pricing and additional information.

Related Information

Refer to Specification Document for programming of this core for a typical application in a system. The user is required to refer to Designer's application note for integrating this core with other cores.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

January 12, 1998 Product Specification

CAST, Inc.

24 White Birch Drive
Pomona, New York 10907 USA
Phone: +1 914-354-4945
Fax: +1 914-354-0325
E-Mail: info@cast-inc.com
URL : www.cast-inc.com

Features

- Based on the AMD2910A device
- Pre-defined implementation for predictable timing
- Verified against a hardware model of the original device
- 12-bit data width that addresses up to 4,096 words
- Internal loop counter - pre-settable 12-bit down-counter for repeating instructions and counting loop interactions
- Four address sources
  - Microprogram counter
  - Branch address bus
  - 9-level push/pop stack
  - Internal holding register
- 16 powerful microinstructions
- Output Enable Controls for three branch address sources
- Positive-edge-triggered registers

Applications

The C2910A core is used in high speed bit slice designs.

General Description

The C2910A microprogram controller core is an address sequencer that controls the sequence of execution for the microinstructions stored in microprogram memory. The core can sequentially access the microinstructions, and it provides conditional branching to any microinstructions within the 4,096-microword range. In addition, a nine-deep LIFO stack provides a microsubroutine return linkage and looping capability.

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4005XL
CLBs Used	185	185
IOBs Used	38[1]	38[1]
System Clock f_{max}	27.7 MHz	34.5 MHz
Device Features Used	Global Buffers	

Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC4005XL-09	75[1]	11
XCS10-3	75[1]	11

Provided with Core	
Documentation	Core Documentation
Design File Formats	XNF Netlist VHDL Source RTL
Constraint Files	NCF
Verification Tool	Test Bench Test Vectors
Schematic Symbols	Viewlogic
Evaluation Model	VHDL behavioral model
Reference designs & application notes	None
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	M1.3
Entry/Verification Tool	1076 compliant VHDL Simulator

Support
Support provided by CAST, Inc.

Notes:
1. Assuming all core signals are routed off-chip.

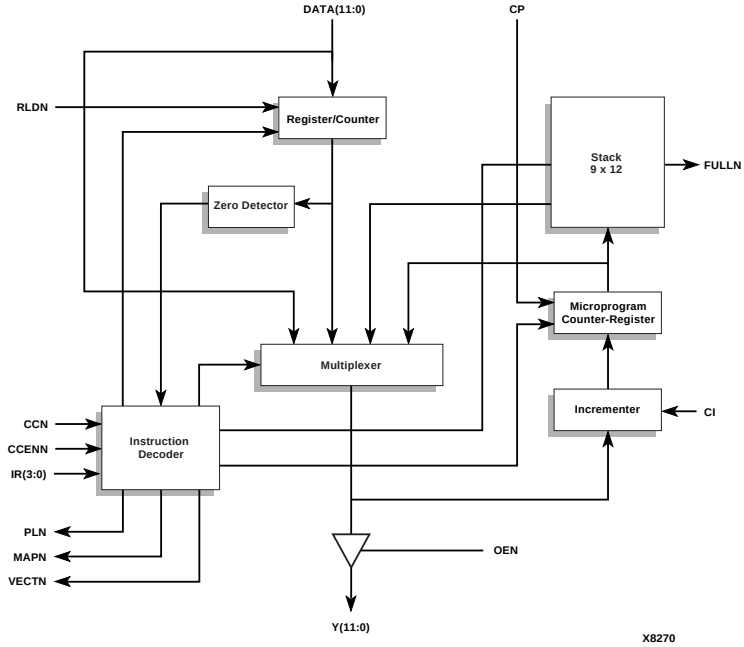


Figure 1: C2910A Microprogram Controller Block Diagram

Functional Description

The C2910A core is partitioned into modules as shown in Figure 1 and described below. Xilinx netlists are provided for each module.

Multiplexer

The four-input multiplexer is used to select either the register/counter, direct input, microprogram counter, or stack as the source of the next microinstruction address.

Register Counter

This block consists of 12 D-type, edge-triggered flip-flops, with a common enable. When its load control, RLDN is low, new data is loaded on a positive clock transition. The output of the register/counter is available to the multiplexer as a source for the next microinstruction address. The direct input furnishes a source of data for loading the register/counter.

Microcontroller Counter/Register (µPC)

This block consists of a 12-bit incrementer followed by a 12-bit register. The mPC can be used in either of two ways: When the carry-in to the incrementer is high, the microprogram register is loaded on the next clock cycle with the current Y output word plus one (Y + 1 → µPC). Sequential microinstructions are thus executed . When the carry-in is low, the incrementer passes the Y output word unmodified so that mPC is reloaded with the same Y word on the next clock cycle (Y → µPC). The same microinstruction is thus executed any number of times.

Stack

This 9-word by 12-bit stack is used to provide return address linkage when executing microsubroutines or loops. The stack contains a built-in stack pointer which always points to the last word written. This allows stack reference operations (looping) to be performed without a pop.

Instruction Decoder

This block decodes the incoming instruction and generates the appropriates control signals for all the other blocks. The instruction decoder block also generates the outputs PLN, MAPN, and VECTN.

Core Modifications

The C2910A core can easily be customized to include:

- Larger data width
- Different stack depth

Please contact CAST directly for any required modifications.

Pinout

The pinout of the C2910A core has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are shown in the block diagram in Figure 1, and in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
CP	Input	Clock
IR (3:0)	Input	Instruction
DATA (11:0)	Input	Direct Data
CCN	Input	Condition Code
CCENN	Input	Condition Code Enable
RLDN	Input	Register Load Enable
CI	Input	Carry-In
OEN	Input	Output Enable
Y (11:0)	Output	Microprogram Address
FULLN	Output	Full Flag
PLN	Output	Pipeline Address Enable
MAPN	Output	Map Address Enable
VECTN	Output	Vector Address Enable

Core Assumptions

Stack PUSH

After a depth of 9 is reached, the stack is full and the FULLN output goes LOW. If further PUSHes are attempted when the stack is full, the stack information at the top of the stack (9) will be over-written.

It has been observed that the AM2910A device inconsistently destroys the top 2 locations of the stack (8 and 9). This functionality is apparently not documented, and should not be relied upon.

Stack POP

Further POPs from an empty stack will place the bottom data on the Y-Bus (no changes). Is has been observed that the AM2910A device will place random data in the Y-Bus during POPs from an empty stack.

To achieve proper result, do not exercise the stack beyond its range (i.e. no PUSHes while the stack is full and no POPs while stack is empty).

Verification Methods

The C2910A Microprogram Controller core's functionality was verified by means of a proprietary hardware modeler. The same stimulus was applied to a hardware model which contained the original AM2910A chip, and the results compared with the core's simulation outputs.The C2910A core has also been successfully implemented into silicon.

Recommended Design Experience

The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment.

Ordering Information

This product is available from CAST, Inc. Please contact CAST for pricing and more information.

Related Information

Bipolar Microprocessor Logic and Interface Data Book

Contact:

Advanced Micro Devices
One AMD Place
P.O. Box 3453
Sunnyvale, California 94088-3453
Phone: 408-732-2400
800-538-8450
800-222-9323 (literature)
E-mail: AMDlit@gomez.amd.com
URL: www.amd.com

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

M8237 DMA Controller

February 8, 1998 Product Specification

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408-733-3344
Fax: +1 408-733-9922
E-mail: sales@virtualipgroup.com
URL: www.virtualipgroup.com

Features

- Multimode Direct Memory Access (DMA) controller
- Functionally compatible to Intel 8237
- Four independent DMA channels
- Independent auto-initialization of all channels
- Directly expandable to any number of channels
- Memory-to-memory transfers
- Memory block initialization
- Enable/disable control of individual DMA requests
- Address increment/decrement selection control for all channels
- High performance transfers up to 1.6 MBytes/second with 5 MHz
- End of process input to terminate transfers
- Software DMA request
- Programmable polarity control for DMA Request and DACK signals

Applications

- Multi-mode Programmable, multi-channel DMA
- Support Controller for Microprocessor based systems

AllianceCORE™ Facts

Core Specifics	
Device Family	XC4000E
CLBs Used	392
IOBs Used	37[1]
CLKIOBS used	1
System Clock fmax	8 MHz
Device Features Used	Global buffers

Supported Devices/Resources Remaining		
	I/O	CLBs
XC4036EX HQ240-4	156[1]	904

Provided with Core	
Documentation	Core Design Document Designer's Application Note
Design File Formats	.ngd, XNF netlist Verilog Source RTL (available extra)
Constraint Files	.cst file, xactinit.dat
Verification Tool	Test Vectors
Schematic Symbols	None
Evaluation Model	None
Reference designs & application notes	FPGA Design Document included
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	Alliance 1.3
Entry/Verification Tool	Verilog RTL/Verilog XL Simulator

Support
Support provided by Virtual IP Group, Inc.

Note:
1. Assuming all core signals are routed off-chip.

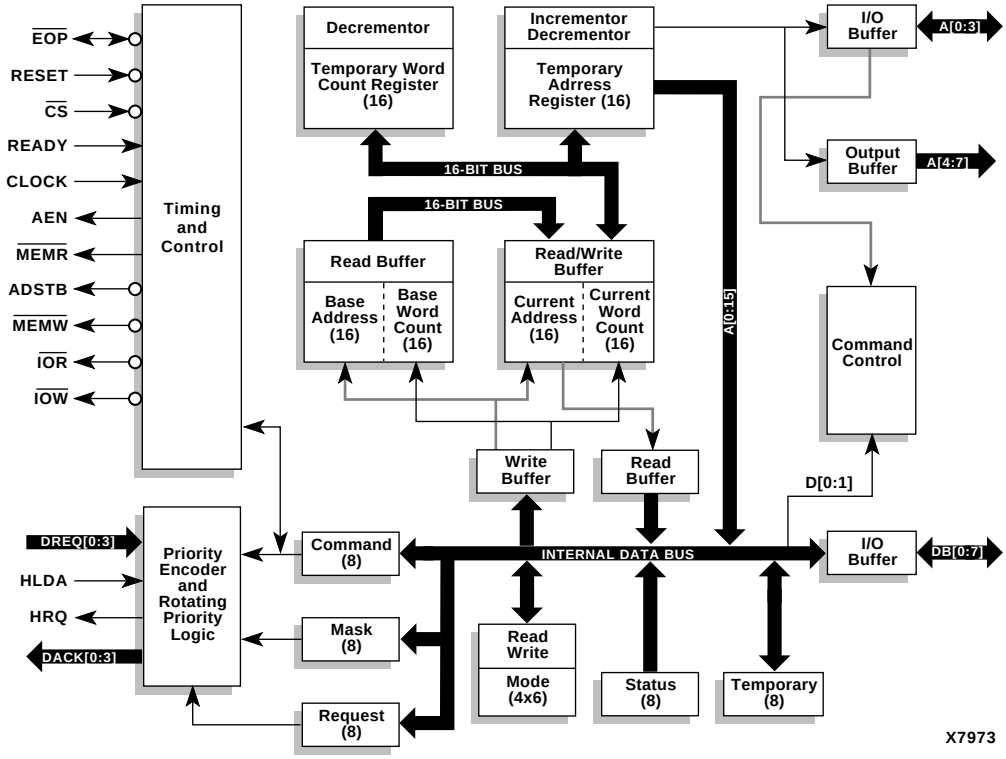


Figure 1: M8237 DMA Controller Block Diagram

General Description

The M8237 core is a 4 channel programmable DMA controller that can be interfaced with a standard microprocessor. All four DMA channel operations are independently programmable by the microprocessor during initialization. The core is functionally compatible with the standard 8237 chip.

Functional Description

The M8237 DMA Controller core is partitioned into modules as shown in Figure 1 and described below.

Timing Control Block

The Timing Control block generates internal timing and external control signals for the M8237. It also generates DMA bus cycle timing based on internal state cycles using the clock.

Program Command Control Block

The Program Command Control block decodes commands given to the M8237 by the microprocessor prior to servicing a DMA request. It also decodes the Mode Control word used to select the type of DMA service.

Priority Encoder Block

The Priority Encoder block resolves priority contention between DMA channels requesting service simultaneously.

Register Block

The M8237 contains 344 bits of internal memory in the form of registers. Table 2 lists these registers and shows size of each.The registers are independently addressable and are separate for each DMA channel except a few which are common.

Core Modifications

Multiple cores can be integrated to provide additional channels. Modifications can be done to reduce number of channels.These modifications are performed by Virtual IP Group, Inc. for additional cost.

Pinout

The pinout has not been fixed to specific FPGA I/O flexibility with the user application. A pinout is suggested for use with a user constructible evaluation board for this is in the FPGA Design Document included with the core. Signal names are

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Timing and Control Signals		
/EOP	Input	End Of Process input to stop DMA service; active low.
RESET	Input	Reset: clears command mode, status, request, temporary registers and first/last flip flop; sets mask register; active high.
/CS	Input	Chip Select to select M8237 as an I/O allowing CPU to communicate on the data bus; active low.
READY	Input	Ready, used to extend memory read and write pulses from M8237 to accommodate slower memories or I/O peripheral devices.
CLOCK	Input	Clock input.
AEN	Output	Address Enable signal used to disable other system bus drivers during DMA cycle; active high.
/MEMR	Output	Memory Read output from selected memory during a DMA read transfer or for memory to memory read cycle; active low tristate.
ADSTB	Output	Address Strobe, strobes up Upper address nibble; per address byte into an external latch during DMA cycle; active high.
/MEMW	Output	Memory Write output to write data on selected memory during DMA write transfer or for memory to memory write cycle; active low tristate.
/IOR	Input	I/O Read input control line used by CPU to read registers; active low.
/IOW	Input	I/O Write input control line used by CPU to write to registers; active low.
Priority Logic Signals		
DREQ[0:3]	Input	DMA Request [0:3], individual asynchronous channel request lines used by peripheral devices to obtain service. Programmable polarity; DREQ must be maintained until corresponding DACK line goes active.
HLDA	Input	Hold Acknowledge, tells M8237 that CPU has relinquished control of system bus, active high.
HRQ	Output	Hold Request given to CPU to request control of system bus.
DACK[0:3]	Output	DMA Acknowledge, notifies Address Enable signal used peripherals when DMA cycle is granted. Polarities are programmable.
Address/Data Signals		
A[0:3]	In/Out	Lower address nibble used by CPU to address registers for reading or writing; M8237 outputs lower 4 bits of output address during DMA service.
A[4:7]	Output	Upper address nibble; M8237 outputs upper 4 bits of output address during DMA service.
DB[0:7]	In/Out	Data bus used by CPU to write to and read from internal registers and to read memory to memory read cycle data. During DMA service 8 most significant address bits are output and strobed into an external latch by ADSTB.

Table 2: Internal Registers

Name	Size	Number
Base Address Registers	16 bits	4
Base Word Count Registers	16 bits	4
Current Address Registers	16 bits	4
Current Word Count Registers	16 bits	4
Temporary Address Register	16 bits	1
Temporary Word Count Register	16 bits	1
Status Register	8 bits	1
Command Register	8 bits	1
Temporary Register	8 bits	1
Mode Registers	6 bits	4
Mask Register	4 bits	1
Request Register	4 bits	1

provided in the block diagram shown in Figure 1 and described in Table 1.

Verification Methods

The core has been tested in house developed test vectors that are provided with the core.

Recommended Design Experience

Knowledge of DMA interfaces in a microprocessor based systems is required. The user must be familiar with HDL design methodology, instantiation of Xilinx netlists in a hierarchical design environment and usage of Xilinx development tools.

Available Support Products

FPGA Evaluation Board

The FPGA Design Document included with the core gives directions for constructing a general purpose FPGA evaluation daughter board that can be plugged into a standard part socket on the target system through a flat cable.

Ordering Information

This product is available from Virtual IP Group, Inc. Please contact them for pricing and additional information.

Related Information

Refer to Specification Document for programming of this core for a typical system application. The user should refer to the Designer's Application Note for integrating this with other cores. Both documents are included with the core.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

XF8250 Asynchronous Communications Core

January 12, 1998 Product Specification

Maria Aguilar, Project Coordinator
Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, AZ 85202
Phone: +1 888-360-9044 (USA)
+1 602-491-4311
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Features

- Software and function compatible with Industry Standard 8250
- Single macro UART and Baud Rate Generator
- DC to 625K baud (DC to 10 MHz Clock)
- External clock input In-macro Baud Rate Generator
- 1 to 65535 divisor generates 16X clock
- Prioritized interrupt mode
- Microprocessor bus oriented interface
- 80C86/80C88 compatible
- Modem interface Line break generation and detection
- Loopback and Echo modes

Applications

- Serial data communications applications
- Logic consolidation

General Description

The XF8250 Asynchronous Communications Core is a high-performance programmable Universal Asynchronous Receiver/Transmitter (UART) and Baud Rate Generator (BRG).

The Harris 82C50A datasheet, dated January 1992, is the target functional specification for the XF8250. The AC Specifications, both Timing Requirements and Timing, are met or exceeded in every case. A post-route timing analysis should be used to verify performance. Memec Design Services (MDS) can assist with this.

AllianceCORE™ Facts

Core Specifics			
Device Family	Spartan	XC4000E	XC5200
CLBs Used	97	97	64
IOBs Used	35[1]		
Systems Clock f_{max}	10+ MHz[2]		
Device Features Used	Tbufs, global clock buffers		

Supported Devices/Resources Remaining		
	I/O[1]	CLBs
XC4005E-6[3]	26	99
XC5202-6[3]	30	0
XCS05-3 PC84[3]	26	99

Provided with Core	
Documentation	Core schematics Implementation instructions
Design File Formats	ViewLogic Schematic files Live hierarchical tree LCA Files
Constraint Files	TimeSpecs
Verification Tool	Machine-readable simulation vectors for ViewLogic ViewSim
Schematic Symbols	ViewLogic
Evaluation Method	None
Reference designs & application notes	None
Additional Items	Warranty by MDS

Design Tool Requirements	
Xilinx Core Tools	Alliance/Foundation 1.3

Memec Design Services warrants that the design delivered by Memec Design Services will conform to the design specification. This warranty expires 3 months from the date of delivery of the design database. Contact Memec Design Services for the Design License Agreement with complete “Terms and Conditions of Sale.”

Notes:
1. Assuming all core signals are routed off-chip.
2. Minimum guaranteed speed to meet Harris specification.
3. Specific devices are minimum size and speed recommended - the core will work in larger or faster devices from the same families.

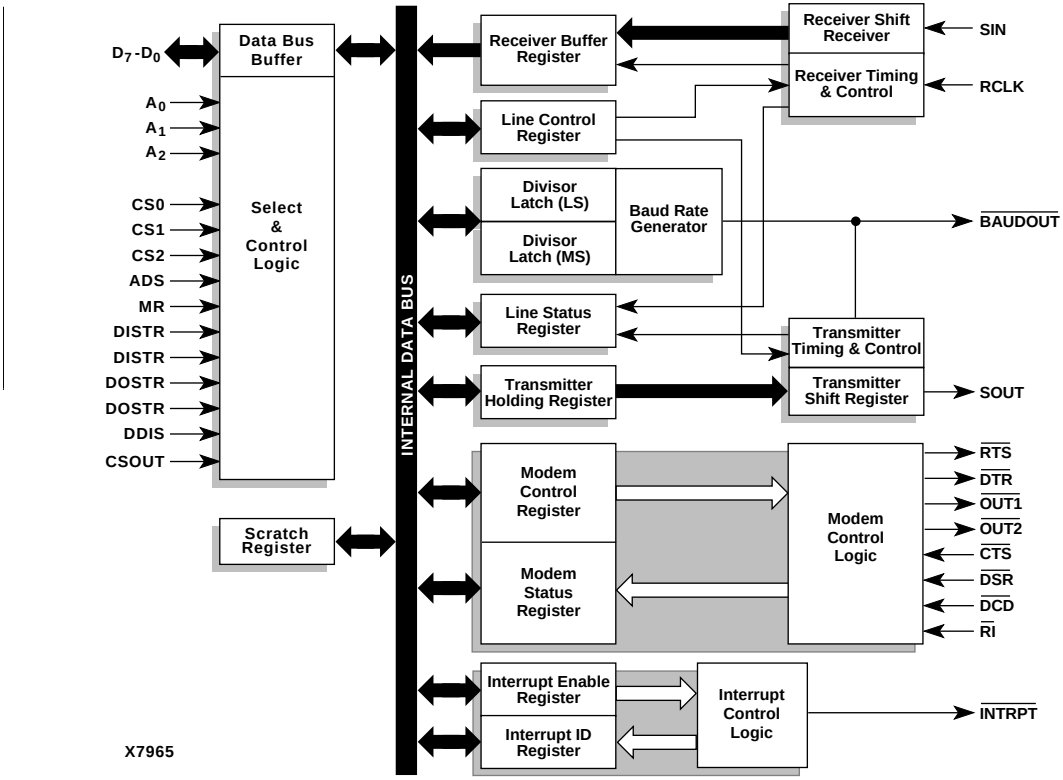


Figure 1: XF8250 Block Diagram

The Absolute Maximum Ratings, Operating Conditions, DC Electrical Specifications, and Capacitance are device dependent and can be found in the Xilinx datasheet for the target device.

Functional Description

The XF8250 is partitioned into modules as shown in Figure 1 and described below.

Select and Control Logic

This controls decodes for the core and the direction of the data bus buffer.

Receiver Timing & Control

This block detects the start-bit, controls the sampling of the asynchronous receive data (SIN), and determines when a complete word has been shifted into the receiver shift register.

Line Control Register (LCR)

This register controls the format of the data character. The contents of the LCR may be read, eliminating the need for separate storage of the line characteristics in system memory.

Line Status Register (LSR)

A single register that provides status indications. The LSR is usually the first register read by the CPU to determine the cause of an interrupt or to poll the status of the XF8250.

Modem Control Register

This controls the interface to the modem or data set. The MCR can be written to and read from.

Modem Status Register

This provides the CPU with the status of the modem input lines from the external device. The MSR allows the CPU to read the modem signal inputs by accessing the data bus interface of the XF8250.

Baud Rate Generator

This divides the clock by a 16-bit divisor to generate the 16x baud rate clock (BAUDOUT).

Receiver Shift Register

This register is programmable for 5, 6, 7, or 8 data-bits per character.

Transmitter Holding Register

This register holds parallel data from the data bus (D0-D7) until the Transmitter Shift Register is empty and ready to accept a new character for transmission.

Scratch Register

This register is an 8-bit Read/Write register which has no effect on the XF8250. It is intended as a scratch pad register to be used by the programmer to hold temporary data.

Interrupt ID Register

This register provides the XF8250 with interrupt capability for interfacing to microprocessors. In order to minimize software overhead during data character transfers, the XF8250 four levels of interrupt priorities:

- Priority 1 - Receiver Line Status
- Priority 2 - Received Data Ready
- Priority 3 - Transmitter Holding Register Empty
- Priority 4 - Modem Status

Interrupt Enable Register

This is a Write register used to independently enable the four XF8250 interrupts which activate the interrupt (INTRPT) output.

Core Modifications

The XF8250 meets or exceeds the AC Specifications of the Harris 82C50A-5. However, in most cases the TimeSpecs can be tightened significantly. Successful operation with 120 ns bus cycles has been achieved. In all cases, a post-route timing analysis should be performed to verify performance. Implementation beyond 10 MHz and other customizing is available through Memec Design Services.

Pinout

The XF8250 may be implemented as stand-alone logic using the provided pinout or may be implemented internally with the user's design. Both versions are included with the Core deliverables. Signal names are provided in the block diagram shown in Figure 1, and Table 1.

Core Assumptions

Deviations from the Harris 82C50 functional specification are described below. Memec Design Services will perform any of the listed modifications upon request.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D7-D0	In/Out	Data bits 7-0
A0	Input	Register Reset 0
A1	Input	Register Reset 1
A2	Input	Register Reset 2
CS0	Input	Chip Select 0
CS1	Input	Chip Select 1
/CS2	Input	Chip Select 2
/ADS	Input	Address Strobe
MR	Input	Master Reset
DISTR	Input	Data In Strobe
/DISTR	Input	Data In Strobe
DOSTR	Input	Data Out Strobe
/DOSTR	Input	Data Out Strobe
DDIS	Output	Driver Disable
CSOUT	Output	Chip Select Out
SIN	Input	Serial Data Input
RCLK	Input	16x Baud for Receiver
/BAUDOUT	Output	BAUDOUT
SOUT	Output	Serial Data Output
/RTS	Output	Request to Send
/DTR	Output	Data Terminal Ready
/OUT1	Output	Output 1
/OUT2	Output	Output 2
/CTS	Input	Clear To Send
/DSR	Input	Data Set Ready
/DCD	Input	Data Carrier Detect
/RI	Input	Ring Indicator
INTRPT	Output	Interrupt Requests

Stop-Bit Programming

When the UART is programmed for two stop-bits (1.5 in 5-bit mode), the receiver does not check for the second stop-bit. If the first stop-bit is high and the second stop-bit is either high or low, the receiver moves the shift register contents into the

Receiver Buffer Register without posting a framing error. In the majority of applications, this will not cause a problem. However, in cases where a corrupted second stop-bit must be reported as a framing error, the receiver will need modification.

External Crystal Support

This core does not support connection of a crystal directly to the device. The XC4000 and XC5200 families will require a clock input.

Internal Loopback

The internal loopback function (enabled by setting MCR(4)) loops the four lower bits of the Modem Control Register (MCR) to the four upper bits of the Modem Status Register (MSR), however, the Harris 82C50A data sheet is ambiguous as to which MCR bit loops to which MSR bit. The XF8250 Core loops these bits as follows:

MCR(0) (DTR) ----> MSR(5) (DSR)

MCR(1) (RTS) ----> MSR(4) (CTS)

MCR(2) (OUT1) ----> MSR(6) (RI)

MCR(3) (OUT2) ----> MSR(7) (DCD)

If it is discovered that this is incorrect, the design will need modification.

Baud Rate Generator

The Baud Rate Generator differs from the Harris datasheet in the following ways.

Divide by Zero - A divisor value of 0x0000 acts like divide by 65536, where “BAUDOUT” is high for one “XINCLK” period and low for 65535 “XINCLK” periods. The Harris datasheet does not specify what results from a divisor of zero.

Divide by One - A divisor of one is not supported. The “BAUDOUT” signal is always high when a divisor of 0x0001 is used. If only a divisor of one is required, “XINCLK” can be connected directly to “RCLK” and “TXCLK” and the Baud Rate Generator can be removed entirely. If various divisors are required including divide by one, then the Baud Rate Generator will have to be modified.

Other Division - For all divisor values between 0x0002 and 0xffff, the “BAUDOUT” signal is high for one “XINCLK” period and low for N-1 “XINCLK” periods, where N is the divisor value. For divisors of 0x0002 and 0x0003, the “BAUDOUT” signal is identical to the Harris datasheet.

For divisors greater than 0x0003, the Harris BAUDOUT signal is high for two XTAL1 clock periods and low for N-2 XTAL1 clock periods. The duty cycle is not an issue within the XF8250 macro, but external devices that use “BAUDOUT” must consider this.

Verification Methods

Complete functional and timing simulation has been performed on the XF8250 using ViewSim. (Simulation vectors used for verification are provided with the core.) This core has also been used successfully in customer designs. The README.TXT file has a short description of three top level CMD files used to simulate the design. These CMD files call other CMD files (19 in all) and all the CMD files have descriptions and comments as to their purpose.

Recommended Design Experience

Users should be familiar with ViewLogic PROcapture or Office schematic entry and Xilinx design flows. Users should also have experience with microprocessor systems and asynchronous communication controllers.

Ordering Information

The XF8250 Asynchronous Communications Core is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire gate arrays. Please contact Memec for pricing and more information.

Information furnished by Memec Design Services is believed to be accurate and reliable. Memec Design Services reserves the right to change specifications detailed in this data sheet at any time without notice, in order to improve reliability, function or design, and assumes no responsibility for any errors within this document. Memec Design Services does not make any commitment to update this information.

Memec Design Services assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction, if such be made, nor does the Company assume responsibility for the functioning of undescribed features or parameters.

Memec Design Services will not assume any liability for the accuracy or correctness of any support or assistance provided to a user.

Memec Design Services does not represent that products described herein are free from patent infringement or from any other third-party right. No license is granted by implication or otherwise under any patent or patent rights of Memec Design Services.

Memec Design Services products are not intended for use in life support appliances, devices, or systems. Use of a Memec Design Services product in such application without the written consent of the appropriate Memec Design Services officer is prohibited.

All trademarks, registered trademarks, or servicemarks are property of their respective owners.

Related Information

Harris Semiconductor’s *Microprocessor Products for Commercial and Military Digital Applications*, 1992.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

M8254 Programmable Timer

January 12, 1998 Product Specification

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408-733-3344
Fax: +1 408-733-9922
E-mail: sales@virtualipgroup.com
URL: www.virtualipgroup.com

Features

- Multiple, programmable, multi-mode timers
- Real-time clock
- Event counter
- Digital one-shot
- Programmable rate generator
- Square wave generator

Applications

- Binary rate multiplier
- Complex waveform generator
- Complex motor controller
- Baud rate generator

General Description

The M8254 is a programmable interval timer/counter core designed for use with standard micro-processor systems. It has three 16 bit counters, each of which is programmable to generate an interrupt at the end of a user-defined interval.

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000E
CLBs Used	260	260
IOBs Used	22[1]	22[1]
System Clock f_{max}	12.4 MHz	12.4 MHz
Device Features Used	Global Buffers	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XCS40PQ240-3	171[1]	524
XC4020EHQ240-2	171[1]	524
Provided with Core		
Documentation	Core Design Document Designer's Application Note	
Design File Formats	.ngd, XNF netlist Verilog Source RTL available extra	
Constraint Files	.cst file, xactinit.dat	
Verification Tool	Test Vectors	
Schematic Symbols	None	
Evaluation Model	None	
Reference designs & application notes	FPGA Design Document included	
Additional Items	None	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
Entry/Verification Tool	Verilog RTL/Verilog XL simulator	
Support		
Support provided by Virtual IP Group Inc.		

Notes:
1. Assuming all core signals are routed off-chip.

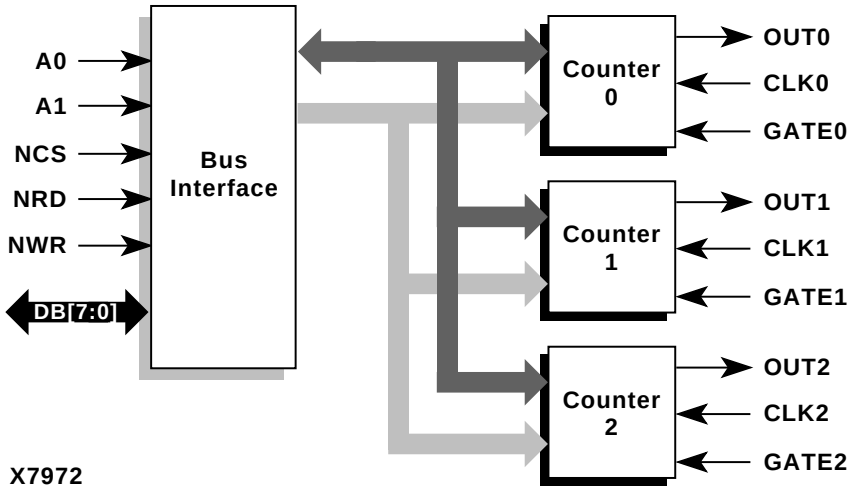


Figure 1: M8254 Block Diagram

Functional Description

The M8254 Core is partitioned into modules as shown in Figure1 and described below.

Bus Interface Block

This block contains decoders to generate counter select and the counter control select signals.

Counter 0-2 Blocks

These independently programmable counter blocks generate output based on the mode selected.

Core Modifications

Virtual IP Group, Inc. can modify this core to vary the number of timers.

Pinout

The pinout has not been fixed to specific FPGA I/O allowing flexibility with the user application. A pinout is suggested for use with a user-constructible evaluation board. Information for this is in the FPGA Design Document included with the core. Signal names are provided in the block diagram shown in Figure 1 and described in Table 1.

Verification Methods

The core has been tested with in-house developed test vectors that are provided with the core.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Bus Interface Signals		
A0, A1	Input	Address Signals
NCS	Input	Chip select, active low
NRD	Input	Read signal, active low
NWR	Input	Write signal, active low
DB[7:0]	Output	8-bit bidirectional CPU data bus
Counter Signals		
OUT0	Output	Output of counter 0
CLK0	Input	Clock input for counter 0
GATE0	Input	Gate input for counter 0
OUT1	Output	Output of counter 1
CLK1	Input	Clock input for counter 1
GATE1	Input	Gate input for counter 1
OUT2	Output	Output of counter 2
CLK2	Input	Clock input for counter 2
GATE2	Input	Gate input for counter 2

Recommended Design Experience

Knowledge of microprocessor based systems is required. The user must be familiar with HDL design methodology, instantiation of Xilinx netlists in a hierarchical design environment and usage of Xilinx Alliance or Foundation development tools.

Available Support Products

FPGA Evaluation Board

The FPGA Design Document included with the core gives directions for constructing a general purpose FPGA evaluation daughter board that can be plugged into a standard part socket on the target system through a flat cable.

Ordering Information

This product is available from Virtual IP Group, Inc. Please contact them for pricing and additional information.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

M8255 Programmable Peripheral Interface

January 12, 1998 Product Specification

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408-733-3344
Fax: +1 408-733-9922
E-mail: sales@virtualipgroup.com
URL: www.virtualipgroup.com

Features

- Multi-mode programmable parallel I/O port peripheral interface
- 24 programmable general purpose I/O signals
- Functionally compatible with Intel 8255A
- Control word read-back facility
- Direct bit set/reset capability
- I/O data transfer with handshaking

Applications

- General purpose I/O ports used to control external peripherals

General Description

The M8255 Programmable Peripheral Interface (PPI) core is a general purpose I/O component to interface peripheral equipment to a microcomputer system bus. The functional configuration of the M8255 is programmable. This core is functionally compatible with the INTEL 8255.

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000E
CLBs Used	111	111
IOBs Used	38[1]	38[1]
System Clock f_{max}	15.2 MHz	15.2 MHz
Device Features Used	Global Buffers	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XCS40PQ240-3	155[1]	673
XC4020EHQ240-2	155[1]	673
Provided with Core		
Documentation	Core Design Document Designer's Application Note	
Design File Formats	.ngd, XNF netlist Verilog Source RTL available extra	
Constraint Files	.cst file, xactinit.dat	
Verification Tool	Test Vectors	
Schematic Symbols	None	
Evaluation Model	None	
Reference designs & application notes	FPGA Design Document included	
Additional Items	None	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
Entry/Verification Tool	Verilog RTL/Verilog XL simulator	
Support		
Support provided by Virtual IP Group, Inc.		

Notes:
1. Assuming all core signals are routed off-chip.

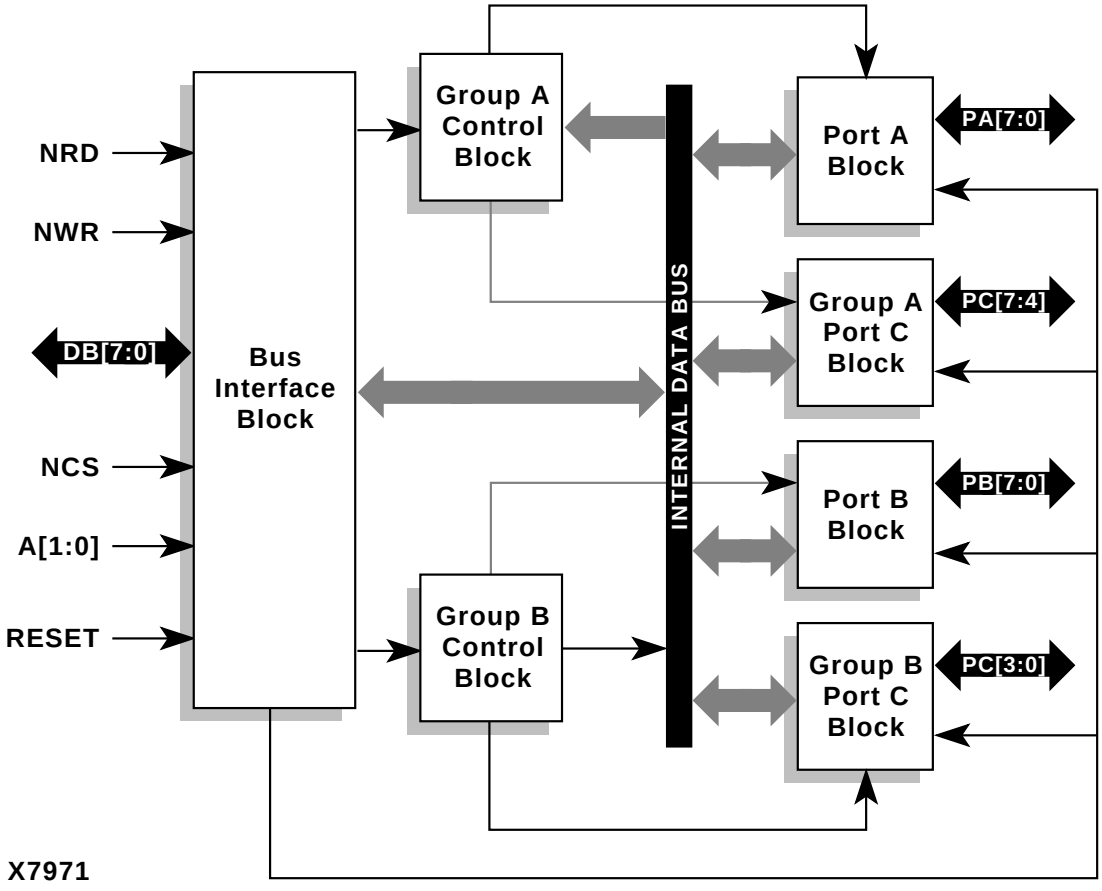


Figure 1: M8255 Functional Block Diagram

Functional Description

The M8255 core is partitioned into modules as shown in Figure 1 and described below.

Bus Interface Block

This block handles CPU Interface, and generates necessary control signals to pass CPU data to control and port blocks.

Group A Control Block

This block handles mode selection and control functions of Port A and Port C blocks for PC[7:4].

Group B Control Block

This block handles mode selection and control functions of Port B and Port C blocks for PC[3:0].

Port A Block

This block provides input and output latches to interface PA[7:0] data to external peripherals.

Port B Block

This block provides input and output latches to interface PB[7:0] data to external peripherals.

Group A Port C Block

This block provides the input and output latches to interface PC[7:4] data to external peripherals. This block also provides the interface to handshake signals in modes 1 and 2.

Group B Port C Block

This block provides the input and output latches to interface PC[3:0] data to external peripherals. This block also provides the interface to handshake signals in modes 1 and 2.

Core Modifications

Virtual IP Group, Inc. can integrate multiple M8255s to increase I/O capacity upon request.

Pinout

The pinout has not been fixed to specific FPGA I/O allowing flexibility with the user application. A pinout is suggested for use with a user-constructible evaluation board. Information for this is in the FPGA Design Document included with the core. Signal names are provided in the block diagram shown in Figure 1 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Bus Interface Signals		
NRD	Input	Read Enable used by host processor to read a byte from selected data port register; active low.
NWR	Input	Write Enable used by host processor to write a byte to data port or control register; active low.
DB[7:0]	In/Out	8 bit bi-directional CPU data bus.
NCS	Input	Chip Enable for accessing internal registers including Port registers for reads or writes; active low.
A[1:0]	Input	2 bit address from CPU to determine In/Out port accessed during Read or Write cycles.
RESET	Input	Reset, active high.
Port Signals		
PA[7:0]	In/Out	8 bit bi-directional Port A data bus.
PB[7:0]	In/Out	8 bit bi-directional Port B data bus.
PC[7:0]	In/Out	8 bit bi-directional Port C data bus.

Verification Methods

The core has been tested with in-house developed test vectors that are provided with the core.

Recommended Design Experience

Knowledge of DMA interfaces in a microprocessor based systems is required. The user must be familiar with HDL design methodology, instantiation of Xilinx netlists in a hierarchical design environment and usage of Xilinx Foundation or Alliance development tools.

Available Support Products

FPGA Evaluation Board

The FPGA Design Document included with the core gives directions for constructing a general purpose FPGA evaluation daughter board that can be plugged into a standard part socket on the target system through a flat cable.

Ordering Information

This product is available from Virtual IP Group, Inc. Please contact them for pricing and additional information.

Related Information

Refer to Specification Document for programming of this core for a typical system application. The user should refer to the Desginer's Application Note for integrating this with other cores. Both documents are included with the core.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

XF8255 Programmable Peripheral Interface

October 1, 1997 Product Specification

Maria Aguilar, Project Coordinator
Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, AZ 85202
Phone: +1 888-360-9044 (USA)
+1 602-491-4311
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Features

- Software and function compatible with Industry Standard 8255
- MCS-85 compatible
- 24 programmable I/O pins
- Fully compatible with most microprocessor families
- Direct bit set/reset capability easing control application interface

Applications

- Embedded Microprocessor Control

General Description

The XF8255 Programmable Peripheral Interface Core is a general purpose programmable I/O device designed for use with most microprocessors. It has 24 I/O pins which may be individually programmed in two groups of 12 and used in three major modes of operation.

The first mode, MODE 0, is basic input/output operation where ports A and B are 8 bits wide and Port C is split into upper and lower halves of 4 bits each. Port A, Port B, Port C upper, and Port C lower can each be independently configured as input or output. This gives a total of 16 direction combinations.In MODE 1 each group may be programmed to have eight lines of input or output. Of the remaining four pins, three are used for handshaking and interrupt control signals.

The third mode of operation, MODE 2, is only available on the Group A ports (Port A and Port C upper). It is a bi-directional bus mode that uses eight lines for a bi-directional bus, and five lines, borrowing one from the other group, for handshaking and interrupt control signals.

AllianceCORE™ Facts

Core Specifics		
Device Family	XC5200	
CLBs Used	57	
IOBs Used	38[1]	
System Clock f_{max}	No External Clock[2]	
Device Features Used	Tbufs, global clock buffers	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC5202-5[3] PC84	27[1]	7
Provided with Core		
Documentation	Core schematics Implementation instructions	
Design File Formats	ViewLogic schematic files Live hierarchical tree LCA Files	
Verification Tool	Machine-readable simulation vectors for ViewLogic ViewSim	
Schematic Symbols	ViewLogic	
Constraint Files	TimeSpecs	
Evaluation Model	None	
Reference designs & application notes	None	
Additional Items	Warranty by MDS	
Design Tool Requirements		
Xilinx Core Tools	XACTstep 6.0.0	
Entry/Verification Tools	ViewLogic PROcapture 6.1 or Workview Office	
Support		
Memec Design Services warrants that the design delivered by Memec Design Services will conform to the design specification. This warranty expires 3 months from the date of delivery of the design database. Contact Memec Design Services for the Design License Agreement with complete Terms and Conditions of Sale.		

Notes:

1. Assuming all core signals are routed off-chip.
2. Function does not have a clock, but will, at a minimum, perform no wait-state operation alongside an 8 MHz 80C86.
3. Specific devices are minimum size and speed recommended - the core will work in any larger or faster devices from the same families.

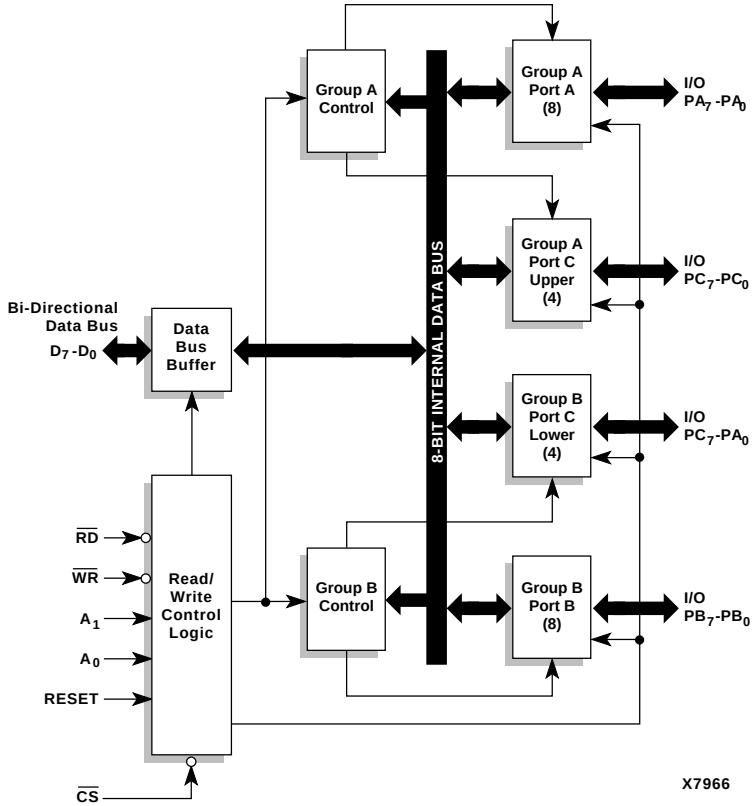


Figure 1: XF8255 Block Diagram

Functional Description

The XF8255 is partitioned into modules as shown in Figure 1 and described below.

Data Bus Buffer

This is a 3-state bi-directional 8-bit buffer used to interface the XF8255 to the system data bus.

Read/Write Control Logic

This logic manages all of the internal and external transfers of both Data and Control or Status words.

Group A and Group B Controls

The functional configuration of each port is programmed by the system software. Each of the Control blocks (Group A and Group B) accepts “commands” from the Read/Write Control Logic, receives “control words” from the internal data bus, and issues the proper commands to its associated ports.

Port A:

One 8-bit data output latch/buffer and one 8-bit data input latch.

Port B:

One 8-bit data input/output latch/buffer and one 8-bit data input buffer.

Port C:

One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control.

Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.

Pinout

The XF8255 may be implemented as stand-alone logic using the provided pinout or may be implemented internally with the users design. Both versions are included with the Core deliverables. Signal names are provided in the block diagram shown in Figure 1, and Table 1.

Core Assumptions

Deviations from the Harris 82C55A functional specification are described below. Memec Design Services will perform any of the listed modifications upon request.

IBF Flip-flops

The Input Buffer Full (IBF) flip-flops on ports A and B are not cleared when RESET is asserted. This is because the Xilinx architecture does not easily support D flip-flops with both asynchronous Set and asynchronous Reset.

Workaround: Perform a dummy read of ports A and B sometime after RESET is pulsed. This will clear the IBF flip-flops.

Input Side Interrupt Request Flip-flops

The input side Interrupt Request (IN_INTR) flip-flops for Port A MODEs 1 and 2 and Port B MODE 1 are cleared asynchronously (level sensitive) when a read is performed on the port. The timing diagram (Figure 7) in the Harris 82C55A data sheet agrees with our implementation, however, the text in the data sheet says it is reset by the falling edge of RD.

Workaround: Since the Harris data sheet is inconsistent on this point and since our XF8255 still meets all the timing relationships indicated in Figure 7 of the Harris 82C55A data sheet, this change should be transparent to the user.

The input side Interrupt Request (IN_INTR) flip-flops for Port A MODEs 1 and 2 and Port B MODE 1, are set on the RISING EDGE of STBx_L when IBFx, INTEx, and Px_RD_L are high. The Harris data sheet indicates that this should be LEVEL sensitive. Since the Xilinx architecture does not easily support flip-flops with both asynchronous Set and Asynchronous Reset, we chose to use the rising edge of STBx_L to set the flip-flop.

Workaround: Since our XF8255 still meets all the timing relationships indicated in Figure 7 of the Harris 82C55A data sheet, this change should be transparent to the user.

Output Side Interrupt Request Flip-flops

The output side Interrupt Request (OUT_INTR) flip-flops for Port A MODEs 1 and 2 and Port B MODE 1, are cleared asynchronously (level sensitive) when a write is performed on the port. The timing diagram (Figure 9) in the Harris 82C55A data sheet agrees with our implementation, however, the text in the data sheet says it is reset by the falling edge of WR.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Control Signals		
RD-	Input	READ: used by the CPU to read status information or data via the data bus; active low
WR-	Input	WRITE: used by the CPU to load control words and data into the 82C55A; active low
A0-A1	Input	ADDRESS: used in conjunction with RD- and WR- inputs to control selection of the control word register ports. A0 and A1 are normally connected to the least significant bits of address bus A0, A1
RESET	Input	RESET: A high on this input clears the control register and all ports (A,B,C) are set to input mode with “bus hold” circuitry turned on
CS-	Input	CHIP SELECT: used to enable the 802C55A onto the data bus for CPU communications; active low
Port and Data Bus Signals		
D0-D7	In/Out	DATA BUS: bidirectional three-state pins connected to system data bus
PA0-PA7	In/Out	PORT A: 8-bit input and output port. Both bus hold high and bus hold low circuitry are present on this port
PB0-PB7	In/Out	PORT B: 8-bit input and output port. Bus hold high circuitry is present on this port
PC0-PC7	In/Out	PORT C: 8-bit input and output port. Bus hold circuitry is present on this port

Workaround: Since the Harris data sheet is inconsistent on this point and since our XF8255 still meets all the timing relationships indicated in Figure 9 of the Harris 82C55A data sheet, this change should be transparent to the user.

The output side Interrupt Request (OUT_INTR) flip-flops for Port A MODEs 1 and 2 and Port B MODE 1, are set on the RISING EDGE of ACKx_L when OBFx_L, INTEx, and Px_WR_L are high. The Harris data sheet indicates that this should be LEVEL sensitive. Since the Xilinx architecture does not easily support flip-flops with both asynchronous Set and asynchronous Reset, we chose to use the rising edge of ACKx_L to set the flip-flop.

Workaround: Since our XF8255 still meets all the timing relationships indicated in Figure 9 of the Harris 82C55A data sheet, this change should be transparent to the user.

Bus-Hold Capability

Ports A, B, and C do not have Bus-Hold capability. This would require two OBUFTs per port pin. Since Xilinx devices only have one OBUFT per pin, it would take two Xilinx pins connected together externally for each port pin.

Workaround: Most applications don't require Bus-Hold capability, but for those that do, a customized version of the XF8255 macro can be developed.

Core Modifications

The XF8255 is designed to meet or exceed the AC Specifications of the Harris 82C55A. However, in most cases the Timespecs can be tightened significantly. In all cases, a post-route timing analysis should be performed to verify performance. Implementation and other customizing is available through Memec Design Services.

Verification Methods

Complete functional and timing simulation has been performed on the XF8255 using Viewsim. (Simulation vectors used for verification are provided with the core.) The README.TXT file has a short description of four top level CMD files used to simulate the design. These CMD files call other CMD files and all the CMD files have descriptions and comments as to their purpose.

Recommended Design Experience

Users should be familiar with ViewLogic PROcapture or Office schematic entry and Xilinx design flows. Users should also have experience with microprocessor systems.

Ordering Information

The XF8255 Programmable Peripheral Interface Core is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire™ gate arrays. Please contact the partner for pricing and more information.

Information furnished by Memec Design Services is believed to be accurate and reliable. Memec Design Services reserves the right to change specifications detailed in this data sheet at any time without notice, in order to improve reliability, function or design, and assumes no responsibility for any errors within this document. Memec Design Services does not make any commitment to update this information.

Memec Design Services assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction, if such be made, nor does the Company assume responsibility for the functioning of undescribed features or parameters. Memec Design Services will not assume any liability for the accuracy or correctness of any support or assistance provided to a user.

Memec Design Services does not represent that products described herein are free from patent infringement or from any other third-party right. No license is granted by implication or otherwise under any patent or patent rights of Memec Design Services.

Memec Design Services products are not intended for use in life support appliances, devices, or systems. Use of a Memec Design Services product in such application without the written consent of the appropriate Memec Design Services officer is prohibited.

All trademarks, registered trademarks, or servicemarks are property of their respective owners.

Related Information

Harris Semiconductors 82C55A CMOS Programmable Peripheral Interface data sheet at: www.harris.com.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

XF8256 Multifunction Microprocessor Support Controller

October 20, 1997 Product Specification

Maria Aguilar, Project Coordinator
Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, AZ 85202
Phone: +1 888-360-9044 (USA)
+1 602-491-4311 (international)
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Features

- Programmable serial asynchronous communications interface for 5-, 6-, 7-, or 8-bit characters, 0.75, 1, 1.5, or 2 stop bits, and parity generation
- On-board baud rate generator programmable for 13 common baud rates up to 19.2 K bits/second, or an external baud clock maximum of 1M bit/second
- Five 8-bit programmable timer/counters; four can be cascaded to two 16-bit timer/counters
- Two 8-bit programmable parallel I/O ports; port 1 can be programmed for port 2 handshake controls and event counter input
- Eight-level priority interrupt controller programmable for 8085, 8086/88, 80186/188 systems and for fully nested interrupt capability
- Programmable system clock to 1x, 2x, 3x, or 5x 1.024 MHz

Applications

- Serial communications
- Process control
- Embedded systems

AllianceCORE™ Facts

Core Specifics		
Device Family	XC4000E	XC5200
CLBs Used	137	64
IOBs Used	38[1]	38[1]
System Clock f_{max}	10+ MHz[2]	
Device Features Used	Tbufs, global clock buffers	

Supported Devices/Resources Remaining		
	I/O	CLBs
XC4005E-4[3] PC84C	29[1]	59
XC5202PC84-5[3]	33[1]	0

Provided with Core	
Documentation	Core schematics Implementation instructions
Design File Formats	ViewLogic schematic files Live hierarchical tree LCA Files
Constraint Files	None
Schematic Symbols	ViewLogic
Verification Tool	Machine-readable simulation vectors for ViewLogic ViewSim
Evaluation Model	None
Reference designs & application notes	None
Additional Items	Warranty by MDS

Design Tool Requirements	
Xilinx Core Tools	XACTstep 6.0.0
Entry/Verification Tools	ViewLogic PROcapture 6.1 or Workview Office

Support

Memec Design Services warrants that the design delivered by Memec Design Services will conform to the design specification. This warranty expires 3 months from the date of delivery of the design database. Contact Memec Design Services for the Design License Agreement with complete Terms and Conditions of Sale.

Notes:
1. Assuming all core signals are routed off-chip.
2. Minimum guaranteed speed.
3. Specific devices are minimum size and speed recommended - the core will work in any larger or faster devices from the same families.

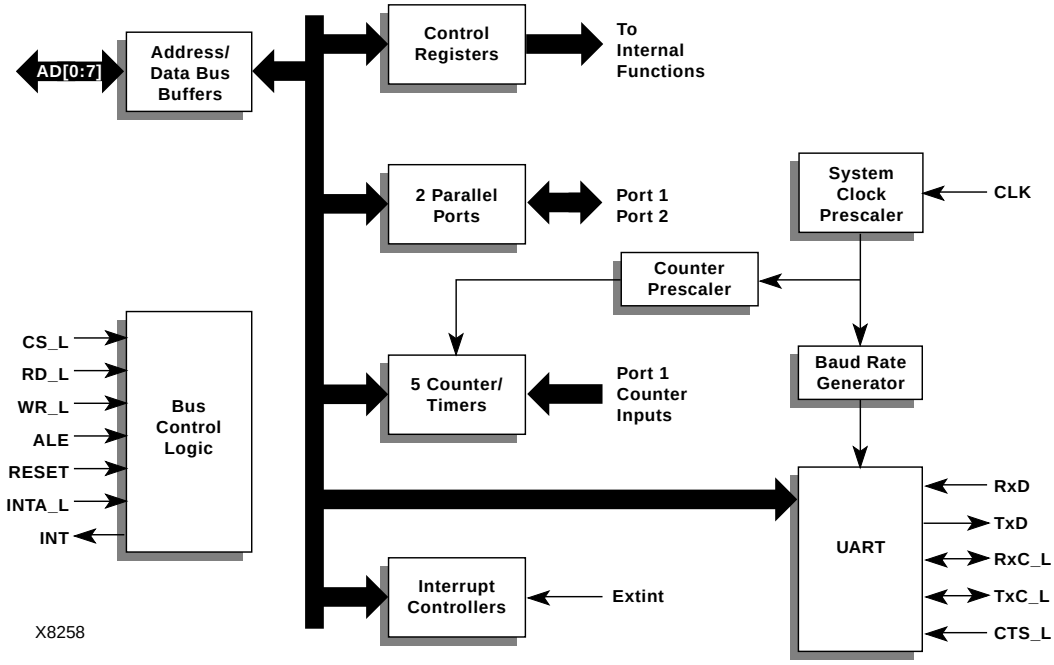


Figure 1: XF8256 Block Diagram

General Description

The XF8256 multifunction universal asynchronous receiver -transmitter combines five commonly used functions into a single device: serial communications, parallel I/O, timing, event counting, and priority interrupt functions.

It is designed to interface to the 8085, 8086/88, 80186/188, and 8051. All of these functions are fully programmable through internal registers. In addition, the five timer/ counters and two parallel I/O ports can be accessed directly by the microprocessor.

Functional Description

The block diagram with internal structure is shown in Figure 1, and described below.

Control Registers

These store commands and input and output data.

Parallel Ports

This block provides two 8-bit general purpose parallel ports. Port 2 has handshake capability and the eight bits of port 1 can be individually defined as input or output.

Counter/Timers

This block provides five 8-bit counter/timers with either 1 kHz or 16 kHz clocks. Four of them can be cascaded to form two 16-bit counter/timers.

Interrupt Controller

This block functions as an eight-level priority interrupt controller with fully nested or normal interrupt priority modes.

System Clock Prescaler

This provides division ratios of 5, 3, 2, and 1 to generate the internal clock operating frequency of 1.024 MHz.

Baud Rate Generator

This block is a programmable baud rate generator with selectable internal or external clock inputs.

UART

This block is a full-duplex serial asynchronous receiver-transmitter with 32x or 64x sampling rate.

Core Modifications

With little effort, the XF8256 can be broken into sub-functions that can be used as needed. Multiple XF8256s can be instantiated into one device to perform functions in parallel. Timing specifications are not critical and can be tightened significantly.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Bus Control Signals		
CS_L	Input	**Chip select:** Low input enables interface functions to send or receive, latched with address on falling edge of ALE. RD_L and WR_L have no effect unless CS_L was latched low during ALE cycle.
RD_L, WR_L	Input	**Read and write control:** Enables data buffers to send or receive, data external.
ALE	Input	**Address latch enable:** Latches address on AD[0:4] and CS_L on falling edge.
RESET	Input	**Reset:** Active high input forces core into initial state until control information is written.
INTA_L	Input	**Interrupt acknowledge:** If core enabled for interrupts, INTA_L indicates interrupt request is being acknowledged by microprocessor. During acknowledgment, the core puts an RSTn instruction (8-bit mode) or a vector (16-bit mode) on data bus.
INT	Output	**Interrupt request:** A high signals processor for interrupt service.
UART Signals		
RxD	Input	**Receive data:** Serial data input.
TxD	Output	**Transmit data:** Serial data output.
RxC_L	In/Out	**Receive clock:** In/Out programmable. As input, it clocks serial data into RxD pin on rising edge of RxC_L. As output, rising edge indicates data on RxD is being sampled. Output remains high during start, stop, and parity bits.
TxC_L	In/Out	**Transmit clock:** In/Out program. As input, it clocks data out of transmitter on falling edge or it can permit use of 32x or 64x clock for receiver and transmitter. As output, it presents internal transmitter clock.
CTS_L	Input	**Clear to send:** Enables serial transmitter; as level sensitive, with CTS_L low, characters loaded into transmit buffer register are transmitted serially; a negative pulse transmits previous character. For edge sensitive, a negative edge on CTS_L transmits next character.
I/O and Clock Signals		
AD[0:7]	In/Out	**Address/data:** Three-state interface to lower 8-bits of processor's multiplexed address/data bus. 5-bit address latched on falling edge of ALE. In 8-bit mode, AD[0:3] selects proper register and AD [1:4]is ignored. In 16-bit mode AD[1:4] selects register and AD0 acts as second chip select, active low.
P1[0:7]	In/Out	**Parallel I/O port 1:** Pin-programmable as general purpose input or output. All outputs latched, inputs are not. Can also serve as functional control pins.
P2[0:7]	In/Out	**Parallel I/O port 2:** 8-bit general-purpose I/O port. Each nibble can input or output. Outputs are latched, inputs are not. Can also be used as 8-bit input or output port when using two-wire handshake mode where both inputs and outputs are latched.
CLK	Input	**System clock:** Used to generate internal timing.

Table 1: Core Signal Pinout (cont.)

Signal	Signal Direction	Description
EXTINT	Input	**External interrupt request:** Input is level-high sensitive; must be held high until an interrupt acknowledge occurs.

In all cases, a functional and post-route timing analysis should be performed to verify performance. Implementation and customizing is available through Memec Design Services.

Pinout

The XF8256 may be implemented internally with the user's design or as stand-alone logic with the pinout that is provided with the Core. For a fast replacement of the industry standard 8256, MDS offers a 40-pin device carrier that is pin-compatible. Signal names are provided in the block diagram shown in Figure 1 and described in Table 1.

Verification Methods

Basic functional simulation has been performed on the XF8256 using ViewSim. Simulation vectors used for verification are provided with the core. This FPGA design was also physically tested and compared with the industry standard 8256 for verification purposes.

Additional Support Products

Memec Design Services supplies a Xilinx-based FPGA Development Module that can be used to hardware test this and other MDS cores. To purchase this, or obtain more information, contact MDS directly.

Recommended Design Experience

Users should be familiar with ViewLogic PRO series or Workview Office schematic entry and Xilinx design flows. Users should also have experience with microprocessor systems.

Ordering Information

The XF8256 Multifunction Microprocessor Support Controller is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire™ gate arrays. To purchase or make further inquiries about this or other Memec Design Services products, contact MDS directly at the location listed on the front page.

Information furnished by Memec Design Services is believed to be accurate and reliable. Memec Design Services reserves the right to change specifications detailed in this data sheet at any time without notice, in order to improve reliability, function or design, and assumes no responsibility for any errors within this document. Memec Design Services does not make any commitment to update this information.

Memec Design Services assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction, if such be made, nor does the Company assume responsibility for the functioning of undescribed features or parameters. Memec Design Services will not assume any liability for the accuracy or correctness of any support or assistance provided to a user.

Memec Design Services does not represent that products described herein are free from patent infringement or from any other third-party right. No license is granted by implication or otherwise under any patent or patent rights of Memec Design Services.

Memec Design Services products are not intended for use in life support appliances, devices, or systems. Use of a Memec Design Services product in such application without the written consent of the appropriate Memec Design Services officer is prohibited.

All trademarks, registered trademarks, or servicemarks are property of their respective owners.

Related Documentation and Information

Intel 8256AH Multifunction Microprocessor Support Controller, data sheet, 1994, order number 230759-002.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products
/logicore/alliance/tblpart.htm

M8259 Programmable Interrupt Controller

January 12, 1998 Product Specification

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408-733-3344
Fax: +1 408-733-9922
E-mail: sales@virtualipgroup.com
URL: www.virtualipgroup.com

Features

- Eight Level Priority Controller
- Expandable to 64 Levels
- Programmable Interrupt Mode
- Individual Request Mask capability

Applications

Applications that require Programmable Interrupt Mode of operation for multiple interrupts.

General Description

The M8259 is a programmable interrupt controller core is used in most microcontroller/microprocessor systems to control and prioritize interrupts. It can handle up to 8 interrupt with programmable masking and priority for the interrupts. It can also be cascaded with up to 8 more M8259 blocks to handle more than 8 interrupts without any additional logic circuits. This design is functionally compatible with Intel's 8259 part.

AllianceCORE™ Facts

Core Specifics		
Device Family	Spartan	XC4000E
CLBs Used	191	191
IOBs Used	26[1]	26[1]
System Clock f_{max}	7.5 MHz	7.5 MHz
Device Features Used	Global Buffers	
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XCS40PQ240-3	167[1]	593
XC4020EHQ240-2	167[1]	593
Provided with Core		
Documentation	Core Design Document Designer's Application Note FPGA Design Document	
Design File Formats	.ngd, XNF netlist Verilog Source RTL	
Constraint Files	.cst file, xactinit.dat	
Schematic Symbols	None	
Verification Tool	Test Vectors	
Evaluation Model	None	
Reference designs & application notes	FPGA Design Document included	
Additional Items	None	
Design Tool Requirements		
Xilinx Core Tools	Alliance 1.3	
Entry/Verification Tool	Verilog RTL/Verilog XL simulator	
Support		
Support provided by Virtual IP Group, Inc.		

Notes:
1. Assuming all core signals are routed off-chip.

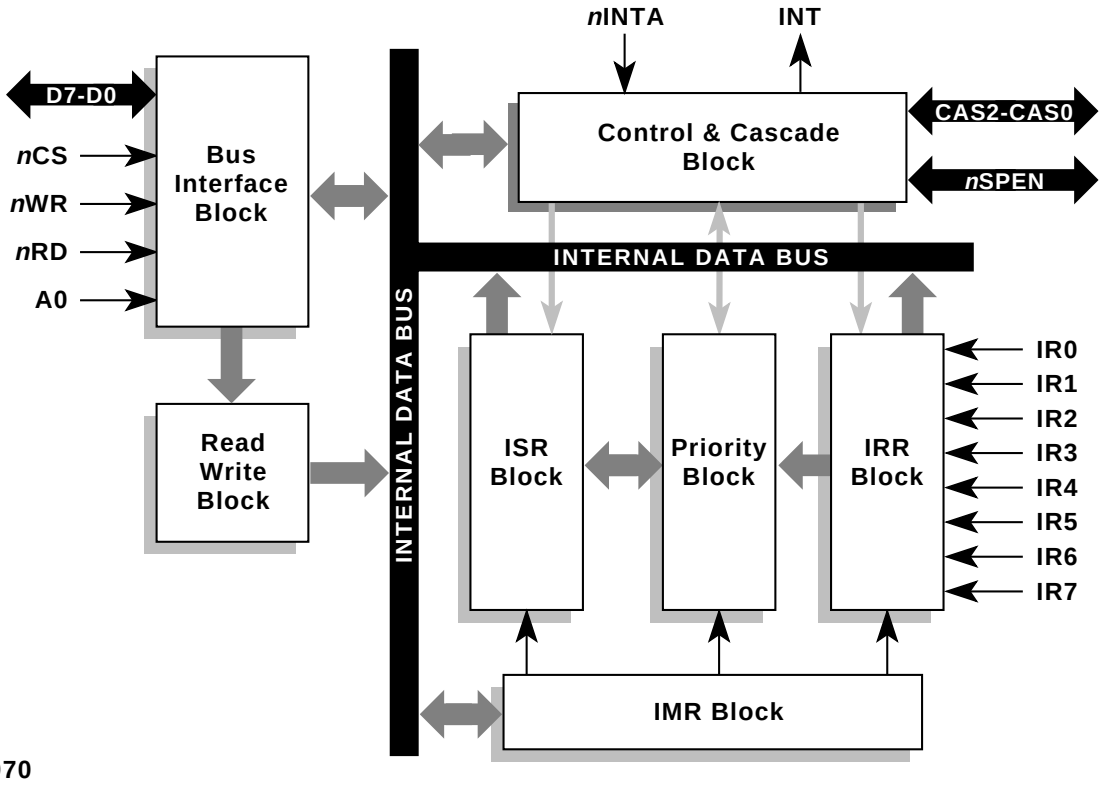


Figure 1: M8259 Functional Block Diagram

Functional Description

The M8259 core is partitioned into modules as shown in Figure 1 and described below.

Bus Interface Block

This block interfaces the core to the system bus. It also generates the internal read, write signals for the read/write block of the core.

Read/Write Block

The read / write block generates various read and write signals for reading status and writing command words.

Interrupt Mask Register (IMR) Block

This block is used for masking the interrupt inputs and masking the ISR bits in the special mask mode.

Interrupt Request Register (IRR) Block

This block stores all interrupt levels that are requesting service.

Interrupt Service Register (ISR) Block

The main function of this block is to store interrupt levels that are being serviced.

Priority Block

This block resolves the priority of valid interrupt request inputs and generates an encoded interrupt request for the control block of the core.

Control Block

This block controls the entire function of core and generates the external interrupt output signal.

Core Modifications

Multiple cores can be cascaded to build support for more Interrupts.

Pinout

The pinout has not been fixed to specific FPGA I/O allowing flexibility with the users application. However for the evaluation board, a pinout has been fixed. This information is provided in the FPGA Design Documentation. Signal names are provided in the block diagram shown in Figure 1 and Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Bus Interface Signals		
D[7:0]	In/Out	8 bit bidirectional CPU data bus through which CPU reads from or writes into core
NCS	Input	Active low chip select signal
NWR	Input	Active low write signal
NRD	Input	Active low read signal
A0	Input	Address signal for selection of internal registers
Control and Cascade Signals		
NINTA	Input	When active low, interrupt acknowledge signal is asserted, core drives programmed interrupt vector onto data bus
INT	Output	Interrupt output signal.
CAS [2:0]	In/Out	Cascade lines used to cascade up to 8 Interrupt controllers for a total capacity of 64 interrupts
NSPEN	Input	Slave program/enable; indicates master/slave mode operation for M8259 in non-buffered mode
IRR Signals		
IR [7:0]	Input	Input interrupt requests for core

Verification Methods

The core has been tested with in-house developed test vectors that are provided with the core. It has also been tested in the FPGA using a hardware evaluation board.

Recommended Design Experience

Knowledge of interface in Microprocessor based systems is required. The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment. Experience in usage of Alliance or Foundation tools is required.

Available Support Products

FPGA Evaluation Board

The FPGA Design Document included with the core gives directions for constructing a general purpose FPGA evaluation daughter board that can be plugged into a standard part socket on the target system through a flat cable.

Simulation Model

In-house developed test vectors are provided to test complete functionality and interface of the core.

Ordering Information

This product is available from Virtual IP Group, Inc. Please contact them for pricing and additional information.

Related Information

The user should refer to the Specification Document for programming this core for a typical application in a system. The user should also refer to the Designer's application note for integrating this with other cores.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

XF8279 Programmable Keyboard Display Interface

October 20, 1997 Product Specification

Maria Aguilar, Project Coordinator
Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, AZ 85202
Phone: +1 888-360-9044 (in the USA)
+1 602-491-4311 (international)
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Features

- Simultaneous keyboard display operations
- Scanned keyboard mode
- Scanned sensor mode
- Strobed input entry mode
- 8-character keyboard FIFO
- 2-key lockout or N-Key rollover with contact debounce
- Dual 4, 8, or 16 numerical display
- Single 8 or 16 character display
- Mode programmable from CPU
- Right or left entry 16-Byte display RAM
- Programmable scan timing
- Interrupt output on key entry

Applications

- User interface for embedded systems

General Description

The XF8279 is a general purpose programmable keyboard and display I/O interface device designed for use with 8-bit microprocessors. This core is functionally compatible with the industry standard 8279 and relieves the CPU from scanning the keyboard or refreshing the display.

The keyboard portion can provide a scanned interface to a 64-contact key matrix such as typewriter style keyboards or thumb switches. The keyboard portion will also interface to an array of sensors or a strobed interface keyboard, such as the Hall effect and ferrite variety. Keyboard entries are debounced and strobed into an 8-character FIFO. Key entries set the interrupt output line to the CPU.

AllianceCORE™ Facts

Core Specifics	
Device Family	XC4000E
CLBs Used	171
IOBs Used	38[1]
System Clock f_{max}	8 MHz[2]
Device Features Used	RAM, OSC4, 3BUFGs

Supported Devices/Resources Remaining		
	I/O	CLBs
XC4005EPQ100-4	39[1]	25

Provided with Core	
Documentation	Core schematics Implementation instructions
Design File Formats	ViewLogic schematic files Live hierarchical tree LCA Files
Constraint Files	None
Schematic Symbols	ViewLogic
Verification Tool	Machine-readable simulation vectors for ViewLogic ViewSim
Evaluation Model	None
Reference designs & application notes	None
Additional Items	Warranty by MDS

Design Tool Requirements	
Xilinx Core Tools	XACTstep 6.0.0/XACT M1.2
Entry/Verification Tools	ViewLogic PRO series or Workview Office

Support

Memec Design Services warrants that the design delivered by Memec Design Services will conform to the design specification. This warranty expires 3 months from the date of delivery of the design database. Contact Memec Design Services for the Design License Agreement with complete Terms and Conditions of Sale.

Notes:
1. Assuming all core signals are routed off-chip.
2. Minimum guaranteed speed.
3. Specific devices are minimum size and speed recommended - the core will work in any larger or faster devices from the same families.

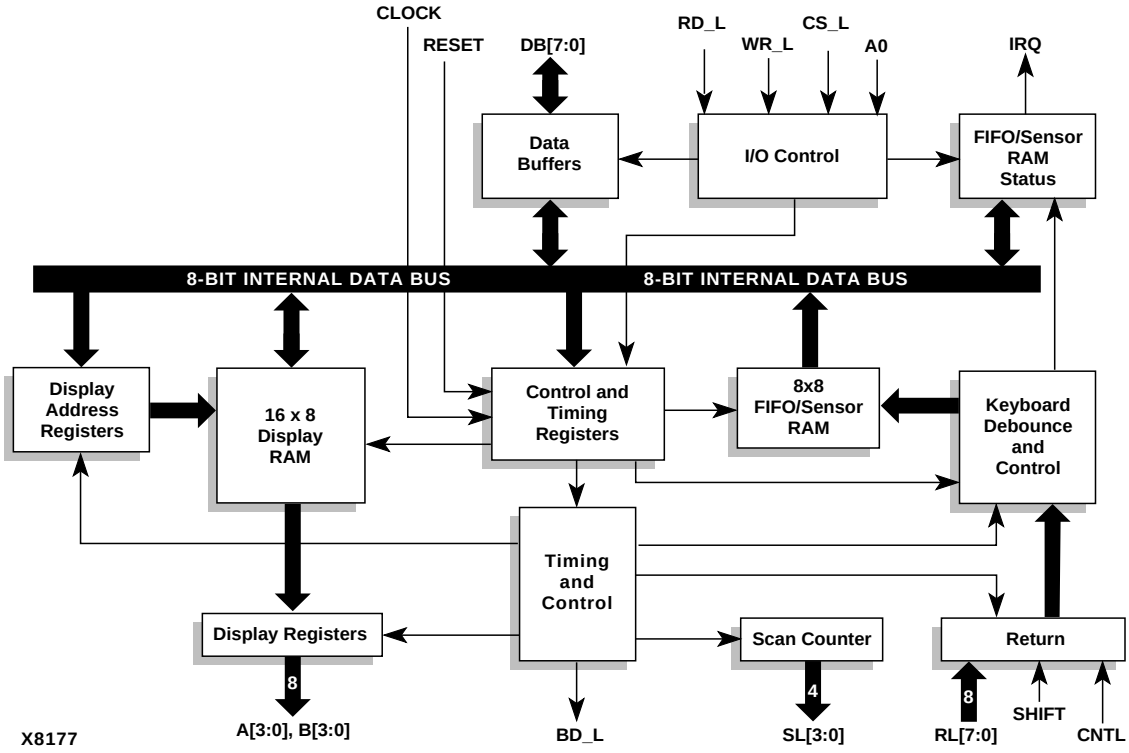


Figure 1: XF8279 Block Diagram

The display portion provides a scanned display interface for LED, incandescent, and other popular display technologies. Both numeric and alphanumeric segment displays may be used as well as simple indicators. The XF8279 has a 16 x 8 display RAM that can be organized into dual 16 x 4 RAMs. The RAM can be loaded or interrogated by the CPU. Both right entry calculator and left entry typewriter display formats are possible. Both read and write of the display RAM can be done with auto increment of the display RAM address.

Functional Description

The XF8279 is partitioned into modules as shown in Figure 1 and described below.

Display Address Registers

These registers hold the write or read display RAM address. They also perform the auto increment function.

Display RAM

This stores the data being displayed.

Display Registers

These registers hold display data during digit switching.

Data Buffers

This block provides direct data output and interface to external bus.

I/O Control

This block controls data flow between internal registers and external bus.

Control and Timing Registers

CPU commands are stored in these registers.

Timing and Control

Reset operations and command functions are accomplished in this block.

FIFO/Sensor RAM

This block provides FIFO or 8x8 RAM for sensor mode.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
CLOCK	Input	**Clock:** Clock from system used to set scan frequency.
RESET	Input	**Reset:** A high input resets the XF8279.
DB[7:0]	In/Out	**Bi-directional data bus:** Three-state data and command bus between CPU and XF8279.
RD_L, WR_L	Input	**I/O Read and Write:** Enables data buffers to send or receive data from external bus.
CS-L	Input	**Chip Select:** A low input enables interface functions to read or write.
A_0	Input	**Buffer Address:** A high input indicates signals are interpreted as a command or status; a low indicates they are data.
IRQ	Output	**Interrupt Request:** In keyboard mode, IRQ is high when data is in FIFO/Sensor RAM. In sensor mode, interrupt line goes high whenever a sensor change is detected.
A[3:0] B[3:0]	Output	**Outputs:** Two data outputs for 16X4 display; ports may be blanked independently.
BD-L	Output	**Blank Display:** Output blanks display during digit switching or can be set by a display blanking command.
SL[3:0]	Output	**Scan Lines:** Used to scan key switch or sensor matrix and display digits; can be either encoded (1 of 16) or decoded (1 of 4).
RL[7:0]	Input	**Return Line:** Inputs connected to scan lines through keys or sensor switches; internal pull-up resistors keep them high until a switch closure pulls one low.
Shift	Input	**Shift:** Shift-input status is stored along with key position on key closure in keyboard modes; it has an internal pull-up resistor.
CNTL	Input	**Control/Strobed Input Mode:** Status is stored on rising-edge; data into FIFO. It has an internal pull-up resistor.

Scan Counter

This counter has two modes. In encoded mode, the scan counter needs an external decoder to provide the scan lines for the keyboard and display. In decoded mode, the scan counter provides a decoded output, however, the keyboard matrix is reduced to 4x8 and only 4 characters can be displayed.

FIFO/Sensor RAM Status

This block keeps track of how many characters are in the FIFO and provides flags for different functional conditions.

Keyboard Debounce and Control

In keyboard mode, this block debounces the return lines. In sensor and strobed modes, data is directed and transferred directly to the FIFO/Sensor RAM.

Return

This block buffers and registers data from the Return, Control, and Shift input lines.

Core Modifications

The XF8279 is designed to meet or exceed the AC Specifications of the Intel 8279. However, in most cases the Timespecs can be tightened significantly. In all cases, a post route timing analysis should be performed to verify performance. Implementation and customizing are available through Memec Design Services.

Pinout

The XF8279 may be implemented internally with the user's design or as stand alone logic with the pinout that is provided with the Core. For a fast replacement of the industry standard 8279 we provide a 40-pin device carrier which is pin compatible. Signal names are provided in the block diagram shown in Figure 1 and in Table 1.

Core Assumptions

There is an important deviation in the design implementation. The internal timing is fixed to 8 MHz (using OSC4) and only the scan frequency is controlled by the external clock. The scan frequency is then synchronized with the internal clock. The reason for this is to meet and exceed the read and write cycle time. However, to our best understanding and practical experience, there are no deviations from the

Intel 8279 functional specifications. Memec Design Services will perform any modification if a deviation is found.

Verification Methods

Basic functional simulation has been performed on the XF8279 using ViewSim. (Simulation vectors used for verification are provided with the core). The design was also physically tested in three different environments and two different Xilinx FPGA devices without any problem.

Recommended Design Experience

Users should be familiar with ViewLogic PRO series or Workview Office schematic entry and Xilinx design flows. Users should also have experience with microprocessor systems.

Available Support Products

Memec Design Services supplies a Xilinx-based FPGA Development Module that can be used to hardware test this and other MDS cores. To purchase this, or obtain more information, contact MDS directly.

Ordering Information

The XF8259 Programmable Keyboard Display Interface is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire™ gate arrays. To purchase or make further inquiries about this or other Memec Design Services products, contact MDS directly at the location listed on the front page.

Information furnished by Memec Design Services is believed to be accurate and reliable. Memec Design Services reserves the right to change specifications detailed in this data sheet at any time without notice, in order to improve reliability, function or design, and assumes no responsibility for any errors within this document. Memec Design Services does not make any commitment to update this information.

Memec Design Services assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction, if such be made, nor does the Company assume responsibility for the functioning of undescribed features or parameters. Memec Design Services will not assume any liability for the accuracy or correctness of any support or assistance provided to a user.

Memec Design Services does not represent that products described herein are free from patent infringement or from any other third party right. No license is granted by implication or otherwise under any patent or patent rights of Memec Design Services.

Memec Design Services products are not intended for use in life support appliances, devices, or systems. Use of a Memec Design Services product in such application without the written consent of the appropriate Memec Design Services officer is prohibited.

All trademarks, registered trademarks, or servicemarks are property of their respective owners.

Related Documentation and Information

Intel 8279/8279-5 Programmable Keyboard/Display Interface, data sheet, 1993, order number 290123-002.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products
/logicore/alliance/tblpart.htm

XF9128 Video Terminal Logic Controller

February 8, 1998 Product Specification

Maria Aguilar, Project Coordinator
Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, AZ 85202
Phone: +1 888-360-9044 (USA)
+1 602-491-4311
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Features

- Software & function compatible with Industry Standard CRT 9128
- Built-in oscillator (XC3190A only)
- Built-in video shift register
- External character generator in LCA configuration ROM
- Bi-directional smooth scroll capability
- Visual attributes include reverse video, intensity control, underline and character blank
- Separate HSYNC, VSYNC and VIDEO outputs
- Composite Sync (RS170 compatible) output
- Absolute (RAM address) cursor addressing
- Software enabled non-scrolling 25th data row available with 25 data row/page display
- Non-interlace display format
- Separate display memory bus eliminates contention problems
- Fill (erase) screen capability
- Standard 8-bit microprocessor data bus interface
- Wide graphics with six independently addressable segments per character space
- Thin graphics with four independently addressable segments per character space
- Single +5V supply
- TTL or CMOS compatible I/O
- Additional optional video parameters (contact MDS for details and availability):
  - Dots per character block (6-8)
  - Raster scans per data row (6-12)
  - Characters per data row (32, 48, 64, 80)

AllianceCORE™ Facts

Core Specifics		
Device Family	XC3190A	XC4005E
Macrocells Used	287	190
IOBs Used	62[1]	61[1]
System Clock fmax	14MHz[2]	14MHz[2]
Device Features Used	Internal oscillator	None
Supported Devices/Resources Remaining		
	I/O	**CLBs**
XC3190A-5PC84C[3]	8[1]	33
XC4005E-4PQ100C[3]	16[1]	6
Provided with Core		
Documentation	Core schematics Implementation instructions	
Design File Format	ViewLogic Workview Office	
Verification Tool	Simulation vectors for ViewLogic's ViewSim or Workview Office	
Schematic Symbols	ViewLogic	
Constraint Files	None	
Evaluation Model	None	
Reference designs & application notes	None	
Additional Items	Warranty by MDS	
Design Tool Requirements		
Xilinx Core Tools	XACTstep 6.0.1 for the XC3190; Alliance 1.3 for the XC4005E	
Entry/Verification Tool	ViewLogic PRO series or Workview Office	
Additional Information		
The Video Terminal Controller Logic Core is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire™ gate arrays		
Support		
Support provided by Memec Design Services		

Notes:
1. Assuming all core signals are routed off-chip.
2. 4MHz maximum recommended clock speed to meet industry standard specifications.
3. Specific devices are minimum size and speed recommended - the core will work in any larger or faster devices from the same families.

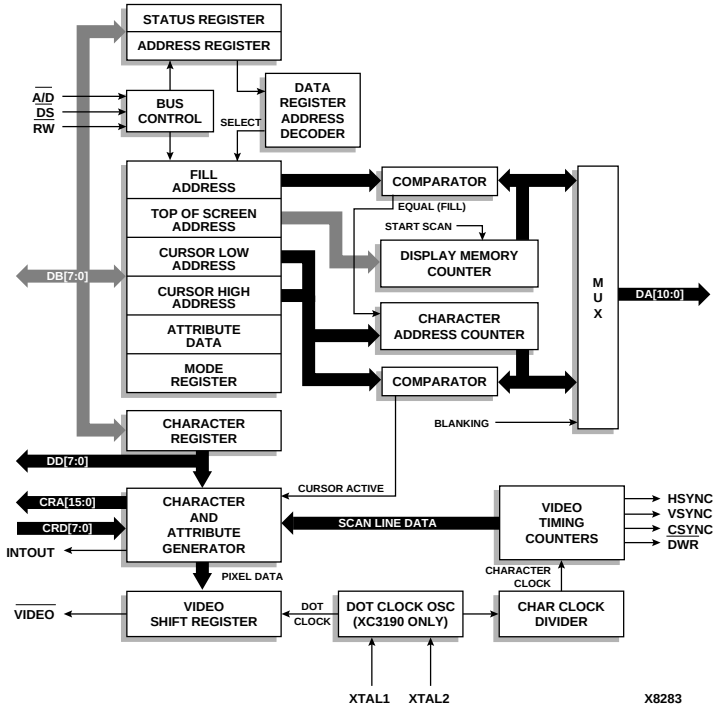


Figure 1: XF9128 Block Diagram

- Data rows per page (8, 10, 12, 16, 20, 24, or 25)
- Horizontal blanking (8-64 characters)
- Horizontal sync front porch (0-7 characters)
- Horizontal sync duration (1-64 characters)
- Horizontal sync polarity
- Two values of vertical blanking
- Two values of vertical sync duration (1-16 scan lines)
- Vertical sync polarity
- External 128-character 5x8-dot font
- Character/cursor underline position
- Scan row and column for thin graphics entity segments
- Scan rows and columns for wide graphics entity elements).

Applications

- ASCII terminal controllers
- Replacement for obsolete Video Terminal Logic Controllers
- Integration of video terminal control logic into an FPGA with other system logic

General Description

The XF9128 Video Terminal Logic Controller (VTLC) Core is a programmable video display controller that combines video timing, video attributes, alphanumeric and graphics generation, smooth scroll, and screen buffer interface functions.

The XF9128 VTLC incorporates many of the features required in building a low-cost, yet versatile, display interface. An external programmable 128-character font provides a full-ASCII character set. Wide graphics allow plotting and graphing capabilities while thin graphics and visual attributes can make the display of forms straight-forward.

The VTLC Core regulates the data flow with data strobe (DS-) and read/write (R/W) enable signals for use with the 6500, Z80®, 68000, and similar microprocessors and microcomputers. The VTLC provides two independent data buses; one bus that interfaces to the processor, and one that interfaces to the display memory. Data is transferred to the display memory from the processor through the XF9128, eliminating contention problems and the need for a separate row buffer.

An internal crystal oscillator is included in the XC3190A version of the VTLC Core, therefore requiring only an external crystal to operate. Pre-programmed constants for critical video timing simplify programming, operation, and improve reliability. A separate non-scrolling status line (enabled or disabled by the processor) is available for displaying system status.

Functional Description

The XF9128 is partitioned into numerous functional blocks as shown in Figure 1, and described below.

XF9128 VTLC Internal Registers

Addressing of the XF9128 internal data registers is accomplished through use of the A/D- and R/W- select inputs qualified by the DS- strobe.

Address Register

Writing to a byte to the ADDRESS register will select the specified register the next time the processor writes to or reads the VTLC data registers.

Status Register

When reading the STATUS register, the DONE bit (DB7 of STATUS Register) will represent the current status of the CHARACTER register.

Data Registers

- FILADD (Fill Address): This register contains the RAM address of the character following the last address to be billed.
- TOSADD (Top of Screen Address): This register contains the RAM address of the first character displayed at the top of the video monitor screen.
- CURLO (Cursor Low): This register contains the eight lower order address bits of the RAM cursor address.
- CURHI (Cursor High): This register contains the three higher address bits of the RAM cursor address (DA10, DA9, DA8).
- ATTDAT (Attribute Data): This register specifies the visual attributes of the video data and the cursor presentation.
- MODE: The AUTO INCREMENT bit in this register specifies whether or not the display memory character address is automatically incremented by the VTLC after every read/write of the CHARACTER register. Note: the visible cursor position is not affected.
- CHARACTER: This register allows access to the display memory for both type transfers and FILL operations.
- CHARACTER SET: Using the DB7-DB0 data bus I/O pins and the MOD SEL bit in the ATTDAT register, the user can address 128 characters, a six segment “wide graphics” and a four segment “thin graphics” entity.

Bus Control

The Bus Control block decodes the control signals from the microprocessor to direct data to/from the registers in the VTLC.

Character and Attribute Generator

This block modifies character cells as directed by the attribute data associated with each character.

Video Shift Register

This block is simply the high-speed shift register that generates the serial video stream.

Video Timing Counters

These counters generate horizontal sync, vertical sync, and composite sync. It also generates the Write Timing for the display memory.

Display Memory Counter

This counter is initialized from the Top Of Screen Address register and counts through a full screen of data.

Cursor Active Comparator

This comparator compares the Cursor Address registers to the Display Memory Counter and informs the Character and Attribute Generator block when the Display Memory Counter is equal to the Cursor Address register.

Fill Address Comparator

This comparator is used to define the boundaries of the area in memory to be filled with a particular character.

Character Address Counter

This counter is used in conjunction with the Display Memory Counter to generate the address for the Display Memory.

DOT Clock Divider

This is the primary scaler for the crystal frequency which generates DOT Clock.

Character Clock Divider

This is the divider which counts the number of dots per character to generate Character Clock.

MUX

The MUX controls the source of address to the display memory.

Core Modifications

The XF9128 is designed to meet or exceed the AC Specifications of the Stand Microsystems Corporation's CRT 9128. However, in most cases the Timespecs can be tightened significantly. In all cases, post-route timing analysis should be performed to verify performance. Implementation and other customizing are available through Memec Design Services.

Pinout

The XF9128 may be implemented internally with the users design or as stand-alone logic with the pinout (XC3190A only) that is provided with the Core. Signal names are provided in the block diagram shown in Figure 1 and in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A/D-	Input	**Register Select**: state of this pin determines if data is being read from, or written to address or status register, or a data register.
DS-	Input	Causes data to be strobed into or out of VTLC from microprocessor data bus depending on state of R/W signal.
R/W-	Input	**Read/Write Select**: determines whether processor is reading data from or writing data into VTLC (high for read, low for write).
DB[7:0]	In/Out	**Processor or Data Bus**: 8-bit bi-directional processor data bus.
DD[7:0]	In/Out	8-bit bi-directional data bus to display memory.
CRA[15:0]	Output	Address to external character ROM.
CRD[7:0]	Input	Data from external character ROM.
XTAL1,2 (XC3190A) XTAL1 (XC4005E)	Input	**External Crystal**: external TTL level clock may be used to drive XTAL1 (in which case XTAL2 is left floating). Note: 4005E has no XTAL2 connection.
DA[10:0]	Output	11-bit address bus to display memory
HSNYC	Output	Horizontal sync signal or monitor.
VSNYC	Output	Vertical sync signal or monitor.
CYSNYC	Output	Used to generate RS170 compatible composite VIDEO signal for output to a composite VIDEO monitor.
DWR-	Output	**Display Write:** Write strobe to display memory.
INTOUT	Output	intensity level modification attribute bit (synchronized with video data output).
VIDEO-	Output	**Video Output**: digital TTL waveform used to develop VIDEO and composite VIDEO signals to monitor. Signal polarity is: HIGH = BLACK LOW = WHITE

Core Assumptions

The character generator ROM and the ROM that contains the VTLC macro bitstream are the same ROM. The FPGA loads itself from the ROM using master-parallel mode and then the VTLC macro uses the same ROM for the character look-up table.

Verification Methods

Basic functional simulation has been performed on the XF9128 using ViewSim. (Simulation vectors used for verification are provided with the core). This design was also physically tested in one application using the XC3190A without any problem.

Recommended Design Experience

Users should be familiar with ViewLogic PRO series or Workview Office schematic entry and with Xilinx FPGAs. Users should also have experience with microprocessor systems using video system controllers.

Ordering Information

The XF9128 Video Terminal Logic Controller Core is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire gate arrays. Please contact Memec for pricing and more information.

Information furnished by Memec Design Services is believed to be accurate and reliable. Memec Design Services reserves the right to change specifications detailed in this data sheet at any time without notice, in order to improve reliability, function or design, and assumes no responsibility for any errors within this document. Memec

Design Services does not make any commitment to update this information.

Memec Design Services assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction, if such be made, nor does the Company assume responsibility for the functioning of undescribed features or parameters.

Memec Design Services will not assume any liability for the accuracy or correctness of any support or assistance provided to a user.

Memec Design Services does not represent that products described herein are free from patent infringement or from any other third-party right. No license is granted by implication or otherwise under any patent or patent rights of Memec Design Services.

Memec Design Services products are not intended for use in life support appliances, devices, or systems. Use of a Memec Design Services product in such application without the written consent of the appropriate Memec Design Services officer is prohibited.

All trademarks, registered trademarks, or servicemarks are property of their respective owners.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

DRAM Controller

November 11, 1997 Product Specification

NMI Electronics Ltd.
Fountain House
Great Cornbow
Halesowen
West Midlands
B63 3BL
United Kingdom
Phone: +44 (0) 121 585 5979
Fax: +44 (0) 121 585 5764
E-mail: ip@nmi.co.uk
URL: www.nmi.co.uk

Features

- Single-chip DRAM controller in a Xilinx XC9500 CPLD
  - Easy to use and modify due to predictable XC9500 timing
- Designed for high speed, high performance applications
- Support for burst mode CPU's
- Utilizes fast page mode or EDO DRAM's
- CAS before RAS refresh
- Supports multiple DRAM types
- Available in source code format for ease of customization
- Compatible with industry standard 72 pin SIMM's

Applications

- Embedded systems in industrial, telecommunications, test or point-of-sale applications
- High performance peripheral equipment, e.g. printers

AllianceCORE™ Facts

Core Specifics	
Device Family	XC9500
Macrocells Used	75[1,4]
IOBs Used	58[1]
System Clock fmax	50+ MHz[2]
Device Features Used	GCLK

Supported Devices/Resources Remaining		
	I/O	Macrocells
XC95216-15 HQ208C[3]	110[1]	141[1]

Provided with Core	
Documentation	Design User Guide Sample files for top level module in VHDL
Design File Formats	VHDL source code
Verification Tool	Viewlogic ViewSim
Schematic Symbols	None
Evaluation Model	Evaluation Board available extra
Reference designs & application notes	Application notes included in Design User Guide
Additional Items	TimingDesigner™ design file

Design Tool Requirements	
Xilinx Core Tools	XACTstep M1.3
Entry/Verification Tool	Viewlogic Workview Office™ 7.4

Support
90 days e-mail and telephone support from NMI Electronics included in the Core price. Support does not cover user core modifications; please refer to "Core Modifications" section. Maintenance contracts available.

Notes:
1. Assumes default configuration, see Functional Description.
2. Depends on device speed selected and other system device speeds. 50MHz operation has been verified with the i486DX4™ and XC95216-15. 40MHz operation has been verified with the same system and XC95216-20.
3. The core has been proven in this device but will work in any XC9500 device with sufficient resources.
4. This number can vary slightly due to device utilization and fitter settings. Typically the number of macrocells varies between 70 and 75 but may be higher in devices with a high utilization and/or speed requirement.

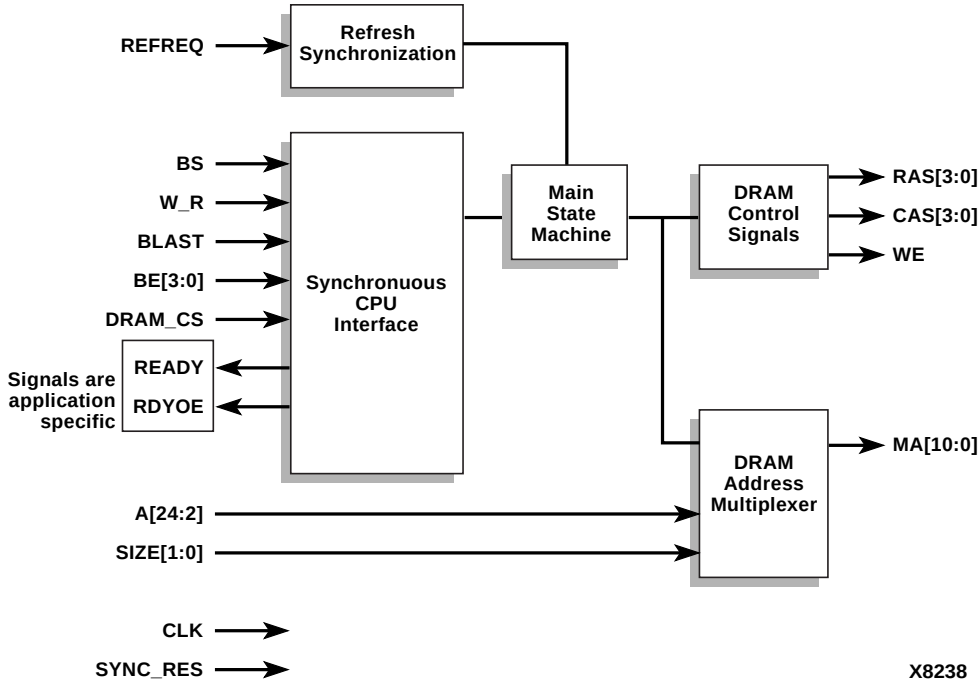


Figure 1: DRAM Controller Block Diagram

General Description

The DRAM Controller is extremely flexible and can be configured to be used with most DRAM types, most microprocessors and many other applications, see Core Modifications.

Functional Description

The DRAM Controller is supplied as a single, VHDL source code module. The functional blocks shown in Figure 1 are for descriptive purposes only.
The default configuration of the DRAM Controller is as follows:

- i486™-like CPU interface, including burst mode
- Two bank, 32 bit wide, 72 pin SIMM compatible DRAM control signals
- Support for 256K, 1M and 4M bit deep DRAM's

Synchronous CPU Interface

In order to achieve maximum performance the DRAM Controller utilizes a fully synchronous interface to its host CPU or control logic. This does not, however, prevent the user from adding extra logic to create an asynchronous interface.

Main State Machine

The DRAM Controller is state machine driven. The state machine, along with the input clock frequency, controls the timing of the DRAM signals.

Refresh Synchronization

Refresh requests are not automatically generated by the DRAM Controller as these can often be generated using other, lower frequency clocks which are available in many typical system designs. Due to this fact, the DRAM Controller fully supports asynchronous refresh request inputs.

DRAM Control Signals

The DRAM control signals RAS, CAS and WE are synchronously generated from the state machine outputs. Burst mode is supported using page mode, refresh utilizes CAS-before-RAS, while writes use “early write” mode.

DRAM Address Multiplexer

The DRAM address multiplexer can be configured to support most DRAM sizes and organizations. The Design User Guide describes this process in detail.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
CLOCK	Input	Primary system clock; rising edge triggered.
SYNC_RES[1]	Input	System reset, synchronous to CLOCK; active High.
BS	Input	Bus cycle start signal; active High. One CLOCK period wide. Identical to inverse of i486™ ADS signal.
W_R[1]	Input	System read or write cycle indicator. Cannot change state during whole bus cycle, or during a complete burst sequence; 0=read 1=write.
BLAST[1]	Input	Burst last indicator. Active High for last bus cycle in a burst sequence.
BE[3:0][1]	Input	Byte enable signals for default, 32 bit configuration. Writes are gated by state of byte enables; however all reads return 32 bit data; active Low.
DRAM_CS[1]	Input	Select signal for DRAM Controller; active High. Must be active while BS is active for DRAM Controller to respond to a bus cycle. Usually driven by an address decode.
A[24:2][1]	Input	System address lines for default, 32 bit configuration; active High.
SIZE[1:0]	Input	DRAM module size selection inputs. These should not be changed during normal system operation. For default configuration these are: 00 - 256 KBit DRAM's 01 - 1 MBit DRAM's 1x - 4 MBit DRAM's
REFREQ	Input	Refresh request input; active High. May be asynchronous to CLOCK, and is rising edge triggered. Typical period is 15.625 microseconds maximum.
RAS[3:0]	Output	DRAM RAS outputs; active Low. To be compatible with standard 72 pin SIMM's, RAS0 and RAS1 are functionally identical and drive half of the first bank of DRAM's each. RAS2 and RAS3 drive half of the second bank of DRAM's each.
CAS[3:0]	Output	DRAM CAS outputs; active Low; Configured for fast page mode operation by default and used in early-write mode, when CAS activation is gated by their respective BE's.
WE	Output	DRAM WE output; active Low. DRAM Controller uses early-write mode, so WE is used as an indicator only.
MA[10:0]	Output	Multiplexed row and column address outputs; active High. Multiplexing scheme determined by state of SIZE inputs.
READY	Output	System Ready output; active High. Default configuration is for i486™ microprocessor. Output is usually system host specific.
RDYOE	Output	System Ready 3-state enable output; active High. Default configuration utilizes a 3-state driver for System Ready. Output is usually system host specific.

Note:
1. These signals must be applied synchronously to CLOCK.

Core Modifications

As supplied, the DRAM Controller is designed for optimum performance in the 40MHz to 50MHz range, using 70ns fast-page-mode DRAM's. The system interface is i486™-like.

Many system designs will require that the core be modified before it can be used, especially with regard to the system interface. For this reason the core is supplied in source code format, and has been written with modification in mind.

The Design User Guide supplied with the core describes in detail how the core can be modified for different system interfaces, different clock speeds and different DRAM types

and speeds.
Please note that NMI support does not cover user core modifications. NMI offers design services, including core modifications, for additional cost.

Pinout

The pin functions of the DRAM Controller core in its default configuration are shown in Table 1. The pinout is not fixed to any specific CPLD I/O, and can generally be modified to suit the user's application.
Signal names are provided in the block diagram shown in Figure 1, and described in Table 1.

Verification Methods

Both functional and timing simulation have been carried out using ViewSim under Viewlogic Workview Office™. Simulation vectors used for verification are provided with the core in the form of ViewSim command files.
The core has been extensively tested on target hardware using an NMI developed evaluation card.

Recommended Design Experience

Users should be familiar with VHDL and Xilinx design flows. Experience with microprocessor or similar system design is recommended. The core can easily be integrated into hierarchical VHDL designs.

Available Support Products

NMI has developed a Xilinx core evaluation card and this will be available, at additional cost, in the near future.

Ordering Information

To make further enquiries or purchase the DRAM Controller Core, please contact NMI directly at the location detailed on the front page. NMI also offers core integration and design services, the latter covering not only CPLD and FPGA design but also complete systems design.
NMI cores are purchased under a Licence Agreement, copies of which are available on request.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

GV & Associates, Inc.

GV & Associates, Inc.

23540 Oriente Way
Ramona, CA 92065 USA
Phone: +1 760-789-7015
Fax: +1 760-789-7015
E-mail: gvander@sd.cts.com

Features

- 40 MHz maximum input A/D sample rate
- On-board 40 MHz Direct Digital Synthesizer
- 15 MHz sample Bandwidth
- FPGA Logic Expansion (from 13K gates to 100K gates)
- 40 MHz TI TMS320C541 DSP Processor
- 128K X 16 Flash EPROM
- 64K X 16 SRAM
- RS232 Serial Port
- FPGA Access to DSP Processor SRAM
- DSP Processor Interface to FPGA
- 24 bit FPGA Local Bus
- Xilinx FPGA Processor Peripheral, Master Parallel and Download Cable Configurations supported
- 40 MHz maximum output D/A sample rate
- Stereo Audio Amplifier with Stereo Jack
- Selectable 3.3V source for Xilinx FPGAs
- Programmable A/D Sample Clock
- Programmable DDS Sample Clock

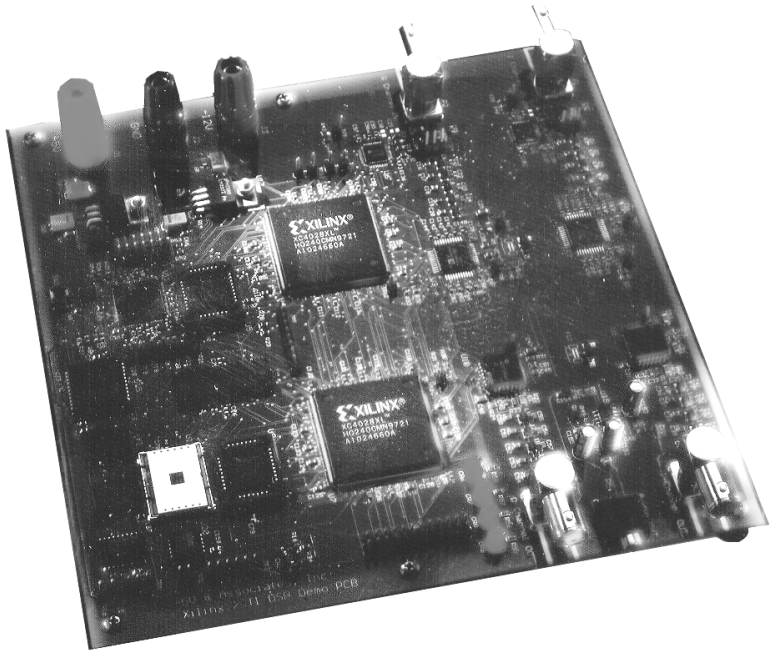


Figure 1: GVA-100 DSP Prototyping Platform

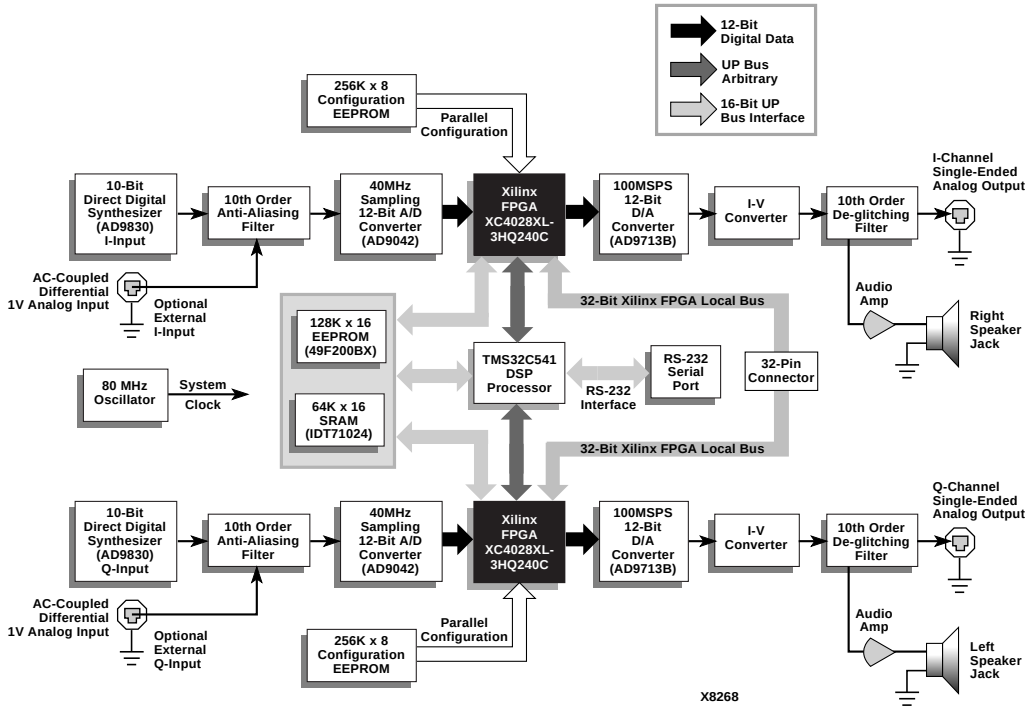


Figure 2: GVA-100 Block Diagram

General Description

The GVA-100 Digital Signal Processing Prototyping Platform is designed for testing complex DSP or other channel coding designs. This prototyping platform provides a highly flexible tool for testing various software and hardware DSP applications using the Xilinx XC4000 family.

The GVA-100 Supports the following Xilinx FPGAs:

- XC4013E-3PQ240C
- XC4020E-3PQ240C
- XC4025E-3PQ240C
- XC4028XL-3HQ240C
- XC4036XL-3HQ240C
- XC4044XL-3HQ240C
- XC4052XL-3HQ240C
- XC4062XL-3HQ240C

Functional Description

The platform's general configuration consists of an I and Q channel which pass through a 10th order low pass filter. The 10th order low pass filter band limits the input signals to a 15 MHz bandwidth. The signal rejection is -60 dB at 25 MHz. Next, the signals are digitized by a 12 bit A/D. The sample rate (maximum of 40 MHz) of the A/D is programmable since it is generated by the Xilinx FPGA. The digitized signals are then ready to be processed by the customer's algorithm which could be implemented in hardware by the Xilinx FPGA and/or in software by the TI TMS320C541 DSP Processor.

The digitized data may be accessed by the DSP Processor via the microprocessor / Xilinx bus interface. Once the signals have been processed, they are converted back to an analog waveform by a 100 MSPS D/A via the Xilinx FPGA. The processed analog waveforms are passed through a 10th order de-glitching filter which is band limited to 15 MHz. The de-glitched analog signal is passed to a 50 ohm BNC output for viewing and to an audio amp which may be accessed via the stereo jack.

Additionally, Direct Digital Synthesizers (DDS) on each channel provide the option for an on-board signal source. The DDS can be programmed by the FPGA or the DSP Processor via the microprocessor address/data bus interface. The clock rate of the DDS is programmable since it is generated by the FPGA.

The FPGA may access the Processor memory (SRAM) through the bus arbitration circuitry of the TMS320C541. Also, the two Xilinx FPGAs have a 32 bit local bus which allows for the direct transfer of data between the two devices. This could be used as direct digital data port or as a data transfer path between both FPGAs to implement the desired DSP algorithm.

Ordering Information

This product is available directly from GV & Associates. Please contact them for pricing and more information.

Related Information

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

MDS FPGA Development Module

October 20, 1997 Product Specification

Maria Aguilar, Project Coordinator
Memec Design Services
1819 S. Dobson Rd., Suite 203
Mesa, AZ 85202
Phone: +1 888-360-9044 (in the USA)
+1 602-491-4311 (international)
Fax: +1 602-491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Features

- Xilinx FPGA-based hardware development module
  - Evaluate Memec Design Services AllianceCORE modules
  - Debug custom FPGA logic
- 40-pin DIP-compatible footprint
  - Small physical size
  - Replace obsolete devices in existing systems, without PCB re-layout
- Flexible FPGA programming options:
  - Serial download cable with readback capability
  - Socketed serial PROM
- Device read-back capability for debug
- Two versions available:
  - XC4005XL (up to 9,000 gates, including RAM)
  - XC5206 (up to 10,000 gates, no RAM)

General Description

The MDS FPGA Development Module is an ideal platform for Xilinx-based hardware development. The module provides a 40-pin DIP socket-compatible footprint for direct replacement of industry standard components using a Xilinx FPGA. It can be used to evaluate MDS cores in a target system, eliminating the need for custom board fabrication or relayout.

Functional Description

The module is built on a small profile PCB, slightly larger than a 40-pin DIP with a height of only 0.6 inches. It contains a single Xilinx FPGA and a socketed serial configuration PROM.

The module provides two options for FPGA configuration. One is an 8-pin DIP socket that makes changing serial

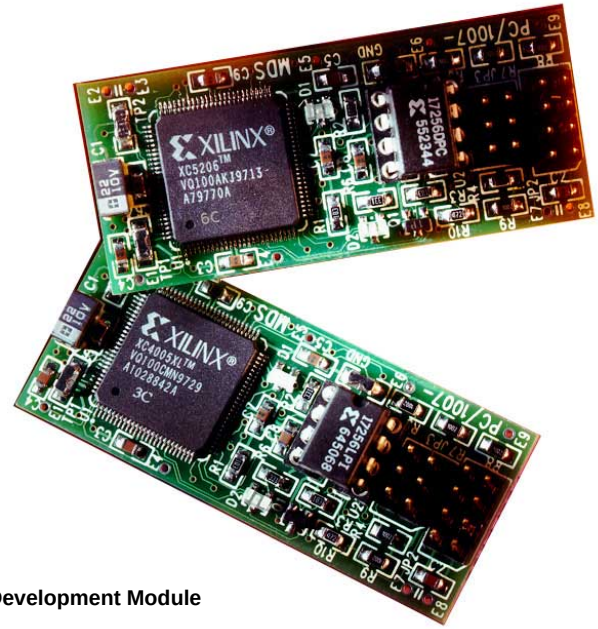


Figure 1: MDS FPGA Development Module

PROMs easy. This is also an option if the module is to be used in a permanent configuration, such as direct socket replacement of an existing device.

Alternatively, a serial download cable can be attached to on-board headers and used to configure the FPGA during the prototyping stage of the design. It can also be used to perform device readback during debug.

Two versions of the module are available. The first includes a Xilinx XC5206 FPGA that provides up to 10,000 logic gates for 5V systems.

The second includes a Xilinx XC4005XL 3.3V FPGA that provides up to 9000 gates, including on-chip RAM. This module can also be used in a 5V system that provides a regulator for the module power supply.

Module power and ground pins match industry-standard configuration where ground is at Pin 20 and V_{CC} is Pin 40.

Additional Support Products

Memec Design Services provides Xilinx FPGA design services and Xilinx FPGA cores.

MDS has available cores that, when implemented in a Development Module can form direct plug compatible replacements for the following industry functions:

- 8250 UART
- 8255 Programmable Peripheral Interface
- 8256 Peripheral and Multifunction UART
- 8279 Keyboard/Display Controller

Ordering Information

The MDS FPGA Development Module is provided under license from Memec Design Services for use in Xilinx programmable logic devices and Xilinx HardWire™ gate arrays. To purchase or make further inquiries about this or other Memec Design Services products, contact MDS directly at the location listed on the front page.

Information furnished by Memec Design Services is believed to be accurate and reliable. Memec Design Services reserves the right to change specifications detailed in this data sheet at any time without notice, in order to improve reliability, function or design, and assumes no responsibility for any errors within this document. Memec Design Services does not make any commitment to update this information.

Memec Design Services assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction, if such be made, nor does the Company assume responsibility for the functioning of undescribed features or parameters. Memec Design Services will not assume any liability for the accuracy or correctness of any support or assistance provided to a user.

Memec Design Services does not represent that products described herein are free from patent infringement or from any other third-party right. No license is granted by implication or otherwise under any patent or patent rights of Memec Design Services.

Memec Design Services products are not intended for use in life support appliances, devices, or systems. Use of a Memec Design Services product in such application without the written consent of the appropriate Memec Design Services officer is prohibited.

All trademarks, registered trademarks, or servicemarks are property of their respective owners.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Microprocessor Based Core Evaluation Card

February 8, 1998 Product Specification

NMI Electronics Ltd.

Fountain House
Great Cornbow
Halesowen
West Midlands
B63 3BL
United Kingdom
Phone: +44 (0) 121 585 5979
Fax: +44 (0) 121 585 5764
E-mail: ip@nmi.co.uk
URL: www.nmi.co.uk

Features

- Microprocessor based IP Core evaluation card
- i486DX™ based system, supporting both +5V and +3.3V CPU's with selectable clock speeds up to 50MHz
- Single XC95216 system control CPLD with integrated NMI DRAM Controller Core
- XC4000XL site available as a dedicated memory controller (in place of the XC95216)
- Single, industry standard 72 pin SIMM DRAM memory connector
  - Configurable for SDRAM operation with custom NMI module
- Parallel connected XC95216 and XC4000E available for general purpose I/O core integration:
  - Connected to full 32 bit i486DX™ bus
  - RS232 port connection
  - PC-compatible parallel port connection
  - I^2C master/slave controller connection
  - General purpose pin strip header connection
- XC9500 parts programmable in-circuit via JTAG chain

Figure 1: Microprocessor Based Core Evaluation Card

- XC4000 parts programmable in-circuit via either programming cable or directly from the system CPU
- System support devices include boot EPROM and/or Flash EPROM, SRAM and serial port
- Full featured debug monitor facilitates thorough evaluation and test of cores
- Operates from single +8V to +20V DC input

General Description

NMI's AllianceCORE™ Evaluation Card is intended to allow developers to quickly evaluate the functionality of CPU peripheral and memory controller cores. Developers can also test, using real target hardware, modifications to the cores and/or their own peripheral logic. This increases confidence that user designs will be fully functional when integrated into the final target hardware platform.

Functional Description

The Evaluation Card's host CPU is i486DX™ and both +5V (e.g. i486DX™ and i486DX2™) and +3.3V (e.g. i486DX4™) versions of the CPU are supported. CPU operating voltage is auto-sensed using the standard Intel 486DX4™ mechanism.

The i486DX™ has a general purpose, high speed synchronous bus and this allows developers to easily migrate designs to other microprocessor or system interfaces. The bus clock speed can be set to a number of frequencies ranging from 16MHz to 66MHz (although the Intel 486DX4™ is limited to 50MHz maximum). The bus clock is distributed using individual, low skew (PLL controlled) clock drivers to the CPU and each of the Xilinx PLD's.

The XC95216 system controller CPLD is the only required PLD device as it provides all of the basic CPU state and address decoding. NMI's AllianceCORE DRAM Controller can also be integrated into this device, providing the board with increased memory resources. The CPU has a central resource of boot EPROM, Flash EPROM, SRAM and RS232 serial port with which it can run its built in debug and test monitor.

An optional XC4KXL device can be used to provide the memory controller function instead of the XC95216 (the XC95216 is still required in the system though). A single, industry standard 72 pin DRAM SIMM connector is provided on the board, allowing the use of any standard SIMM module with either one or two banks of DRAM. The SIMM power supply can be set to either +3.3V or +5V. A single configuration jumper also allows the bus clock to be routed to a no connect pin on the SIMM connector, allowing custom Synchronous DRAM SIMM modules to be used also.

Two (optional) peripheral controller PLD's are also provided on the board: an XC95216-HQ208 and an XC4KE-HQ240. The devices are effectively connected in parallel, with one side each connected to the full i486DX™ system bus, the other side connected to some general purpose I/O. Both devices employ +5V I/O buffers.

The general purpose I/O consists of:

- A full 9 pin RS232 serial port (level translators provided)
- A PC parallel port (ESD protection provided)
- A separate, Philips Semiconductors I^2C master/slave controller
- Pin headers (16 off)
- Two LED's

The board is provided with a full featured debug monitor which has been optimized for debugging and testing hardware. The monitor includes:

- Standard memory and I/O display and modification commands.
- Assembly language coded fill and test commands which ensure maximum CPU bandwidth utilization of memory. These commands are very useful for stress testing designs.
- Software assembler and disassembler (80186 instruction set).
- File transfer via the serial port. Includes binary format plus Intel HEX and Microsoft EXE program loaders.
- Basic DOS INT 21 emulator to ease user test software development.
- Built in macro language.

The two XC95216 devices are configured in a single JTAG chain and can easily be programmed using the XACTstep M1 JTAG CPLD Programmer software. The two XC4K devices can be programmed using asynchronous peripheral mode via the debug monitor; alternatively each device is provided with a separate serial download cable connection.

The Evaluation Card is powered from a single, +8V to +20V DC input. Switched-mode power converters are used, resulting in low heat dissipation by the board (other than by the CPU). The power supplies are protected against input voltage reversal.

The Evaluation Card will be CE approved by NMI.

Available Support Products

NMI will develop a Synchronous DRAM SIMM module which will be compatible with the Evaluation Card and this will be available, at additional cost, in the near future.

Ordering Information

To make further enquiries or purchase the Core Evaluation Card, please contact NMI directly at the location detailed on the front page. NMI also offers core products and design services, the latter covering not only CPLD and FPGA design but also complete systems design.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

February 8, 1998

Overview

Xilinx is proud to present our AllianceCORE partner companies. Each is an expert in its respective field and ready to assist you with products and services for your application. This section profiles each partner and their particular areas of expertise. This way, you can learn more about the partners since many of them may be new to you.

Most partners supply products and services unrelated to the AllianceCORE program that may be of value to you. These are profiled in tables under the *Areas of Technical Expertise* section for each partner. It is from this pool of products and knowledge that future AllianceCORE products will emerge.

Some partners have Xilinx versions of cores that have yet to go through the process to become AllianceCORE's. Other cores may never make this transition because a programmable logic implementation is not practical today. AllianceCORE products are clearly differentiated in separate *AllianceCORE Product Tables*. Each also has a complete datasheet in the AllianceCORE Products section of this databook.

February 8, 1998

CAST, Inc.

24 White Birch Drive
Pomona, NY 10970 USA
Phone: +1 914 354-4945
Fax: +1 914 354-0325
E-mail: info@cast-inc.com
URL: www.cast-inc.com

Overview

CAST focuses on maximizing the success of FPGA designs by supplying high-quality, high-value models for simulation and synthesis. CAST provides a total modeling solution for FPGA design by delivering and supporting accurate, reliable, and affordable VHDL models ready for use by designers worldwide. CAST supplies a variety of support programs, VHDL training programs, consulting services, and custom modeling services.

CAST's products includes:

- V-Custom Cores - optimized synthesizable cores developed in VHDL for FPGA and ASIC designs. Available cores include UARTs, DMA controllers, interrupt controllers, bus interfaces, microcontrollers, and processors.
- Standard Component VHDL Library - Synthesizable (SCVL-S) consisting of over 600 VHDL synthesizable models of generic and TTL functions. Developed for new FPGA designs, and for conversion of older TTL circuits into FPGAs.
- Pre-packaged libraries feature Standard Component VHDL Library (SCVL) series. Developed for system and device designers, SCVL includes simulation models for 3,000 memory devices and 2,500 standard ECL and TTL devices.
- V-Custom Models - VHDL simulation models for processors, complex functions, and specialized devices. Featuring in-stock models such as the Pentium processor and a variety of peripherals

AllianceCORE Products

Table 1 shows released AllianceCORE products available from CAST. Datasheets of these products are included in the AllianceCORE Products section of this databook. Contact CAST for pricing.

Additional Areas of Expertise

Table 2 shows CAST, Inc.'s areas of technical expertise. Xilinx and CAST are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact CAST, Inc. for information or availability.

Table 1: Released AllianceCORE Products

Function	Description	Datasheet
Base Level Functions		
C2910A	Microprogram Controller	3-129
Communication and Networking		
Viterbi Decoder	Viterbi Decoder	3-91

Table 2: CAST Expertise

Function	Description	Gates
Base Level Functions		
C_UART	Generic UART	650
C49410	Microprogram Controller	9,500
C29116A	16-Bit Microprocessor	
C6850	Asynchronous Communications Interface	1,300
C8051	Microcontroller	
C8251	USART	
C8254	Programmable Timer/Counter	
C8255	Parallel Interface Peripheral	
C8259	Programmable Interrupt Controller	

February 8, 1998

Comit Systems, Inc.

1250 Oakmead Parkway, Suite 210
Sunnyvale, CA 94088 USA
Phone: +1 408 988 7966
Fax: +1 408 988 2133
E-mail: preeth@comit.com
URL: www.comit.com

Overview

Comit System, Inc. is a systems engineering company offering expertise in the areas of FPGA and ASIC design and development.

Comit Systems hardware design services include:

- FPGA/ASIC designs
- PCI design and Xilinx PCI macro customization
- HDL (VHDL/Verilog) based designs
- Design verification (PLI, simulation, test vector generation)

To enhance the productivity of its customers, Comit offers a set of efficient libraries for HDL-based Xilinx designs. These modules are available as VHDL/Verilog netlists. In HDL design, they can be instantiated to get maximum efficiency. Using constraint files, optimum placement can also be achieved. Simulation models provided for functional simulation. Higher level macro-modules are also under development. The areas of development include bus models, communication and DSP cores.

Areas of Technical Expertise

Table 1 shows Comit Systems areas of technical expertise. Xilinx and Comit are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Comit for information or availability.

Table 1: Comit Systems Expertise

Function	Description	CLB's
Base Level Functions		
UART	16450 compatible	130
HDLC controller		200
Wallace Tree Multiplier	8 x8 Multiplier	120
4B/5B Encoder-decoder		9
PRBS Generator	Variable length (4-32)	2-33

CoreEl MicroSystems

46750 Fremont Blvd. #208
Fremont, CA 94538 USA
Phone: +1 510 770-2277
Fax: +1 510 770-2288
E-mail: sales@coreel.com
URL: www.coreel.com

Overview

CoreEl MicroSystems offers an extensive list of ATM "Core Cell" designs that have been specifically optimized for use in Xilinx FPGAs. The majority of the IP for the ATM cores comes from the development of a single-chip 155 Mbps Network Interface Card (NIC) and a two-chip ATM switch design.

AllianceCORE Products

Table 1 shows released AllianceCORE products available from CoreEl MicroSystems. Datasheets of these products are included in the AllianceCORE Products section of this databook. Contact CoreEl for pricing.

Additional Areas of Technical Expertise

Table 2 shows CoreEl Microsystem's areas of technical expertise. Xilinx and CoreEl are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact CoreEl for information or availability.

Table 1: Released AllianceCORE Products

Product	Functional Description	Datasheet
Communications and Networking		
Cell Assembler	Cell stream assembler Corecell	3-49
Cell Delineation	Cell delineation Corecell	3-53
CRC10	CRC10 generator and verifier modules	3-57
CRC32	CRC32 generator and verifier modules	3-61
UTOPIA Slave	UTOPIA Level 2 slave transmitter and receiver modules	3-85

Table 2: CoreEl MicroSystems Expertise

Product	Functional Description	Gates
Communications and Networking		
CS-1000	Gigabit Ethernet MAC	
CS-1001	Gigabit Ethernet 8B/10B	
CS-1002	Gigabit Ethernet Auto Negation	
CS-1000T	General purpose Gigabit Ethernet user programmable test bench	
CS-1003	Flow Control	
CS-1100	Fast 10/100 Ethernet Media Access Controller	
CS-1101	RMII (Reduced Media Independent Interface)	
TC-25	20 Mbps Transmission Convergence	
TC-622	622 Mbps Transmission Convergence	
TC-34	E3 Transmission Convergence	
TC-45	DS-3 Transmission Convergence	
TC-155	STS-3 Transmission Convergence	
CS-203	ATM Cell diagnostic generation/analysis	
CC-204	ATM Traffic Policer	

Product	Functional Description	Gates
CC-140	UTOPIA Master	
SAR-155	ATM Segmentation and Reassembly	
SAR-622	ATM Segmentation and Reassembly	
NIC-155	Single Chip 155 Mbps ATM Network Interface Controller	
CC-303	E1 Frame	
CC-305	E3 Framer	
CC-307	STS-1 Framer	
CC-309	STS-3 Framer	
CC-311	STS-12 Framer	
CC-315	DS3 PLCP Processor	
CC-316	SONET/SDH Pointer Tracking	15,000
CC-316	HDLC controller	
CC-317	Multi-channel HDLC controller	2,200

February 8, 1998

DeDris Embedded Algorithms BV

Marconistraat 34
4004 JM TIEL
THE NETHERLANDS
Phone: +31 344 63 28 38
Fax: +31 344 63 26 31
Email : info@dedris.nl
WWW : www.dedris.nl

Overview

DeDris Embedded Algorithms is specialized in the mapping of complex application-specific signal processing algorithms onto dedicated custom hardware architectures. DeDris bridges the gap between algorithm design and development in high-level languages (such as C, Matlab or Pseudo-code) and custom digital processing architectures. Extremely efficient implementations are gained through combined optimization on algorithmic, architectural and gate level. Implementations of DeDris' signal processing algorithms are named DSP Instant Components and are available in Synthesizable VHDL, various netlist formats and as hard-coded plug and play FPGA and ASIC macro blocks.

The advantages of DeDris' DSP Instant Components are:

- Direct implementation of complex advanced signal-processing blocks
- Compliance with standards and norms
- Significant reduction of time-to-market and design cost
- Very efficient implementation
- Possibility to create very compact multiple channel implementations
- Customer support on both hardware and algorithm level

In addition to DSP Instant Components, DeDris offers custom design and implementation of application-specific signal processing algorithms. Our mathematicians understand and optimize your processing needs. Together with DeDris' architecture and hardware-design engineers, they create fast and optimized solutions for your application. Turn-around times from algorithm to silicon in weeks and months rather than years are feasible when teaming with DeDris. DeDris also has in-depth knowledge of ITU and CCITT standards. Our application engineers can guide you through the possibilities. Together we select the best algorithms for your application, which we instantly implement in hardware for you.

DeDris has a strong expertise in speech-related technologies. Multiple speech-compression algorithms have been implemented offering toll quality speech compression, and bit rates from 32 kbit/sec down to 2.4 kbit/sec. In addition, DeDris has developed expertise in the fields of speech recognition and synthesis. A further area of expertise is telecommunication. Several popular blocks have been implemented by DeDris, which allow rapid development of complex telecommunication systems. DeDris is strongly extending its fields of expertise through in-house R&D and the formation of strategic alliances with scientific and industrial partners. Areas being developed are audio, video and vision, and radar.

Areas of Expertise

The following table of products demonstrates areas of DeDris technical expertise. Xilinx and Dedris are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Dedris for information or availability.

Table 1: DeDris Expertise

Product	Functional Description	Gates[1]	ROM (kbytes)	RAM (kbytes)
DSP Functions				
FFT	Fast-Fourier transform	--	--	--
Speech recognition	Template-based speaker-dependent isolated-word automatic speech recognition	9,000	1	--
Pink-noise generator	Low-ripple pink noise filter with filter characteristic of 0.08 dB per octave over the bandwidth 20Hz to 20kHz	4,000	0.1	0.1
Image processing	Several core image processing blocks	--	--	--
CCIR 656/601	Digital video converter: CCIR to raw-video data and vice versa	1,500	--	--
Triple Des	Triple data-encryption according to standard DES algorithm			
Speaker-independent speech recognition	Hidden-Markov-model based isolated-word speaker-independent speech recognition			
MPEG Audio	Ultra compact and low power MPEG audio encoder/decoder			
Speech synthesis	Phrase-concatenated speech synthesis			
Communications and Networking				
ADPCM	Full duplex ITU-T G.726 compliant 16, 24, 32 and 40 kbit/s speech-compression encoder/decoder. Operates at 4Mhz.	6,500	0.280	0.128
ADPCM-8	Full duplex 8 channel ITU-T G.726 compliant 16, 24, 32 and 40 kbit/s speech-compression encoder/decoder. Operates at 16Mhz.	13,000	0.280	0.512
G.723.1	Low bit-rate ITU compliant speech-compression at 6.3 kbit/s, can be combined with G.723.1A. Operates at 20Mhz.	18,000	22	2.3
G.723.1A	Silence compression scheme by voice activity detection and comfort-noise generation, compliant with Annex A of speech-compression standard CODEC G.723.1. Option for G.723.1 at no additional hardware cost.	--	--	--
G.729	Low bit-rate ITU-T compliant speech-compression codec at 8 kbit/s			
Echo canceler	High-performance Echo-cancellation and suppression processor. Operates at 4Mhz.	6,000	0.28	0.150
DTMF	Full-duplex DTMF transceiver. Operates at 2 Mhz.	4,000	0.28	0.150
Caller-ID	On-hook and off-hook caller line identification. Operates at 4Mhz.	6,000	0.28	0.150
Reed Solomon	Full-duplex Reed-Solomon codec	4,000	0.5	0.2
LU7	A 64 kbit/s data bearer service for DECT/ISDN gateways, performing automatic-repeat requests and RS forward-error control	5,000	0.6	0.7
V23 modem	ITU-T V23 compliant 1200 baud FSK asynchronous modem	6,000	0.28	0.15
MELP	Federal standard compliant speech-compression codec at 2.4 kbit/s			

Note:
1. All gate counts are application dependent and indicative only.

Digital Objects, Corporation

555 Mathilda Ave, Suite 110
Sunnyvale, CA 94086 USA
Phone: +1 408 720-7750
Fax: +1 408 720-7756
E-mail: sales@digitalobjects.com
URL: www.digitalobjects.com

Overview

Digital Objects Corporation, headquartered in Sunnyvale, California s a design and consulting firm employing highly qualified and experienced engineers. The company was formed in early 1997 to market low cost design services to major semiconductor suppliers - particularly ASIC/custom semiconductor suppliers. Simultaneously, the company is developing a line of proprietary IP Cores to market to standard cell ASIC/custom semiconductor suppliers and high-density FPGA users to achieve low cost. The Company has established a design center in Karachi, Pakistan where a large, well-educated pool of qualified computer and EE engineers exists and the engineering costs are much lower than in the US.

Digital Objects' engineering expertise ranges from ASIC/FPGA design to Verilog simulation, synthesis, physical implementation, cell library development and VLSI characterization. Its software development capabilities include C, C++ for Windows 95 and embedded applications. Both hardware and software expertise allow Digital Objects to provide complete system solutions.

Areas of Technical Expertise

Table 1 shows other areas of Digital Objects technical expertise. Xilinx and Digital Objects are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Digital Objects for information or availability.

Table 1: Digital Objects Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
CardBus	CardBus interface macro	

February 8, 1998

Eureka Technology, Inc.

4962 El Camino Real, #108
Los Altos, CA 94022 USA
Phone: +1 415 960-3800
Fax: +1 415 960-3805
E-mail: info@eurekatech.com

Overview

In today's world of fierce business competition and ever changing marketing needs, time-to-market is the key to the success of every development project. Rapid productization and fast prototyping with the use of FPGA and reusable macro functions is quickly becoming the preferred design methodology for successful products. Eureka Technology, Inc. helps you to stay in the forefront of this new design methodology by providing reusable synthesizable modules for PowerPC and PCI bus systems.

These models are designed to handle the complexity of their respective bus systems and to provide a simple but efficient interface to the users' internal design. Using these models dramatically reduces the design cost and the learning curve of PowerPC and PCI bus systems. All models are available in Verilog and VHDL source code format, or in netlist format targeted to specific technologies.

In addition to our standard products, Eureka Technology can also provide features designed to your specifications. Design customization to customer requirements is the key to our success.

Over the years, Eureka Technology, Inc. has provided successful products to many leading companies in the computer, communication, and semiconductor industries. Our goal is to provide state-of-the-art design models to our customers and to continue our success as a leading provider of reusable models.

Areas of Technical Expertise

Table 1 shows Eureka Technology's areas of technical expertise. Xilinx and Eureka are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Eureka for information or availability.

Table 1: Eureka Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
EC100	PCI Bus target	*
EC200	PCI bus master/target	*
EC400	PCI host bridge	*
EP100	PowerPC bus slave	*
EP200	PowerPC bus master	*
EP206	PowerPC bus master simulation model	*
EP300	PowerPC bus arbiter	*
EP400	PowerPC to PCI host bridge	*
Base-Level Functions		
EC500	SDRAM controller	*
EC600	Multi-channel DMA controller	*
REAL86	16-bit microcontroller core	*

* Implementations vary. Contact partner for information.

GV & Associates, Inc.

23540 Oriente Way
Ramona, CA 92065 USA
Phone: +1 760 789-7015
Fax: +1 760 789-7015
E-mail: gvander@sd.cts.com

Overview

GV & Associates, Inc. is an independent engineering and manufacturing organization with an array of engineering talent including digital, analog, RF, and software development. The company strives to offer our customers an innovative solution to various engineering needs in an honest and cost effective manner.

The company has extensive experience in the following areas:

- Digital Design - Error Correction, DSP Poly-Phase Filters, Xilinx FPGA Design
- Microprocessor Design - Embedded Controllers
- Software Design - C, C++, MS Window Program Development
- Analog Design - Active & Passive Filters, Amplifiers
- RF Design - Amplifiers, Mixers, Synthesizers

AllianceCORE Products

Table 1 shows released AllianceCORE products available from GV & Associates, Inc. Datasheets of these products are included in the AllianceCORE Products section of this databook. Contact GVA Inc. for pricing.

Table 1: Released AllianceCORE Products

Product	Description	Datasheet
Supporting Developing Tools		
GVA-100	DSP Prototyping Board	3-179

February 8, 1998

Innovative Semiconductors, Inc.

2570 West El Camino Real, Suite 205
Mountain View, CA 94040
Phone: +1 650 917-5925
Fax: +1 650 948-6220
URL: www.isi96.com
E-mail: sales@isi96.com

Overview

Innovative Semiconductors, Inc. was founded in 1992 by Nabil Takla to develop and manufacture synthesizableintellectual property cores.

The mission of the company is:

- To become the leading provider of Intellectual Property Cores
- To provide our customers with the best service
- To hire the best engineers in our industry and empower them to provide the best solutions to challenging problems

The company develops and manufactures synthesizable intellectual property cores, the building blocks for developing high-performance ICs and chip sets for the video and communications markets.

The company's products include cores that support the Universal Serial Bus (USB), IEEE-1394, and Video Interface Port (VIP) standards, and an Inverse Discrete Cosine Transform (IDCT) synthesizable block used to build decoders/encoders for MPEG-2, JPEG and other DCT, video compression-based, standards.

Areas of Expertise

Table 1 shows other areas of Innovative Semiconductors, Inc. technical expertise. Xilinx and ISI are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Innovative Semiconductors, Inc. for information or availability.

Table 1: Innovative Semiconductors, Inc. Expertise

Function	Description	Gates
Standard Bus Interfaces		
SL10 (VIP)	RTL Synthesizable, Video Interface Port Slave	
SL15 (VIP)	RTL Synthesizable, Video Interface Port Master	
SL750 (FireWire)	IEEE 1394 FireWire Link Layer Core. Synthesizable HDL block that meets IEEE 1394 Specifications requirements.	
SL100 (USB)	USB function core with interface to transceiver and simple 8-bit Bus Interface to application logic. Contains device configuration logic that simplifies firmware development. Supports Data bursting from and to the application.	
DSP Functions		
IDCT MPEG-2	Inverse Discrete Cosine Transfer core for MPEG-2, MPEG-1, JPEG, H261 and H263	

February 8, 1998

Integrated Silicon Systems, Ltd.

50 Malone Rd.
Belfast, BT9 5BS
Northern Ireland
Phone: +44 1232 664 664
Fax: +44 1232 669 664
E-mail: info@iss-dsp.com
URL: www.iss-dsp.com

Overview

Integrated Silicon Systems is a DSP intellectual property company specializing in synthesizable DSP products. ISS has an acknowledged worldwide competence in VLSI architectures for DSP and supplies highly optimized DSP circuit functions for ASIC, FPGA and CPLD implementations.

The key advantages of using ISS products are:

- Highly efficient silicon implementations after synthesis
- Dramatically reduced design times for rapid prototyping and fast time-to-market
- Very high performance - orders of magnitude greater than programmable DSP cores
- Characterized for the worlds principal fabrication processes

ISS's main products are:

- DSP system-level cores - A range of parameterized system level blocks (JPEG, ADPCM) for efficient ASIC, FPGA and CPLD implementation.
- DSP function-level cores - An extensive range of highly parameterized DSP function level blocks for efficient ASIC, FPGA and CPLD implementation.
- DSP ASIC library - A comprehensive library of over 170 parameterized, synthesizable DSP building blocks written in VHDL and are available in a number of fixed and floating point formats.

All blocks are fully parameterized, allowing the ultimate in design flexibility and design reuse. ISS's range of functions is optimized for Xilinx FPGAs, giving Xilinx customers a unique advantage in the design of complex FPGA-based systems.

AllianceCORE Products

Table 1 shows released AllianceCORE products available from ISS. Datasheets for these are included in the AllianceCORE Products section of the databook. Contact ISS for pricing.

Additional Areas of Technical Expertise

Table 2 shows other areas of ISS' technical expertise. Xilinx and ISS are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact ISS for information or availability.

Table 1: Released AllianceCORE Products

Product	Functional Description	Datasheet
DSP Functions		
Reed-Solomon	High performance R-S encoders	3-77
Reed-Solomon	High performance R-S decoders	3-71
Communications and Networking		
HDLC	HDLC Protocol Core	3-65

Table 2: ISS Expertise

Product	Functional Description	Gates
DSP Functions		
JPEG	High performance JPEG encoders and decoders	
ADPCM	Supports 6 ITU ADPCM standards with multi-channel capability	16,000[1]
FFT	High performance FFTs in 16, 64, and 256 point formats	50,000[1]
DCT	DCT, IDCT and combined DCT/IDCT blocks for 8x8 DCT operation	45,000[1]
FIR filter library	A comprehensive FIR filter capability to produce highly customized, high performance FIR filters	16,000[1]
IIR filter library	A comprehensive IIR filter library for custom IIR filters	12,000[1]
Adaptive filters	Including LMS filter and custom adaptive filtering	9,000[1]
Image processing library	High performance front-end image processing functions including edge detectors and image enhancement filters	5,000[1]
Rank order filters	1-D and 2-D variants including programmable filters and median filters	6,000[1]
Arithmetic operators	All of the following are available in fixed and floating point formats: - multipliers - dividers - square root operators - adders - data format converters	

Note:
1. Gate counts provided are for an example core from the particular group and are for reference purposes only. Contact ISS for more information.

LOGIC INNOVATIONS

Partner Profile

February 8, 1998

Logic Innovations, Inc.

6205 Lusk Boulevard
San Diego, CA 92121 USA
Phone: +1 619 455-7200
Fax: +1 619 455-7273
E-mail: info@logici.com
URL: www.logici.com

Overview

Logic Innovations, Inc. (LII) has been providing FPGA and ASIC models and hardware and software design services since 1986. LII provides services in the following areas:

- Turnkey hardware and software development
- ASIC prototyping using Xilinx FPGAs
- Set-top boxes for cable and satellite broadcast systems
- High speed digital, analog and integrated circuit design
- Production test systems for cable and satellite equipment
- Asynchronous Transfer Mode (ATM) logic designs in Xilinx FPGAs
- Embedded systems and real time operating system software design

LII's Intellectual Property (IP) Products currently include the PCI Bus Master/Target Model and ATM Building Blocks. LII has developed numerous storage, communications and portable computing peripherals that use the PCI Bus interface. In these projects the PCI Bus interface was developed using Hardware Design Language (HDL) to allow customization for different applications and porting to specific FPGAs and ASICs to provide full speed, zero wait state operation.

Similarly, LII has developed communications products such as cable modems that use the Asynchronous Transfer Mode (ATM) communications protocol. The ATM Building Blocks were designed as HDL for implementation in FPGAs to accommodate design changes and multiple product configurations. LII has now released these tried-and-proven PCI Bus and ATM HDL models for use by system designers who demand fast time-to-market, design flexibility and the ability to customize. These products are available as source code in VHDL or Verilog format, or object code in device-specific netlist format. Demand for the IP Products is rapidly increasing commensurate with gate capacities in ASICs and FPGAs, as designers seek to reduce development time and cost by using drop-in, standard logic modules.

Additional Areas of Technical Expertise

Table 1 shows Logic Innovations' areas of technical expertise. Xilinx and Logic Innovations are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Logic Innovations for information or availability.

Table 1: Logic Innovations' Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
PCI Bus 32-bit Master/Target	Rev. 2.1 interface in Verilog or VHDL. Supports full speed burst memory transfers (up to 132 MBps) with a simplified, synchronous local bus interface. 16-word by 32-bit transfer buffer of size 2K to 4K; address and data parity generation and checking, and PCI interrupt support. Includes a complete HDL test bench to verify compliance.	9,000
Communications and Networking		
Transmission Convergence Cell Inlet Model	Supports up to 50 Mbps serial input bit transfer rate and serial to parallel conversion; performs cell delineation which generates sequenced ATM cells to send to the internal ATM cell FIFO; HEC check and multi-cell receive FIFOing; UTOPIA Level 1/2 compatible.	4,000
Transmission Convergence Cell Outlet Model	Supports up to 50 Mbps serial output bit transfer rates, and parallel to serial conversion; provides HEC field generation and multi-cell transmit FIFOing; UTOPIA Level 1/2 compatible.	13,000

Product	Functional Description	Gates
UTOPIA Level 2 Interface Building Block	UTOPIA Level 2, v1.0 interface to parallel send/receive cell FIFOs; provides buffering for 3 ATM cells in receive and 3 in transmit direction; generic parallel interface for easy access to internal ATM cell FIFOs. Supports variable ATM cell length of up to 64 bytes, and both 8 and 16 bit data paths.	4,000
ATM Broadband Cell Delineation Building Block	Supports 1.544 Mbit/s to 155.53 Mbit/s. UTOPIA Level 2 ATM layer interface, a serial and parallel physical layer interface, and a microprocessor interface for configuration and monitoring functions. It includes multiple cell input and output FIFOing, and supports HEC generation. Provides scrambler logic for cell transmit and de-scramble logic for cell receive, cell rate decoupling and ATM cell delineation.	4,000

February 8, 1998

Memec Design Services

1819 S. Dobson Rd., Ste. 203
Mesa, Arizona 85202 USA
Phone: +1 888 360-9044
+1 602 491-4311
Fax: +1 602 491-4907
E-mail: info@memecdesign.com
URL: www.memecdesign.com

Overview

Memec Design Services (MDS) is a subsidiary of the Memec International Components Group -- one of the top ten electronics distributors in the world. MDS is part of Raab Karcher, AG, a $10 billion organization in Essen, Germany. At the top of MDS' financial stability is Veba, the fourth largest company in Germany.

MDS is dedicated to bringing you the best Xilinx design services available in the market today. MDS' design engineers have extensive programmable logic (PLD, CPLD, FPGA, ASIC) design experience (9+ years on average). We are equipped with the tools to complete the design on-time and to your specifications.

New Designs

- Project feasibility studies
- Xilinx development system selection
- Optimal Xilinx FPGA selection
- Logic design and capture
- Custom macro design
- High performance state machine design
- Time constraint design
- Partioning/mapping control
- Floorplanning/placement constraints
- Package pin assignment
- Timing simulation

Existing Designs

- Xilinx design review from FPGA design to system integration
- Xilinx troubleshooting
- Conversion of ASIC, FPGA, and CPLD designs to Xilinx
- Recapture of Xilinx FPGA designs from LCA or obsolete CAE tools
- Xilinx FPGA performance improvement
- Xilinx functional and timing simulation
- HardWire conversion

In today's competitive, fast-paced market environment, engineering managers are under the gun to get design projects finished on schedule. The lost revenue and market share associated with missing a product release date can be devastating. MDS understands this competitive environment. We can augment your design team's effort to get your design projects completed on time, improving both time-to-market and time-to-volume for your product. MDS is willing to co-op the development effort and license the design at a substantially reduced rate.

AllianceCORE Products

Table 1 shows released AllianceCORE products available from MDS. Datasheets for these are included in the AllianceCORE Products section of this databook. Contact MDS for pricing.

Additional Areas of Technical Expertise

Table 2 shows areas of MDS' technical expertise. Xilinx and MDS are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact MDS for information or availability.

Table 1: Released AllianceCORE Products

Product	Functional Description	Datasheet
Standard Bus Interfaces		
XF-TWSI	I^2C Two-Wire Serial Interface	3-9
Base Level Functions		
XF8250	Asynchronous Communication Element	3-137
XF8255	Programmable Peripheral Interface	3-151
XF8256	Multifunction Microprocessor Support Controller	3-155
XF8279	Keyboard Controller	3-163
XF9128	Video Terminal Logic Controller (VTLC)	3-167
Supporting Development Tools		
MDS FPGA Development Module	MDS FPGA Development Module	3-183

Table 2: Memec Design Services Expertise

Product	Functional Description	Gates
DSP Functions		
XFRS Encoder	100 Mbit Reed-Solomon Encoder with Interleaver	
Base Level Functions		
XF-SCC	Synchronous Communications Controller (includes SDLC/HDLC controllers	

February 8, 1998

Mentor Graphics Corporation

1001 Ridder Park Drive
San Jose, CA. 95131-2314 USA
URL: www.mentorg.com/inventra

Overview

Inventra™, a Mentor Graphics business unit, offers an extensive list of synthesizable standard component, communications, multimedia, telecom, DSP, and computer connectivity cores to meet current and emerging design methodologies such as block-based design. Inventra also offers physical libraries and design services to enable designers to meet the challenges of increased design complexity and faster time-to-market. All Inventra products support EDA Tool, foundry and derivative design independence thereby offering the maximum degree of flexibility without the time and expense of internal development.

The computer connectivity cores and products (Universal Serial Bus, PCI, Simulation Models, and Evaluation Board) are available as a result of the acquisition of CAE Technology in November 1996. Inventra's Universal Serial Bus (USB) is a full hardware/software solution that is comprised of:

- Synthesizable Logic Blocks (Verilog & VHDL)
- Simulation Models
- Evaluation Board
- Virtual Simulation Framework (VSF)

The VSF is a hardware / software co-design solution. It provides a seamless link between the software and hardware teams during the ASIC and system design phases.

AllianceCORE Products

Table 1 shows released AllianceCORE products available from Inventra. Datasheets of these products are included in the AllianceCORE Products section of this databook. Contact your local Mentor Graphics sales office for pricing.

Table 1: Released AllianceCORE Products

Product	Description	Datasheet
AllianceCORE Modules - Standard Bus Interface		
Low-Speed USB Function Controller	1.5 Mbps function controller with microcontroller interface.	3-29
Full-Speed USB Function Controller	12 Mbps function controller with support for microcontroller and generic back-end interfaces.	3-33
3-Port USB Hub Controller	Xilinx 3-port USB hub controller	3-37
Supporting Development Tools		
USB Function Board	A Xilinx FPGA-based prototyping board for developing USB-complaint peripherals.	3-41
USB Hub Evaluation Board	A Xilinx FPGA-based demo board for developing USB-complaint hubs.	3-43
USB Simulation Model	Simulates full Host, Hub, and Target Functions.	3-45

Additional Areas of Technical Expertise

Table 2 shows other areas of Inventra's offering. Xilinx and Inventra are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you would like product, pricing or availability information for a specific product in this listing. Contact Inventra for pricing.

Table 2: Inventra Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
MPCMCIA1	PCMCIA PC Card Interface	2,500
M82365SL	PCMCIA Card Interface Controller	17,000
M8490	SCSI for 5380 Compatible Asynchronous SCSI Interfacing	1,500
M1284H	IEEE 1284 Host Parallel Port	6,500
MI2C	I2C Bus Interface	1,500
32 Bit PCI	32 Bit PCI Core	10,000
64 Bit PCI	64 Bit PCI Core	13,000
PCI Simulation	PCI Simulation Model	N/A
DSP Functions		
DDS	Direct Digital Synthesizer	Note 1
M320C50	Very High Performance Digital Signal Processor	40,000
M320C25	High Performance Digital Signal Processor	25,000
FFT / IFFT	FFT / IFFT / Fast Convolver	Note 1
FIR	Linear Phase FIR Filter	Note 1
NCO	Numerically Controlled Oscillator	Note 1
DCT	8x8 Discrete Cosine Transform (DCT)	Note 1
Communications and Networking		
ADPCM-G726	ADPCM CODEC, Full Duplex-1 channel	Note 1
ADPCM-G726	ADPCM CODEC, Full Duplex-2 channel	Note 1
ADPCM-G726	ADPCM CODEC, Full Duplex-4 channel	Note 1
BRIM-CORE	I.430 Basic Rate Interface Module (ISDN S-Interface)	Note 1
E1-CORE	E1 Synchronous Framing Structure Core	Note 1
T1-CORE	T1 Synchronous Framing Structure Core	Note 1
Viterbi	Viterbi Encoder / Decoder	Note 1
Reed-Solomon	Reed-Solomon / BCH Decoder	Note 1
RS-IESS308	Intelsat IESS-308 Reed Solomon CODEC with Data Interleaver/Deinterleaver	Note 1
RS CODEC	Reed Solomon CODEC	Note 1
RS-DVB	DVB Compliant Reed Solomon CODEC	Note 1
RS-CDPD	CDPD Compliant Reed Solomon CODEC	Note 1
LILAC	Linked List Access Controller (LILAC)	Note 1
REO	T1 / E1 Timeslot Recorder	Note 1
X.50	X.50 Multiplexing Interface	Note 1
HDLC-CORE	Single Channel HDLC Controller	Note 1
M79C960	Ethernet LAN Controller ISA Interface	15,000
ARCNET	ANSI / ATA 878.1 ARCNET Token Bus LAN	Note 1
Base-Level Functions		
MFDC	High Performance PC Compatible Floppy Disk Controller System (82077SL)	10,500
M765A	Extended Features Floppy Disk Controller Core for FM and MFM Formats	9,500
MDDS24	Enhanced High Margin Floppy & Tape Data Separator for data rates to 2Mbit/s	1,300

Product	Functional Description	Gates
M91C360	High Margin Floppy & Tape Data Separator for data rates to 1.25 Mbit/s	1,300
M91C36	High Margin Floppy Disk Data Separator for data rates to 1.25 Mbit/s	1,100
M80186	Industry Standard 16-Bit Microprocessor	30,000
M8086	Industry Standard 16-Bit Microprocessor	18,000
M8052	High Performance Industry Compatible 8-Bit Microcontroller, 3 timers, serial I/O	11,000
M8051	High Performance Industry Compatible 8-Bit Microcontroller, 2 timers, serial I/O	10,000
M8042	8-Bit Peripheral Interface Microcontroller with timer (Slave Microcontroller)	3,400
M8048	Compact Embedded Industry Compatible 8-Bit Microcontroller with Timer	3,400
MZ80	High Performance Industry Compatible 8-Bit Microprocessor	8,000
DMAxN	Multi Channel DMA Controller	Note 1
M82C206	Integrated Peripheral Controller	18,000
M8237A	General Purpose programmable four channel DMA Controller	3,700
M8254	Extended Features three channel Programmable Interval Timer (PIT)	3,000
M8253	General Purpose three channel Programmable Interval Timer (PIT)	3,000
M6845	General Purpose Programmable CRT Controller	2,600
M146818	Ultra Low Power Real Time Clock with up to 114 bytes of RAM	2,000
M8259A	Eight Channel Cascadable Programmable Interrupt Controller (PIC)	1,600
M8255	General Purpose Programmable Peripheral Interface	1,600
USART	Universal Synchronous / Asynchronous Receiver / Transmitter Core	Note 1
M85C30	Two Channel Enhanced Serial Communications Controller with FIFOs	16,500
M82530	Two Channel Advanced Serial Communications Controller	12,200
M16550A	Universal Asynchronous Receiver / Transmitter (UART) with FIFO	6,600
M8251A	Universal Synchronous / Asynchronous Receiver / Transmitter (USART)	2,200
M16C450	PC Compatible Universal Asynchronous Receiver / Transmitter (UART)	2,000
M8250B	PC Compatible Universal Asynchronous Receiver / Transmitter (UART)	2,000
M8868A	Compact Universal Asynchronous Receiver / Transmitter (UART)	760
M6402	Compact Universal Asynchronous Receiver / Transmitter (UART)	750
FISPbus	68000 MPM Micro Personality Module	Note 1
FISPbus	H8 Micro Personality Module	Note 1
FISPbus	SH Micro Personality Module	Note 1
FISPbus	ARM 7 Micro Personality Module	Note 1
FISPbus	8051 Micro Personality Module	Note 1

Note:
1. Gate counts for these cores are application dependent. Contact Partner for more information.

February 8, 1998

Mobile Media Research, Inc.

39675 Cedar Boulevard, #295A
Newark, CA 94560 USA
Phone: +1 800 799-MMRI (6674)
+1 510 657-4891
Fax: +1 510 657-4892
E-mail: sales@mobmedres.com
URL: www.mobmedres.com

Overview

Mobile Media Research was founded in 1991 with the mission to provide complete solutions and enabling technology for the rapidly growing mobile computing market and specifically the PCMCIA marketplace. Today, Mobile Media Research is the premier supplier of PC Card development tools and testing systems that drastically reduce time-to-market and increase productivity for Card and Host developers. Simultaneously, Mobile Media Research is developing a line of IP Cores for FPGA-based designs to enable designers to meet the challenges of increased design complexity and faster time-to-market. The company also offers low cost design and consulting services in the areas of computer connectivity standards, in particular PCMCIA and CardBus, Multimedia, communications, telecom, and networking. Mobile Media Research will be offering cores for ASIC-based designs in the future.

The SocketWare™ product line offers the best solution for 16-bit PC Card product development and system testing. It is a complete family of unique software and hardware products that provide design solutions for the PCMCIA engineering community. Both hardware and software development tools are designed to accelerate design and debug cycles, thus reducing time-to-market of PCMCIA products.

Mobile Media Research has recently introduced CBWare™, a complete line of CardBus development and test tools for the design of 32-bit CardBus hosts, cards, and applications. The CardBus product line is aimed at providing development solutions and building blocks that, like all Mobile Media Research products will reduce development time immensely and drastically cut time-to-market.

The IP Core family of SoftLogic™ is a diverse line of reusable synthesizable macro functions designed specifically for FPGAs. The cores are customizable and can be used directly in the target device. Each core comes with all the necessary documentation and test bench support to ensure a quick and smooth integration into customer designs. Each core is verified for timing and functionality. The macro functions of the SoftLogic family are supported with optional prototyping boards, test software, and test bench modules.

AllianceCORE Products

Table 1 shows AllianceCORE released products from Mobile Media Research Inc. Datasheets for these are included in the AllianceCORE Products section of this databook. Contact Mobile Media Research for pricing.

Table 1: Released AllianceCORE Products

Product	Description	Datasheet
AllianceCORE Modules - Standard Bus Interfaces		
Xilinx Fax/Modem Macro	A PCMCIA fax/modem interface for implementing a complete fax/modem card using an external fax/modem chipset.	3-15
Xilinx PCMCIA Library R1.2	A comprehensive library for developing PCMCIA compliant hardware products with Xilinx FPGAs.	3-19
Supporting Development Tools		
PCMCIA Prototyping Card	A Xilinx-based configurable development platform for designing and debugging PC-Cards.	3-23
PCMCIA Card Debugger/Exerciser	A fully-compatible, Microsoft Windows based PCMCIA debugger and exerciser for hardware and software development.	3-25
CIS Generator 1.2	This is a feature-rich, Windows-based compiler that automates the generation of a PCMCIA-compatible CIS.	3-27

Additional Areas of Technical Expertise

Table 2 shows other areas of Mobile Media Research technical expertise. Xilinx and Mobile Media Research are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact MMRI for information or availability.

Table 2: Mobile Media Research Inc. Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
XFlash	A complete Xilinx PCMCIA interface macro for Intel Series 2 compatible Flash memory cards.	
HP-Pro Series Logic Analyzer/ Extender Card	A highly sophisticated and versatile PCMCIA development tool.	N/A
Type Converter Card	A uniquely designed PCMCIA type converter/socket saver that expands the mechanical capacity of host sockets.	N/A
CardTRAC	A high-value, single slot AT/PCMCIA adapter that includes card & socket services and provides an ISA-based PCMCIA and JEIDA 4.1 platform.	N/A
PCIC Software Functions Library	An extensive library for developing PCMCIA compliant software products.	N/A
CBWare™ Development Tools	A complete family of products for CardBus product development.	N/A
CB Macro	CardBus interface macro.	N/A
CBWareTM Development Tools	A complete family of products for CardBus product development	N/A
Base-Level Functions		
SDRAM Controller		N/A

February 8, 1998

NMI Electronics Ltd.

Fountain House
Great Cornbow
Halesowen
West Midlands
B63 3BL, UK
Phone: +44 (0) 121 585 5979
Fax: +44 (0) 121 585 5764
E-mail: ip@nmi.co.uk
URL: www.nmi.co.uk

Overview

NMI is an independent design services company and is now making its expertise available in the form of Intellectual Property cores via the Xilinx AllianceCORE program. These cores leverage NMI's skills, particularly in the area of high speed digital design, to provide commonly required and easily customizable functions which are instantly usable.

NMI cores mainly target the XC9500 family and will be available in optimized HDL source code. This allows the cores to provide the flexibility which many applications demand. First time users will find that NMI's cores accelerate their learning process, giving them the confidence to rapidly move forward.

NMI also offers full product design services to customers. These range from modifying AllianceCORE offerings, through providing custom cores, to full core integration, system design and prototyping.

NMI designs systems with diverse requirements: low power; high performance; low cost; ease of manufacture. We also have in-depth experience of PC hardware and software development, including planar boards, PCI and PCMCIA.

NMI is dedicated to providing successful solutions to its customers' product design needs. Incorporated in 1987, NMI has very strong systems design skills, coupled with proven project management abilities. These ensure that product developments are delivered on time and to budget.

Areas of Technical Expertise

Table 1 lists areas of NMI Electronics' technical expertise. Xilinx and NMI are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact NMI for information or availability.

Additional Areas of Technical Expertise

Table 2 demonstrates other areas of NMI technical expertise. Xilinx and NMI are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact NMI for information or availability.

Table 1: Released AllianceCORE Products

Product	Functional Description	Datasheet
Base-Level Functions		
DRAM Controller	DRAM Controller for XC9500	3-173
Supporting Development Tools		
Prototyping Board	Core Prototyping Board for Xilinx FPGAs and CPLDs	3-185

Table 2: NMI Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
I2C	I2C Slave Interface for XC9500	
Base-Level Functions		
DRAM Controller	DRAM Controller for the XC4000	

February 8, 1998

Perigee, LLC

314 Second Street
Liverpool, New York 13088 USA
Phone +1 315 453-7842
Fax: +1 315 453-7917
E-mail: info@PerigeeLLC.com
URL: www.PerigeeLLC.com

Overview

Perigee, LLC is a technology company specializing in new product development. Perigee's focus is electronic system design, partitioning and development. We are committed to leveraging the ever-growing size of programmable logic without sacrificing time-to-market. To this goal, Perigee offers IP cores as well as customization, FPGA/CPLD design and integration services. Our design solutions range from proof of concept designs through final product solutions.

Perigee offers services and contract engineering for varying levels of development from logic design to system level embedded and bus-based products. These services include:

- IC design, VHDL or schematic based
- turnkey hardware/software/firmware development
- system design
- board design
- DSP and image processing
- Embedded firmware
- GUI and Application software

Areas of Expertise

The following table of products demonstrates areas of Perigee technical expertise. Xilinx and Perigee are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Perigee for information or availability.

Table 1: Perigee Expertise

Product	Description	Gates
DSP Functions		
YCrCb to RGB	YCrCb to RGB color space converter	4,000
RGB to YCrCb	RGB to YCrCb color space converter	6,000
	Gamma Corrector	2,000
	Inverse Gamma Corrector	2,000
	Active Matrix LCD controller	2-5,000
	2-D Video Convolver	

February 8, 1998

Phoenix Technologies/Virtual Chips

411 E. Plumeria Drive
San Jose, CA 95134 USA
Phone: +1 408 570-1000
Fax: +1 408 570-1230
E-mail: virtualchips-info@phoenix.com
URL: www.phoenix.com

Overview

The Virtual Chips division of Phoenix Technologies Ltd., is the world's leading supplier of synthesizable cores and test environments for computer connectivity standards including PCI, USB, CardBus, PCMCIA and AGP.

Using Virtual Chips products, design teams are able to bring leading edge, high performance ICs to market faster, with assured compliance to computer connectivity standards. Each Virtual Chips product includes all the design features, documentation guidance and support necessary to ensure fast, smooth integration into customer designs and optimal results in silicon.

Phoenix Technologies Ltd. is headquartered in San Jose, CA, with sales, support and R&D facilities in Norwood, MA, Irvine, CA, Japan, Taiwan, France, and the U.K.

Areas of Technical Expertise

Table 1 shows areas of Phoenix Technologies Ltd. technical expertise. Xilinx and Phoenix are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Phoenix Technologies Ltd. for information or availability.

Table 1: Phoenix Technology Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
	Peripheral Component Interconnect (PCI)	7,000 + FIFO's *
	Universal Serial Bus (USB)	6,000 *
	Accelerated Graphics Port (AGP)	10,000 *
	CardBus Test Environment	N/A
	PCMCIA Test Environment	N/A

* Exact gate count depends on product configuration.

RICE

Partner Profile

February 8, 1998

Rice Electronics

PO Box 741
Florissant, MO 63032 USA
Phone: +1 314 838-2942
Fax: +1 314 838-2942
E-mail: ricedsp@aol.com

Overview

Rice Electronics provides cores for FPGA-based designs. The cores range from multipliers to Discrete Fourier Transforms (DFTs), and are targeted at the Xilinx XC4000E logic family.

Rice Electronics employs a variety of proprietary techniques to reduce DSP circuit complexity. These range from specialized architectures, to highly efficient algorithms. This results in cores which rival the performance of standard DSP components.

The XACTstep 6.0 development system provides the environment for application of the cores. The current focus is on DFT/FFT functions, with additional cores being planned.

Additional Areas of Technical Expertise

Table 1 shows areas of Rice Electronics' technical expertise. Xilinx and Rice are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Rice for information or availability.

Table 1: Rice Technology Expertise

Product	Functional Description	Gates
DSP Functions		
FFTs and DFTs	DSP Modules for Xilinx XC4000 Family Devices	

February 8, 1998

Sand Microelectronics

2121 Ringwood Drive
Milpitas, CA 95131 USA
Phone: +1 408 235-8600
Fax: +1 408 235-6801
E-mail: sales@sandmicro.com
URL: www.sandmicro.com

Overview

Sand Microelectronics provides productivity tools for FPGA and ASIC designers, including Verilog/VHDL simulation models and synthesizable cores, enabling designers to reduce product development time and improve design quality.

Sand Microelectronics' cores are fully synthesizable and silicon-proven. Each core can easily be migrated to almost any ASIC/FPGA technology in a relatively short period of time. Using Sand's RapidScript Utility™, the cores can be quickly configured to match the needs of the user's application. By utilizing Sand's tested cores, designers can significantly reduce development time and engineering risk and bring their products to market faster.

Sand Microelectronic's PCI Synthesizable Core is a set of synthesizable building blocks that ASIC and FPGA designers can use to implement a complete PCI interface. The PCI Core contains all the necessary hooks for customizing and optimizing it for a specific application.

Sand's also provides Universal Serial Bus (USB), Hub and Host controller cores that are fully synthesizable and intergratable in FPGAs or ASICs. Sand also supplies bus functional simulation models and a USB Transaction Analysis tool for USB verification.

Additional Areas of Technical Expertise

Table 1 shows areas of Sand Microelectronics' technical expertise. Xilinx and Sand are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Sand for information or availability.

Table 1: Sand Microelectronics Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
PCI 32bit/33 MHz PCI 64bit/33 MHz PCI 32bit/ 66MHz PCI 64bit/66 MHz	Systhesizable PCI cores, fully Revision 2.1 compliant; bridges PCI to high performance F-Bus (FIFO Bus). Design is synchronous to PCI clock, however, F-Bus interface to a specific peripheral need not be synchronous. RapidScript Utility™ configurable to match an application's interface requirements.	
USB Device Controller (UDC)	General purpose solution for USB peripheral devices. Quickly configured for any number of End-Points using RapidScript. Does not require a microcontroller or firmware.	5,000
USB Hub Controller	RapidScript configurable, does not require a microcontroller or firmware.	

February 8, 1998

SICAN Microelectronics Corp.

400 Oyster Point Blvd., Suite 512
South San Francisco, CA 94080 USA
Phone: +1 650 871-1494
Fax: +1 650 871-1504
E-mail: info@sican-micro.com
URL: www.sican-micro.com

SICAN GmbH
Garbsener Landstrasse 10
Hannover
D-30419, Germany
Phone: +(49)511-277-1601
Fax: +(49)511-277-2600
URL: www.sican.de

Overview

SICAN GmbH, headquartered in Hannover, Germany, is a microelectronics design company specializing in communications, multimedia and networking applications. Its technical staff and infrastructure form an organization with the flexibility and innovation to meet the rapidly changing design needs of its customers.

As a full-service design house, SICAN functions like other American design houses, offering IC development services. The company has expertise in classical ASIC development using gate array and standard cell architectures. It has over 200 designs which have been implemented in various companies and acts as an authorized design center for several world-class ASIC vendors.

SICAN has developed advanced core expertise in topics such as MPEG video and audio encoding and decoding, high-speed networking, broadband media access, and mixed-signal design. SICAN's intellectual property is aggregated in cores called DesignObjects™, are licensed to companies creating systems-on-silicon. It also has a staff of over 300 employees, making it one of the largest independent design houses which licenses its intellectual property. This impressive design force allows SICAN to offer the highest quality, customer specific designs as well as reduce their time-to-market.

Areas of Technical Expertise

Table 1 shows SICAN Microelectronics areas of technical expertise. Xilinx and SICAN are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact SICAN for information or availability.

Table 1: SICAN Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
CAN Bus Module	CAN Bus interface module.	
IIC Slave Interface	IIC Slave Interface	
DSP Functions		
DES	Data Encryption Standard (DES)	
Communications and Networking		
HDLC	HDLC	
	Reed-Solomon Decoder	
UTOPIA	UTOPIA Interface	
Base-Level Functions		
	Pulse-Width Modulator	

February 8, 1998

T7L Technology, Inc.

220 Duncanmill Road
Suite 307
North York, Ontario
Canada, M3B 3J5
Phone: +1 416-445-4783
Fax: +1 416-445-5741
E-mail: t7linfo@t7l.com
URL: www.t7l.com

Overview

T7L Technology is a leading provider of VHDL based synthesizable processor cores for FPGA and ASIC customers. The main advantage of using T7L products are product time-to-market and design optimization. With these cores, users can significantly reduce product design cycles and gain market leadership advantages.

Design optimization also ensures that real time performance can be achieved. High optimization in silicon area will minimize power consumption. With large gate count FPGAs available today, embedded CPU cores becomes an essential element for today's applications.

Based on its proprietary scaleable RISC architecture, T7L has built a family of 26 CPU cores optimized for Xilinx XC4000 series FPGA devices. It is built around a 3-stage pipeline engine with separated instruction and data buses. A flexible and selectable instruction set can be extracted from a supplied master function library. Parameter-driven configurable design allows fast and easy customization. This is the first RISC CPU core dedicated for FPGA application.

Areas of Technical Expertise

Table 1 lists areas of T7L Technology's technical expertise. Xilinx and T7L are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact T7L Technology for information or availability.

Table 1: Released AllianceCORE Products

Product	Functional Description	Datasheet
Base Level Functions		
TX400	FPGA optimized RISC processor core family	3-97
Design Base Board	Xilinx FPGA-based hardware development board for TX400 RISC CPU cores	3-103
SDP Integrated Software	Complete Windows '95 based integrated development environment for TX400 CPU cores	3-107

February 8, 1998

Technology Rendezvous Inc.

3160 De La Cruz Blvd. #101
Santa Clara, CA 95054 USA
Phone: +1 408 566-0280
Fax: +1 408 566-0284
E-mail: info@trinc.com
URL: www.trinc.com

Overview

Technology Rendezvous Inc. (TRI) is focused on the embedded systems, bus interface and networking markets. TRI provides time and cost effective system solutions (hardware and software), design services and system level macros including synthesizable logic cores (softcores) and verification models to FPGA/ASIC designers. Our professional services division provides customized solutions in the design and development of software and hardware.

TRI believes in providing complete solutions for system designs. Apart from supplying generic Logic Cores, TRI also provides additional tools such as drivers, system software for Embedded Systems and prototyping systems. These solutions help the designers to rapidly integrate the core into their ASIC designs, perform system debug in a real-world environment, and then quickly convert the prototype to a production unit. In the area of Embedded Systems TRI has expertise in providing systems solutions, BSPs and drivers for the embedded systems running real time OS like pSOS, VxWorks etc.

TRIs IEEE-1394 (Firewire) Synthesizable Core is a building block that ASIC and FPGA designers can use to implement a complete 1394 interface. In addition to the core, TRI is currently developing an FPGA based 1394 host and peripheral emulation system. This emulation system is an ideal platform for 1394 developers to prototype their hardware, firmware and software stack.

Areas of Technical Expertise

Table 1 shows areas of Technology Rendezvous Inc. technical expertise. Xilinx and TRI are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Technology Rendezvous Inc. for information or availability.

Table 1: Technology Rendezvous Inc. Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
FireWire Core	The core has two components: Link Layer & OHCI Layer; fully compliant with IEEE-1394-1995bus standard specifications; contains all necessary hooks for customizing and optimizing it for a specific application	

February 8, 1998

VAutomation

20 Trafalgar Square Suite 443 (4th Floor)
Nashua NH 03063 USA
Phone: +1 603 882-2282
Fax: +1 603 882-1587
E-mail: sales@vautomation.com
URL: www.vautomation.com

Overview

VAutomation was one of the first companies to recognize the need for re-usable intellectual property (IP) – or cores – when designing with ASICs and FPGAs. Since 1994, it has provided synthesizable HDL-based cores for microprocessors and serial communications controllers. VAutomation pre-designed and verified cores allow microprocessors and serial communications controllers, the 8086 and Universal Serial Bus for example, to be implemented within an ASIC or FPGA for true system-on-chip design.

Our cores are strictly synchronous making them easy to synthesize to any technology. Synthesis and timing analysis are simple, push-button operations with the sample scripts provided with each core. A comprehensive testbench is also provided for design verification and functional test vector extraction.

All of our cores have been thoroughly verified in silicon using Xilinx FPGA technology. Evaluation silicon is available for evaluating our cores and to enable early software/hardware integration. The V8-uRISC 8-bit RISC microprocessor and VUSB Universal Serial Bus Microcontroller cores are supported via the IntelliCore Prototyping System. This system, based on one to four XC4028 FPGAS, allows our cores and user logic to be easily prototyped and debugged.

VAutomation is exclusively focused on providing synthesizable cores. We market and sell our products through a direct sales and support operation headquartered in Nashua, N.H. The company is privately held with seed funding provided by its founders.

Areas of Technical Expertise

Table 1 lists areas of VAutomation's technical expertise. Xilinx and VAutomation are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact VAutomation for information or availability.

Additional Areas of Technical Expertise

Table 2 shows areas of VAutomation's technical expertise. Xilinx and VAutomation are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact VAutomation for information or availability.

Table 1: Released AllianceCORE Products

Product	Functional Description	Datasheet
Base Level Functions		
V8 uRISC	8-bit FPGA-based RISC processor	3-109
Supporting Development Tools		
Proto Board	IntelliCore™ prototyping System - Xilinx FPGA based prototype and core evaluation board for the V8 uRISC and other VAutomation cores.	3-115

Table 2: VAutomation Expertise

Product	Functional Description	Gates
Standard Bus Interfaces		
VUSB	USB Device Microcontroller	8,000
Communications and Networking		
V960	AM79C960 10 Mbps Ethernet LAN	17,000
V526	82526 HDLC controller	5,000
Base Level Functions		
V6502	65C02 8-bit microprocessor	4,000
VZ80	Z80 8-bit microprocessor	8,000
V8086	I8086 16-bit microprocessor	18,000
V186	I80186 16-bit microcontroller	28,000

February 8, 1998

Virtual IP Group, Inc.

1094 E. Duane Ave., Suite 211
Sunnyvale, CA 94086 USA
Phone: +1 408 733-3344
Fax: +1 408 733-9922
E-mail: sales@virtualipgroup.com
URL : www.virtualipgroup.com

Overview

Virtual IP Group, Inc. is a leading provider of Verilog HDL synthesizable industry-standard cores for FPGAs and ASICs.

Virtual IP Group is equipped with state-of-the-art hardware and software, and over thirty talented engineers with extensive backgrounds in system and ASIC design. This allows Virtual IP to offer quality products with superb documentation in time to meet critical product schedules. All cores are clean room designed and proven in silicon.

Virtual IP Group offers synthesizable cores for microcontrollers (8031 and PIC family), telecommunication (E1/T1 framers, muxs etc.), datacom, PC peripherals and bus interface standards. Virtual IP Group also provides ASIC services including custom library creation. Virtual IP Group has optimized some of these cores for use in Xilinx FPGAs. Virtual IP Group will also be offering DSP and analog cores in the future. Contact Virtual IP Group directly for the latest product offering.

AllianceCORE Products

Table 1 shows AllianceCORE products available from the Virtual IP Group. Datasheets for these are included in the AllianceCORE Products section of this databook. Contact Virtual IP Group for pricing

Additional Areas of Technical Expertise

Table 2 demonstrates other areas of Virtual IP Group technical expertise. Xilinx and Virtual IP Group are in the process of evaluating these to determine which are suitable as AllianceCORE products. If you have a need for a specific product then contact Virtual IP Group for information or availability.

Table 1: Released AllianceCORE Products

Product	Description	Datasheet
Communications and Networking		
MT1F	T1 Framer	3-81
Base Level Functions		
M8237	Programmable DMA Controller	3-133
M8254	Programmable Interval Timer	3-143
M8255	Programmable Peripheral Interface	3-147
M8259	Programmable Interrupt Controller	3-159
M16450	UART	3-121
M16550A	UART + FIFO	3-125

Table 2: Virtual IP Group Expertise

Product	Description	Gates
Standard Bus Interfaces		
M6730	PCI-PCMCIA controller bridge	24,000
M82365	PCMCIA host socket controller	11,000
Communications and Networking		
ME1BP	E1 Tx/Rx Baseband Processor	2,300
ME2BP	E2 Tx/Rx Baseband Processor	2,200
ME3BP	E3 Tx/Rx Baseband Processor	2,200
ME1CSC	CSC for E1/DS1	
ME2FRM	E2 framer	2,500
ME3FRM	E3 framer	2,800
ME1CCS	E1 digital cross-connect switch	
ME1CONC	E1 digital concentrator	
ME1ADMUX	E1 add/drop multiplexer	
M2BY8	E2 multiplexer/demultiplexer	3,000
M8BY34	4-E2 to 1-E3 multiplexer/demultiplexer	4,300
M34BY140	4-E3 to 1-E4 multiplexer/demultiplexer	4,300
Base-Level Functions		
EPP/ECPP	Extended parallel port/enhanced capability parallel port	
M8031	8-bit microcontroller w/128-byte RAM	7,400[1]
M8031TBO	8-bit microcontroller w/128-byte RAM, 4 cyc/instruction	7,400[1]
M8032	8-bit microcontroller w/256-byte RAM	8,400[1]
M8032TBO	8-bit microcontroller (256 Byte RAM, 4 cyc/instruction)	11,000[1]
M8042	Keyboard controller (host)	3,400[2]
M8048	Keyboard controller (device, 2K addressing)	3,300[2]
M8049	Keyboard controller (device, 4K addressing)	3,500[2]
M8051	8-bit microcontroller w/128-byte RAM	7,600[2]
M8051TBO	8-bit microcontroller w/128-byte RAM, 4 cyc/instruction	11,000[2]
M8530	2-channel serial communication controller	13,000
M146818	Programmable real-time clock	2,800
PIC1674	PIC 8-bit microcontroller	7,600[2]

Notes:
1. Gate count shown does not include RAM.
2. Gate count shown does not include RAM and ROM.

February 8, 1998

LogiBLOX

Xilinx, Inc.

2100 Logic Drive
San Jose, CA 95124
Phone +1 408-559-7778
Fax: +1 408-559-7114
URL www.xilinx.com

Overview

LogiBLOX is a graphical interactive tool for creating modules, such as counters, shift registers, and multiplexers. LogiBLOX includes both a library of generic modules and a set of tools for customizing them.

The modules created by LogiBLOX can be used in designs generated with schematic editors from Aldec, Viewlogic, Mentor Graphics, and Cadence, as well as third-party synthesis tools such as Synopsys FPGA Compiler, Xilinx Foundation Package, and Mentor Autologic.

LogiBLOX modules are used whenever you need a customized version of a standard function. For example, with a LogiBLOX counter, you start with a generic template for a counter and tailor its functionality to your needs. In contrast, a standard ready-made counter has a previously defined set of functions and in the case where you want a counter with specific capabilities, you would need to have available a library of different counters, one of which contains the functions you need.

Simulation of LogiBLOX modules can be performed using behavioral VHDL, Structural Verilog or Gate level EDIF models.

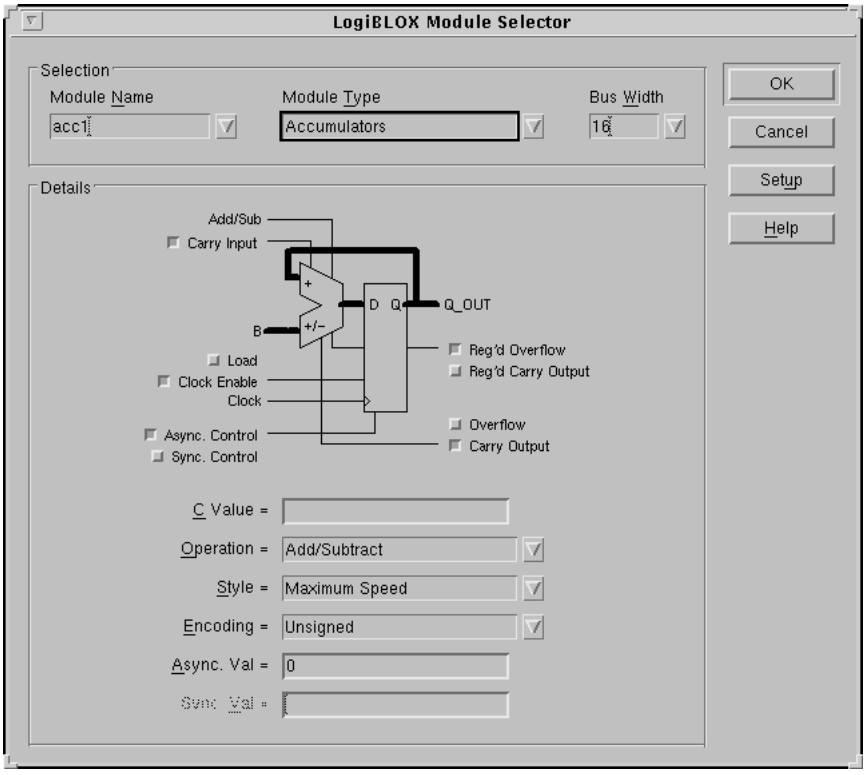


Figure 1. LogiBLOX Module Selector

The LogiBLOX graphical user interface (GUI) is available directly from supported schematic editor packages. LogiBLOX also supports synthesis-based tools by either letting the synthesis tool infer modules from the HDL code, or by letting you specify the module with the LogiBLOX Module Selector in its stand-alone mode and then instantiating it in your HDL design.

LogiBLOX modules are created to take full advantage of the architecture being targeted. Supported architectures are the XC3000A, XC3100A, XC4000E, XC4000EX and XC5200 device families.

Advantages of using LogiBLOX

LogiBLOX includes the following features that enhance design entry and module processing:

LogiBLOX Design Entry Features

- LogiBLOX eases design entry by allowing you to tailor complex logic blocks to precisely match your design's needs.
- The LogiBLOX graphical user interface allows you to quickly and easily specify complex modules with the assistance of interactive Design Rule Checker (DRC) checks and prompts.
- An image of the module with the specified pins and attributes is updated each time you activate a module attribute or connection in the Module Selector.
- The graphical user interface automatically disables selections that are incompatible with your current design selections.

LogiBLOX Processing Features

- In a synthesis-based environment, the modules you create with LogiBLOX are implemented as you incorporate them into the rest of your design.
- In LogiBLOX, a simulation model (VHDL, EDIF, or Verilog) is generated for each LogiBLOX module during design entry. This enables immediate simulation of LogiBLOX design without logic implementation. The behavioral functional model is produced as you enter the LogiBLOX module in your schematic.
- Many synthesis tools automatically infer LogiBLOX modules. You can also incorporate LogiBLOX modules in HDL designs through instantiation without compromising behavioral simulation support.
- Modules are synthesized quickly "on-the-fly" by the LogiBLOX module compiler.

LogiBLOX Modules

LogiBLOX modules are listed below by function:

Modules	Function
Arithmetic	
ACCUMULATOR	Adds data to or subtracts it from the current value stored in the acculator register.
ADDER/SUB-TRACTER	Adds or subtracts two data inputs and a Carry input.
COMPARATOR	Compares the magnitude or equality of two values.
COUNTER	Generates a sequence of count values.
Logic	
CONSTANT	Forces a constant value onto a bus.
DECODER	Routes input data to 1-of-n lines on the output port.
MULTIPLEXER	Routes input data on 1-of-n lines to the output port.
SIMPLE GATES	Implements AND, INVERT, NAND, NOR, OR, XNOR, and XOR logic functions.
TRISTATE	Creates a tri-stated data bus.
I/O	
INPUT/OUTPUT	Connects internal and external pin signals.
PAD	Simulates an input/output pad.
Sequential	
CLOCK DIVIDER	Generates a period that is a multiple of the clock — input period.
COUNTER	Generates a sequence of count values.
SHIFT REGISTER	Shifts the input data to the left or right.
Storage	
DATA REGISTER	Captures the input data on active Clock transitions.
MEMORY: ROM, RAM, SYNC_RAM, DP_RAM	Stores information and makes it readable.
SHIFT REGISTER	Shifts the input data to the left or right.

February 8, 1998

Reference Designs

February 8,1998

Overview

Xilinx offers a number of reference designs in the form of application notes with supporting design files. These designs represent good starting points for implementing simple to complex functions in Xilinx programmable logic.

These designs are supplied free from Xilinx, and come without technical support or warranty. For pre-implemented and verified functions that include technical support, we suggest that you consider the Xilinx LogiCORE or third party AllianceCORE products.

Acquiring Reference Designs

Except where noted, all reference design documentation and design files are available for download from the Xilinx website. The tables included with each reference design description include the names of the associated web files.

All XAPP files have an XAPP number in the file name, and can generally be found on the Xilinx web at:

URL: www.xilinx.com/apps/xapp.htm

All other files can generally be found at the web location:

URL: www.xilinx.com/apps/fpga.htm

If you are unable to find a particular file, you can always use the Xilinx SmartSearch™ and enter the name of the file you want.

Standard Bus Interfaces

Plug and Play ISA Interface in Xilinx

Device Family	XC4000
Documentation	plugplay.pdf
Design File Formats	ViewLogic (pnp_v090.exe pnp_upd.exe)

This Application Note describes a Plug and Play ISA interface reference design using a Xilinx XC4003-6PQ100C, or larger, FPGA device. This design implements the features used in a majority of Plug and Play designs but does not implement every option available within the Plug and Play specification.

DSP Functions

16-Tap, 8-Bit FIR Filter

Device Family	XC4000
Documentation	fir_filt.pdf
Design File Formats	ViewLogic (fir_filt.zip)

This application note describes the functionality and integration of a 16-Tap, 8-Bit Finite Impulse Response (FIR) filter macro with predefined coefficients (e.g. low pass) and a sample rate of 5.44 mega-samples per second or 784 MIPS using an XC4000-4 device. The application note also describes how to set the coefficients of the FIR Filter to meet the needs of other applications.

Adders, Subtracters and Accumulators in XC3000

Device Family	XC3000
Documentation	xapp022.pdf
Design File Formats	ViewdrawLCA (xapp022v.zip) OrCAD 386+ V1.10 (xapp022o.zip)

This application note surveys the different adder techniques that are available for XC3000 designs. Examples are shown, and a speed/size comparison is made.

Multiplexers and Barrel Shifters in XC3000/ XC3100

Device Family	XC3000, XC3100
Documentation	xapp026.pdf
Design File Formats	Viewdraw-LCA (xapp026v.zip) OrCAD 386+ V1.10 (xapp026o.zip)

This application note provides guidance for implementing high performance multiplexers and barrel shifters in XC3000 devices.

Base-Level Functions

Dynamic Microcontroller in XC4000

Device Family	XC4000
Documentation	microcnt.pdf
File Formats	ViewLogic (psmfiles.exe)

An application note and design files for a microcontroller with dynamic bus sizing. Uses Xilinx X-BLOX. A QBASIC-based assembler is available.

Loadable Binary Counters

Device Family	XC3000
Documentation	xapp004.pdf
Design File Formats	Viewdraw-LCA (xapp004v.zip) OrCAD 386+ V1.10 (xapp004o.zip)

The design strategies for loadable and non-loadable binary counters are significantly different. This application note discusses the differences, and describes the design of a loadable binary counter. Up, down and up/down counters are described, with lengths of 16 and 32 bits. Design files are available for all six versions.

Ultra-Fast Synchronous Counters

Device Family	XC3000, XC4000
Documentation	xapp014.pdf
Design File Formats	Viewdraw-LCA (xapp014v.zip) OrCAD 386+ V1.10 (xapp014o.zip)

This fully synchronous, non-loadable, binary counter uses a traditional prescaler technique to achieve high performance. Typically, the speed of a synchronous prescaler counter is limited by the delay incurred distributing the parallel Count Enable. This design minimizes that delay by replicating the LSB of the counter. In this way even the small Longline delay is eliminated, resulting in the fastest possible synchronous counter.

Accelerating Loadable Counters in XC4000

Device Family	XC4000
Documentation	xapp023.pdf
Design File Formats	Viewdraw-LCA (xapp023v.zip) OrCAD 386+ V1.10 (xapp023o.zip)

The XC4000 dedicated carry logic provides for very compact, high-performance counters. This Application Note describes a technique for increasing the performance of these counters using minimum additional logic. Using this technique, the counters remain loadable.

Implementing FIFOs in XC4000 Series RAM

Device Family	XC4000
Documentation	xapp053.pdf
Design Files	Xilinx Web Xilinx Literature (PCI Packet)
File Formats	ViewLogic (xapp053v.zip)

This application note demonstrates how to use the various RAM modes in XC4000-Series logic blocks. A simple FIFO is implemented in several different ways, using combinations of level-sensitive (asynchronous) and edge-triggered (synchronous), single-port and dual-port RAM.

Register-Based FIFO

Device Family	XC3000
Documentation	xapp005.pdf
Design File Formats	Viewdraw-LCA (xapp005v.zip) OrCAD 386+ V1.10 (xapp005o.zip)

While XC3000-series LCA devices do not provide RAM, it is possible to construct small register-based FIFOs. A basic synchronous FIFO requires one CLB for each two bits of FIFO capacity, plus one CLB for each word in the FIFO. Optional asynchronous input and output circuits are provided. Design files are available for two implementations of this design. The fastest of the two implementations uses a constraints file to achieve better placement.

Pulse-Width Modulation in Xilinx

Device Family	XC3000, XC4000, XC5000
Documentation	pwm.pdf
Design File Formats	ViewLogic (pwm.zip)

An application note and design files for building a pulse-width modulation circuit in Xilinx programmable logic. Uses Xilinx X-BLOX.

4Mbit Virtual SPROM

Device Family	XC9500
Documentation	xapp079.pdf
Design File Formats	vsprom.zip

This application note describes the design of a very low cost, CPLD-based virtual SPROM downloader for programming the Xilinx high-density XC4000-Series FPGAs in embedded applications.

Harmonic Frequency Synthesizer and FSK Modulator

Device Family	XC3000, XC4000
Documentation	xapp009.pdf
Design File Formats	Viewdraw-LCA (xapp009v.zip) OrCAD 386+ V1.10 (xapp009o.zip)

The Harmonic Frequency Synthesizer uses an accumulator technique to generate frequencies that are evenly spaced harmonics of some minimum frequency. Extensive pipelining is employed to permit high clock rates.

The FSK Modulator is a modification of the Harmonic Frequency Synthesizer that automatically switches between two frequencies in accordance with an NRZ input.

Frequency/Phase Comparator for Phase-Locked Loops

Device Family	XC3000, XC4000, XC5000
Documentation	xapp028.pdf
Design File Formats	Viewdraw-LCA (xapp028v.zip) OrCAD 386+ V1.10 (xapp028o.zip)

The phase comparator described in this application note permits phase-locked loops to be constructed using LCA devices that only require an external voltage-controlled oscillator and integrating amplifier.

Serial Code Conversion Between BCD and Binary

Device Family	XC3000
Documentation	xapp029.pdf
Design File Formats	Viewdraw-LCA (xapp029v.zip) OrCAD 386+ V1.10 (xapp029o.zip)

Binary-to-BCD and BCD-to-binary conversions are performed between serial binary values and parallel BCD values.

Configuring FPGAs Over a Processor Bus

Device Family	XC3000, XC4000, XC5000
Documentation	bus_conf.pdf
Design File Formats	ViewLogic and C source code: pc_isa.zip

This application note describes how to configure an SRAM-based FPGA over a processor bus. It also illustrates the source code required to download a configuration bitstream using an IBM PC as a host microprocessor.'C' source code is provided. Useful in reconfigurable computing applications.