



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



## Features

- Fast, Extended Block RAM, 1.8 V FPGA Family
  - 560 Kb and 1,120 Kb embedded block RAM
  - 130 MHz internal performance (four LUT levels)
  - PCI compliant 3.3 V, 32/64-bit, 33/66-MHz
- Sophisticated SelectRAM+™ Memory Hierarchy
  - 294 Kb of internal configurable distributed RAM
  - Up to 1,120 Kb of synchronous internal block RAM
  - True Dual-Port block RAM
  - Memory bandwidth up to 2.24 Tb/s (equivalent bandwidth of over 100 RAMBUS channels)
  - Designed for high-performance Interfaces to external memories
    - 200 MHz ZBT\* SRAMs
    - 200 Mb/s DDR SDRAMs
- Highly Flexible SelectIO+™ Technology
  - Supports 20 high-performance interface standards
  - Up to 556 singled-ended I/Os or up to 201 differential I/O pairs for an aggregate bandwidth of >100 Gb/s
- Complete Industry-Standard Differential Signalling Support
  - LVDS (622 Mb/s), BLVDS (Bus LVDS), LVPECL
  - All I/O signals can be input, output, or bi-directional
- LVPECL and LVDS clock inputs for 300+ MHz clocks
- Proprietary High-Performance SelectLink™ Technology
  - 80 Gb/s chip-to-chip communication link
  - Support for Double Data Rate (DDR) interface
  - Web-based HDL generation methodology
- Eight Fully Digital Delay-Locked Loops (DLLs)
- IEEE 1149.1 boundary-scan logic
- Supported by Xilinx Foundation Series™ and Alliance Series™ Development Systems
  - Internet Team Design (Xilinx iTD™) tool ideal for million-plus gate density designs
  - Wide selection of PC or workstation platforms
- SRAM-based In-System Configuration
  - Unlimited re-programmability
- Advanced Packaging Options
  - 1.0 mm FG676 and FG900
  - 1.27 mm BG560
- 0.18 μm 6-layer Metal Process with Copper Interconnect
- 100% Factory Tested

\* ZBT is a trademark of Integrated Device Technology, Inc.

## Introduction

The Virtex™-E Extended Memory (Virtex-EM) family of FPGAs is an extension of the highly successful Virtex-E family architecture. The Virtex-EM family (devices shown in [Table 1](#)) includes all of the features of Virtex-E, plus additional block RAM, useful for applications such as network switches and high-performance video graphic systems.

Xilinx developed the Virtex-EM product family to enable customers to design systems requiring high memory bandwidth, such as 160 Gb/s network switches. Unlike traditional ASIC devices, this family also supports fast time-to-market delivery, because the development engineering is already completed. Just complete the design and program the device. There is no NRE, no silicon production cycles, and no additional delays for design re-work. In addition, designers can update the design over a network at any time, providing product upgrades or updates to customers even sooner.

The Virtex-EM family is the result of more than fifteen years of FPGA design experience. Xilinx has a history of support-

ing customer applications by providing the highest level of logic, RAM, and features available in the industry. The Virtex-EM family, first FPGAs to deploy copper interconnect, offers the performance and high memory bandwidth for advanced system integration without the initial investment, long development cycles, and inventory risk expected in traditional ASIC development.

Table 1: Virtex-E Extended Memory Field-Programmable Gate Array Family Members

Device	Logic Gates	CLB Array	Logic Cells	Differential I/O Pairs	User I/O	BlockRAM Bits	Distributed RAM Bits
XCV405E	129,600	40 x 60	10,800	183	404	573,440	153,600
XCV812E	254,016	56 x 84	21,168	201	556	1,146,880	301,056

## Virtex-E Compared to Virtex Devices

The Virtex-E family offers up to 43,200 logic cells in devices up to 30% faster than the Virtex family.

I/O performance is increased to 622 Mb/s using Source Synchronous data transmission architectures and synchronous system performance up to 240 MHz using singled-ended SelectI/O technology. Additional I/O standards are supported, notably LVPECL, LVDS, and BLVDS, which use two pins per signal. Almost all signal pins can be used for these new standards.

Virtex-E devices have up to 640 Kb of faster (250MHz) block SelectRAM, but the individual RAMs are the same size and structure as in the Virtex family. They also have eight DLLs instead of the four in Virtex devices. Each individual DLL is slightly improved with easier clock mirroring and 4x frequency multiplication.

$V_{CCINT}$ , the supply voltage for the internal logic and memory, is 1.8 V, instead of 2.5 V for Virtex devices. Advanced processing and 0.18  $\mu\text{m}$  design rules have resulted in smaller dice, faster speed, and lower power consumption.

I/O pins are 3 V tolerant, and can be 5 V tolerant with an external 100  $\Omega$  resistor. PCI 5 V is not supported. With the addition of appropriate external resistors, any pin can tolerate any voltage desired.

Banking rules are different. With Virtex devices, all input buffers are powered by  $V_{CCINT}$ . With Virtex-E devices, the LVTTL, LVCMOS2, and PCI input buffers are powered by the I/O supply voltage  $V_{CCO}$ .

The Virtex-E family is not bitstream-compatible with the Virtex family, but Virtex designs can be compiled into equivalent Virtex-E devices.

The same device in the same package for the Virtex-E and Virtex families are pin-compatible with some minor exceptions. See the data sheet pinout section for details.

## General Description

The Virtex-E FPGA family delivers high-performance, high-capacity programmable logic solutions. Dramatic increases in silicon efficiency result from optimizing the new architecture for place-and-route efficiency and exploiting an aggressive 6-layer metal 0.18  $\mu\text{m}$  CMOS process. These advances make Virtex-E FPGAs powerful and flexible alter-

natives to mask-programmed gate arrays. The Virtex-E family includes the nine members in Table 1.

Building on experience gained from Virtex FPGAs, the Virtex-E family is an evolutionary step forward in programmable logic design. Combining a wide variety of programmable system features, a rich hierarchy of fast, flexible interconnect resources, and advanced process technology, the Virtex-E family delivers a high-speed and high-capacity programmable logic solution that enhances design flexibility while reducing time-to-market.

## Virtex-E Architecture

Virtex-E devices feature a flexible, regular architecture that comprises an array of configurable logic blocks (CLBs) surrounded by programmable input/output blocks (IOBs), all interconnected by a rich hierarchy of fast, versatile routing resources. The abundance of routing resources permits the Virtex-E family to accommodate even the largest and most complex designs.

Virtex-E FPGAs are SRAM-based, and are customized by loading configuration data into internal memory cells. Configuration data can be read from an external SPROM (master serial mode), or can be written into the FPGA (SelectMAP™, slave serial, and JTAG modes).

The standard Xilinx Foundation Series™ and Alliance Series™ Development systems deliver complete design support for Virtex-E, covering every aspect from behavioral and schematic entry, through simulation, automatic design translation and implementation, to the creation and downloading of a configuration bit stream.

## Higher Performance

Virtex-E devices provide better performance than previous generations of FPGAs. Designs can achieve synchronous system clock rates up to 240 MHz including I/O or 622 Mb/s using Source Synchronous data transmission architectures. Virtex-E I/Os comply fully with 3.3 V PCI specifications, and interfaces can be implemented that operate at 33 MHz or 66 MHz.

While performance is design-dependent, many designs operate internally at speeds in excess of 133 MHz and can achieve over 311 MHz. Table 2, page 3, shows performance data for representative circuits, using worst-case timing parameters.

Table 2: Performance for Common Circuit Functions

Function	Bits	Virtex-E -7
<b>Register-to-Register</b>		
Adder	16	4.3 ns
	64	6.3 ns
Pipelined Multiplier	8 x 8	4.4 ns
	16 x 16	5.1 ns
Address Decoder	16	3.8 ns
	64	5.5 ns
16:1 Multiplexer		4.6 ns
Parity Tree	9	3.5 ns
	18	4.3 ns
	36	5.9 ns
<b>Chip-to-Chip</b>		
HSTL Class IV		
LVTTL, 16mA, fast slew		
LVDS		
LVPECL		

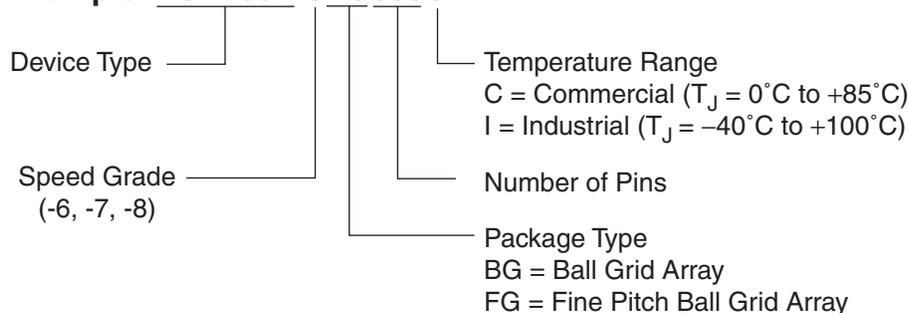
## Virtex-E Extended Memory Device/Package Combinations and Maximum I/O

Table 3: Virtex-EM Family Maximum User I/O by Device/Package (Excluding Dedicated Clock Pins)

Package	XCV405E	XCV812E
BG560	404	404
FG676	404	
FG900		556

## Virtex-E Extended Memory Ordering Information

Example: XCV405E-6BG560C



DS025\_001\_112000

Figure 1: Virtex Ordering Information

## Revision History

The following table shows the revision history for this document.

Date	Version	Revision
03/23/00	1.0	Initial Xilinx release.
08/01/00	1.1	Accumulated edits and fixes. Upgrade to Preliminary. Preview -8 numbers added. Reformatted to adhere to corporate documentation style guidelines. Minor changes in BG560 pin-out table.
09/19/00	1.2	<ul style="list-style-type: none"> <li>In Table 3 (Module 4), <b>FG676 Fine-Pitch BGA — XCV405E</b>, the following pins are no longer labeled as VREF: B7, G16, G26, W26, AF20, AF8, Y1, H1.</li> <li>Min values added to <b>Virtex-E Electrical Characteristics</b> tables.</li> </ul>
11/20/00	1.3	<ul style="list-style-type: none"> <li>Updated speed grade -8 numbers in <b>Virtex-E Electrical Characteristics</b> tables (Module 3).</li> <li>Updated minimums in Table 11 (Module 2), and added notes to Table 12 (Module 2).</li> <li>Added to note 2 of <b>Absolute Maximum Ratings</b> (Module 3).</li> <li>Changed all minimum hold times to -0.4 for <b>Global Clock Set-Up and Hold for LVTTL Standard, with DLL</b> (Module 3).</li> <li>Revised maximum <math>T_{DLLPW}</math> in -6 speed grade for <b>DLL Timing Parameters</b> (Module 3).</li> </ul>
04/02/01	1.4	<ul style="list-style-type: none"> <li>In <b>Table 4, FG676 Fine-Pitch BGA — XCV405E</b>, pin B19 is no longer labeled as VREF, and pin G16 is now labeled as VREF.</li> <li>Updated values in <b>Virtex-E Switching Characteristics</b> tables.</li> <li>Converted data sheet to modularized format. See <b>Virtex-E Extended Memory Data Sheet</b>, below.</li> </ul>
07/17/02	1.5	<ul style="list-style-type: none"> <li>Data sheet designation upgraded from Preliminary to Production.</li> </ul>

## Virtex-E Extended Memory Data Sheet

The Virtex-E Extended Memory Data Sheet contains the following modules:

- DS025-1, Virtex-E 1.8V Extended Memory FPGAs: **Introduction and Ordering Information (Module 1)**
- DS025-2, Virtex-E 1.8V Extended Memory FPGAs: **Functional Description (Module 2)**
- DS025-3, Virtex-E 1.8V Extended Memory FPGAs: **DC and Switching Characteristics (Module 3)**
- DS025-4, Virtex-E 1.8V Extended Memory FPGAs: **Pinout Tables (Module 4)**

## Architectural Description

### Virtex-E Array

The Virtex-E user-programmable gate array (see **Figure 1**) comprises two major configurable elements: configurable logic blocks (CLBs) and input/output blocks (IOBs).

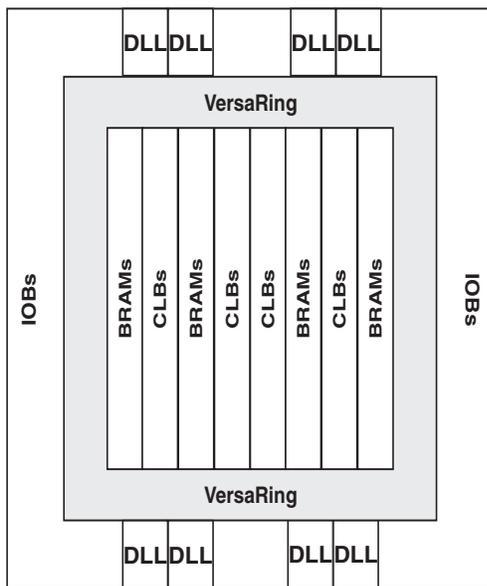
- CLBs provide the functional elements for constructing logic.
- IOBs provide the interface between the package pins and the CLBs.

CLBs interconnect through a general routing matrix (GRM). The GRM comprises an array of routing switches located at the intersections of horizontal and vertical routing channels. Each CLB nests into a VersaBlock™ that also provides local routing resources to connect the CLB to the GRM.

The VersaRing™ I/O interface provides additional routing resources around the periphery of the device. This routing improves I/O routability and facilitates pin locking.

The Virtex-E architecture also includes the following circuits that connect to the GRM:

- Dedicated block memories of 4096 bits each
- Clock DLLs for clock-distribution delay compensation and clock domain control
- 3-State buffers (BUFTs) associated with each CLB that drive dedicated segmentable horizontal routing resources



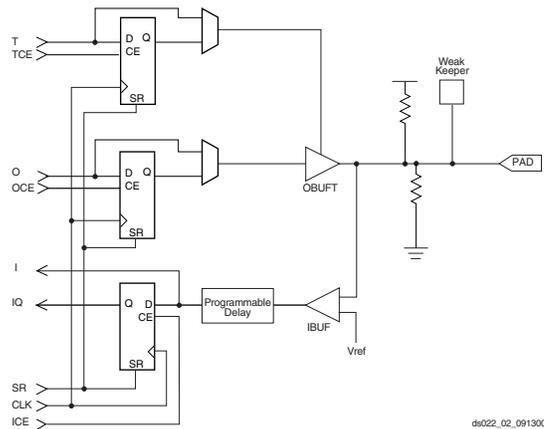
ds022\_001\_121099

Figure 1: Virtex-E Architecture Overview

Values stored in static memory cells control the configurable logic elements and interconnect resources. These values load into the memory cells on power-up, and can reload if necessary to change the function of the device.

### Input/Output Block

The Virtex-E IOB, **Figure 2**, features SelectIO+™ inputs and outputs that support a wide variety of I/O signalling standards (see **Table 1**).



ds022\_02\_091300

Figure 2: Virtex-E Input/Output Block (IOB)

The three IOB storage elements function either as edge-triggered D-type flip-flops or as level-sensitive latches. Each IOB has a clock signal (CLK) shared by the three flip-flops and independent delay clock enable signals for each flip-flop.

Table 1: Supported I/O Standards

I/O Standard	Output $V_{CCO}$	Input $V_{CCO}$	Input $V_{REF}$	Board Termination Voltage ( $V_{TT}$ )
LVTTL	3.3	3.3	N/A	N/A
LVC MOS2	2.5	2.5	N/A	N/A
LVC MOS18	1.8	1.8	N/A	N/A
SSTL3 I & II	3.3	N/A	1.50	1.50
SSTL2 I & II	2.5	N/A	1.25	1.25
GTL	N/A	N/A	0.80	1.20
GTL+	N/A	N/A	1.0	1.50
HSTL I	1.5	N/A	0.75	0.75
HSTL III & IV	1.5	N/A	0.90	1.50
CTT	3.3	N/A	1.50	1.50
AGP-2X	3.3	N/A	1.32	N/A
PCI33_3	3.3	3.3	N/A	N/A
PCI66_3	3.3	3.3	N/A	N/A
BLVDS & LVDS	2.5	N/A	N/A	N/A
LVPECL	3.3	N/A	N/A	N/A

In addition to the CLK and CE control signals, the three flip-flops share a Set/Reset (SR). For each flip-flop, this signal can be independently configured as a synchronous Set, a synchronous Reset, an asynchronous Preset, or an asynchronous Clear.

The output buffer and all of the IOB control signals have independent polarity controls.

All pads are protected against damage from electrostatic discharge (ESD) and from over-voltage transients. After configuration, clamping diodes are connected to  $V_{CCO}$  with the exception of LVC MOS18, LVC MOS25, GTL, GTL+, LVDS, and LVPECL.

Optional pull-up, pull-down and weak-keeper circuits are attached to each pad. Prior to configuration all outputs not involved in configuration are forced into their high-impedance state. The pull-down resistors and the weak-keeper circuits are inactive, but IOs can optionally be pulled up.

The activation of pull-up resistors prior to configuration is controlled on a global basis by the configuration mode pins. If the pull-up resistors are not activated, all the pins are in a high-impedance state. Consequently, external pull-up or pull-down resistors must be provided on pins required to be at a well-defined logic level prior to configuration.

All Virtex-E IOBs support IEEE 1149.1-compatible boundary scan testing.

### Input Path

The Virtex-E IOB input path routes the input signal directly to internal logic and/ or through an optional input flip-flop.

An optional delay element at the D-input of this flip-flop eliminates pad-to-pad hold time. The delay is matched to the internal clock-distribution pad delay of the FPGA, and when used, assures that the pad-to-pad hold time is zero.

Each input buffer can be configured to conform to any of the low-voltage signalling standards supported. In some of these standards the input buffer utilizes a user-supplied threshold voltage,  $V_{REF}$ . The need to supply  $V_{REF}$  imposes constraints on which standards can be used in close proximity to each other. See "I/O Banking" on page 2.

There are optional pull-up and pull-down resistors at each user I/O input for use after configuration. Their value is in the range 50 - 100 k $\Omega$ .

### Output Path

The output path includes a 3-state output buffer that drives the output signal onto the pad. The output signal can be routed to the buffer directly from the internal logic or through an optional IOB output flip-flop.

The 3-state control of the output can also be routed directly from the internal logic or through a flip-flop that provides synchronous enable and disable.

Each output driver can be individually programmed for a wide range of low-voltage signalling standards. Each output buffer can source up to 24 mA and sink up to 48 mA. Drive strength and slew rate controls minimize bus transients.

In most signalling standards, the output High voltage depends on an externally supplied  $V_{CCO}$  voltage. The need to supply  $V_{CCO}$  imposes constraints on which standards can be used in close proximity to each other. See "I/O Banking" on page 2.

An optional weak-keeper circuit is connected to each output. When selected, the circuit monitors the voltage on the pad and weakly drives the pin High or Low to match the input signal. If the pin is connected to a multiple-source signal, the weak keeper holds the signal in its last state if all drivers are disabled. Maintaining a valid logic level in this way eliminates bus chatter.

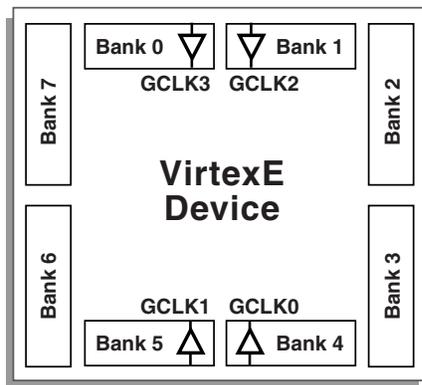
Since the weak-keeper circuit uses the IOB input buffer to monitor the input level, an appropriate  $V_{REF}$  voltage must be provided if the signalling standard requires one. The provision of this voltage must comply with the I/O banking rules.

### I/O Banking

Some of the I/O standards described above require  $V_{CCO}$  and/or  $V_{REF}$  voltages. These voltages are externally supplied and connected to device pins that serve groups of

I/Os, called banks. Consequently, restrictions exist about which I/O standards can be combined within a given bank.

Eight I/O banks result from separating each edge of the FPGA into two banks, as shown in Figure 3. Each bank has multiple  $V_{CCO}$  pins, all of which must be connected to the same voltage. This voltage is determined by the output standards in use.



ds022\_03\_121799

Figure 3: Virtex-E I/O Banks

Within a bank, output standards can be mixed only if they use the same  $V_{CCO}$ . Compatible standards are shown in Table 2. GTL and GTL+ appear under all voltages because their open-drain outputs do not depend on  $V_{CCO}$ .

Table 2: Compatible Output Standards

$V_{CCO}$	Compatible Standards
3.3 V	PCI, LVTTTL, SSTL3 I, SSTL3 II, CTT, AGP, GTL, GTL+, LVPECL
2.5 V	SSTL2 I, SSTL2 II, LVCMOS2, GTL, GTL+, BLVDS, LVDS
1.8 V	LVCMOS18, GTL, GTL+
1.5 V	HSTL I, HSTL III, HSTL IV, GTL, GTL+

Some input standards require a user-supplied threshold voltage,  $V_{REF}$ . In this case, certain user-I/O pins are automatically configured as inputs for the  $V_{REF}$  voltage. Approximately one in six of the I/O pins in the bank assume this role.

The  $V_{REF}$  pins within a bank are interconnected internally and consequently only one  $V_{REF}$  voltage can be used within each bank. All  $V_{REF}$  pins in the bank, however, must be connected to the external voltage source for correct operation.

Within a bank, inputs that require  $V_{REF}$  can be mixed with those that do not. However, only one  $V_{REF}$  voltage can be used within a bank.

In Virtex-E, input buffers with LVTTTL, LVCMOS2, LVCMOS18, PCI33\_3, PCI66\_3 standards are supplied by  $V_{CCO}$  rather than  $V_{CCINT}$ . For these standards, only input and output buffers that have the same  $V_{CCO}$  can be mixed together.

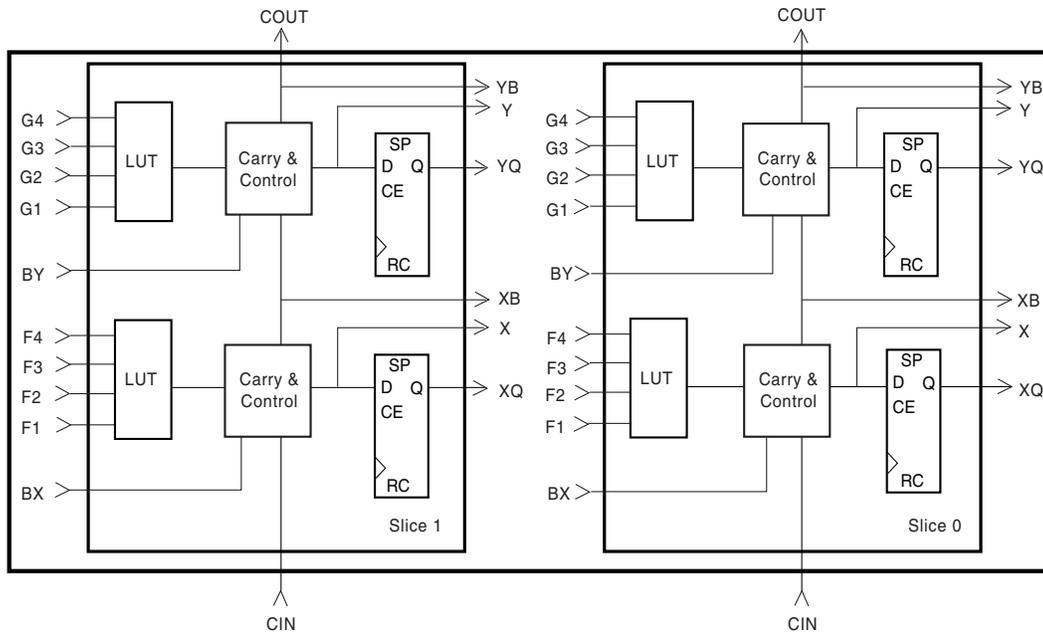
The  $V_{CCO}$  and  $V_{REF}$  pins for each bank appear in the device pin-out tables and diagrams. The diagrams also show the bank affiliation of each I/O.

Within a given package, the number of  $V_{REF}$  and  $V_{CCO}$  pins can vary depending on the size of device. In larger devices, more I/O pins convert to  $V_{REF}$  pins. Since these are always a super set of the  $V_{REF}$  pins used for smaller devices, it is possible to design a PCB that permits migration to a larger device if necessary. All the  $V_{REF}$  pins for the largest device anticipated must be connected to the  $V_{REF}$  voltage, and not used for I/O.

In smaller devices, some  $V_{CCO}$  pins used in larger devices do not connect within the package. These unconnected pins can be left unconnected externally, or they can be connected to the  $V_{CCO}$  voltage to permit migration to a larger device, if necessary.

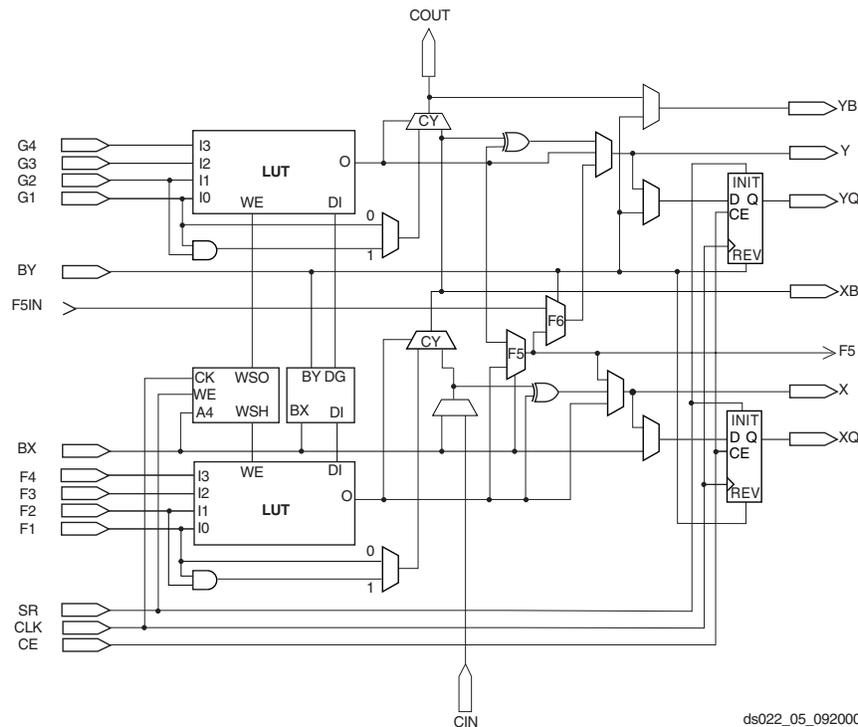
### Configurable Logic Block

The basic building block of the Virtex-E CLB is the logic cell (LC). An LC includes a 4-input function generator, carry logic, and a storage element. The output from the function generator in each LC drives both the CLB output and the D input of the flip-flop. Each Virtex-E CLB contains four LCs, organized in two similar slices, as shown in Figure 4. Figure 5 shows a more detailed view of a single slice.



ds022\_04\_121799

Figure 4: 2-Slice Virtex-E CLB



ds022\_05\_092000

Figure 5: Detailed View of Virtex-E Slice

In addition to the four basic LCs, the Virtex-E CLB contains logic that combines function generators to provide functions of five or six inputs. Consequently, when estimating the number of system gates provided by a given device, each CLB counts as 4.5 LCs.

### Look-Up Tables

Virtex-E function generators are implemented as 4-input look-up tables (LUTs). In addition to operating as a function generator, each LUT can provide a 16 x 1-bit synchronous RAM. Furthermore, the two LUTs within a slice can be com-

bined to create a 16 x 2-bit or 32 x 1-bit synchronous RAM, or a 16 x 1-bit dual-port synchronous RAM.

The Virtex-E LUT can also provide a 16-bit shift register that is ideal for capturing high-speed or burst-mode data. This mode can also be used to store data in applications such as Digital Signal Processing.

**Storage Elements**

The storage elements in the Virtex-E slice can be configured either as edge-triggered D-type flip-flops or as level-sensitive latches. The D inputs can be driven either by the function generators within the slice or directly from slice inputs, bypassing the function generators.

In addition to Clock and Clock Enable signals, each Slice has synchronous set and reset signals (SR and BY). SR forces a storage element into the initialization state specified for it in the configuration. BY forces it into the opposite state. Alternatively, these signals can be configured to operate asynchronously. All of the control signals are independently invertible, and are shared by the two flip-flops within the slice.

**Additional Logic**

The F5 multiplexer in each slice combines the function generator outputs. This combination provides either a function generator that can implement any 5-input function, a 4:1 multiplexer, or selected functions of up to nine inputs.

Similarly, the F6 multiplexer combines the outputs of all four function generators in the CLB by selecting one of the F5-multiplexer outputs. This permits the implementation of any 6-input function, an 8:1 multiplexer, or selected functions of up to 19 inputs.

Each CLB has four direct feedthrough paths, two per slice. These paths provide extra data input lines or additional local routing that does not consume logic resources.

Table 3: CLB/Block RAM Column Locations

Virtex-E Device	0	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68	72	76	80	84	
XCV405E	√	√	√	√	√	√	√			√	√	√	√	√	√	√							
XCV812E	√	√	√	√	√	√	√	√	√	√			√	√	√	√	√	√	√	√	√	√	√

Table 4 shows the amount of block SelectRAM memory that is available in each Virtex-E device.

Table 4: Virtex-E Block SelectRAM Amounts

Virtex-E Device	# of Blocks	Block SelectRAM Bits
XCV405E	140	573,440
XCV812E	280	1,146,880

**Arithmetic Logic**

Dedicated carry logic provides fast arithmetic carry capability for high-speed arithmetic functions. The Virtex-E CLB supports two separate carry chains, one per Slice. The height of the carry chains is two bits per CLB.

The arithmetic logic includes an XOR gate that allows a 2-bit full adder to be implemented within a slice. In addition, a dedicated AND gate improves the efficiency of multiplier implementation.

The dedicated carry path can also be used to cascade function generators for implementing wide logic functions.

**BUFTs**

Each Virtex-E CLB contains two 3-state drivers (BUFTs) that can drive on-chip busses. See "Dedicated Routing" on page 7. Each Virtex-E BUFT has an independent 3-state control pin and an independent input pin.

**Block SelectRAM+ Memory**

Virtex-E FPGAs incorporate large block SelectRAM memories. These complement the Distributed SelectRAM memories that provide shallow RAM structures implemented in CLBs.

Block SelectRAM memory blocks are organized in columns, starting at the left (column 0) and right outside edges and inserted every four CLB columns (see notes for smaller devices). Each memory block is four CLBs high, and each memory column extends the full height of the chip, immediately adjacent (to the right, except for column 0) of the CLB column locations indicated in Table 3.

Each block SelectRAM cell, as illustrated in Figure 6, is a fully synchronous dual-ported (True Dual Port) 4096-bit RAM with independent control signals for each port. The data widths of the two ports can be configured independently, providing built-in bus-width conversion.

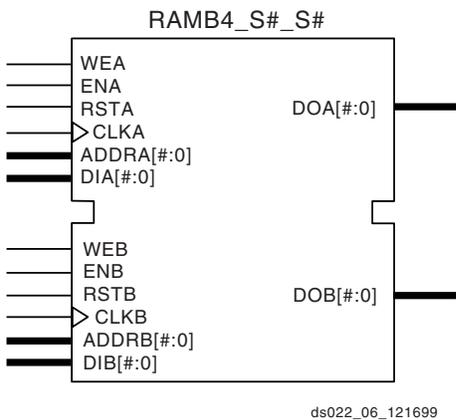


Figure 6: Dual-Port Block SelectRAM

Table 5 shows the depth and width aspect ratios for the block SelectRAM. The Virtex-E block SelectRAM also includes dedicated routing to provide an efficient interface with both CLBs and other block SelectRAM modules. Refer to XAPP130 for block SelectRAM timing waveforms.

Table 5: Block SelectRAM Port Aspect Ratios

Width	Depth	ADDR Bus	Data Bus
1	4096	ADDR<11:0>	DATA<0>
2	2048	ADDR<10:0>	DATA<1:0>
4	1024	ADDR<9:0>	DATA<3:0>
8	512	ADDR<8:0>	DATA<7:0>
16	256	ADDR<7:0>	DATA<15:0>

## Programmable Routing Matrix

It is the longest delay path that limits the speed of any worst-case design. Consequently, the Virtex-E routing architecture and its place-and-route software were defined in a joint optimization process. This joint optimization minimizes long-path delays, and consequently, yields the best system performance.

The joint optimization also reduces design compilation times because the architecture is software-friendly. Design cycles are correspondingly reduced due to shorter design iteration times.

### Local Routing

The VersaBlock, shown in Figure 7, provides local routing resources with the following types of connections:

- Interconnections among the LUTs, flip-flops, and GRM
- Internal CLB feedback paths that provide high-speed connections to LUTs within the same CLB, chaining them together with minimal routing delay
- Direct paths that provide high-speed connections between horizontally adjacent CLBs, eliminating the

delay of the GRM

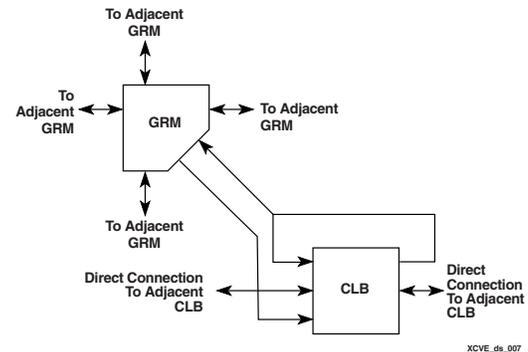


Figure 7: Virtex-E Local Routing

### General Purpose Routing

Most Virtex-E signals are routed on the general purpose routing, and consequently, the majority of interconnect resources are associated with this level of the routing hierarchy. The general routing resources are located in horizontal and vertical routing channels associated with the CLB rows and columns. The general-purpose routing resources are listed below.

- Adjacent to each CLB is a General Routing Matrix (GRM). The GRM is the switch matrix through which horizontal and vertical routing resources connect, and is also the means by which the CLB gains access to the general purpose routing.
- 24 single-length lines route GRM signals to adjacent GRMs in each of the four directions.
- 72 buffered Hex lines route GRM signals to another GRMs six-blocks away in each one of the four directions. Organized in a staggered pattern, Hex lines are driven only at their endpoints. Hex-line signals can be accessed either at the endpoints or at the midpoint (three blocks from the source). One third of the Hex lines are bidirectional, while the remaining ones are uni-directional.
- 12 Longlines are buffered, bidirectional wires that distribute signals across the device quickly and efficiently. Vertical Longlines span the full height of the device, and horizontal ones span the full width of the device.

### I/O Routing

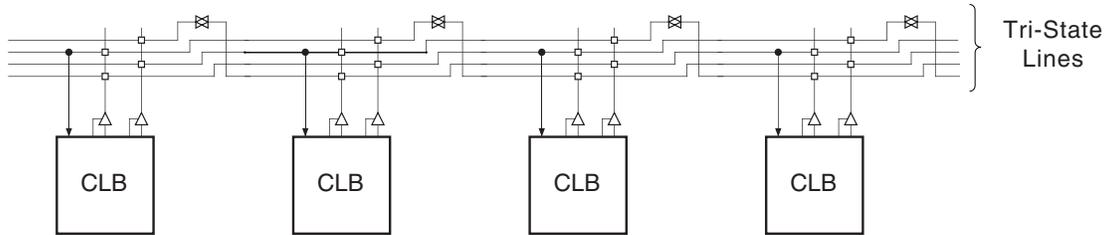
Virtex-E devices have additional routing resources around their periphery that form an interface between the CLB array and the IOBs. This additional routing, called the VersaRing, facilitates pin-swapping and pin-locking, such that logic redesigns can adapt to existing PCB layouts. Time-to-market is reduced, since PCBs and other system components can be manufactured while the logic design is still in progress.

**Dedicated Routing**

Some signal classes require dedicated routing resources to maximize performance. In the Virtex-E architecture, dedicated routing resources are provided for two signal classes.

- Horizontal routing resources are provided for on-chip 3-state busses. Four partitionable bus lines are provided per CLB row, permitting multiple busses within a row, as shown in **Figure 8**.

- Two dedicated nets per CLB propagate carry signals vertically to the adjacent CLB. Global Clock Distribution Network.
- DLL Location



buft\_c.eps

**Figure 8: BUFT Connections to Dedicated Horizontal Bus Lines**

**Clock Routing**

Clock Routing resources distribute clocks and other signals with very high fanout throughout the device. Virtex-E devices include two tiers of clock routing resources referred to as global and local clock routing resources.

- The global routing resources are four dedicated global nets with dedicated input pins that are designed to distribute high-fanout clock signals with minimal skew. Each global clock net can drive all CLB, IOB, and block RAM clock pins. The global nets can be driven only by global buffers. There are four global buffers, one for each global net.

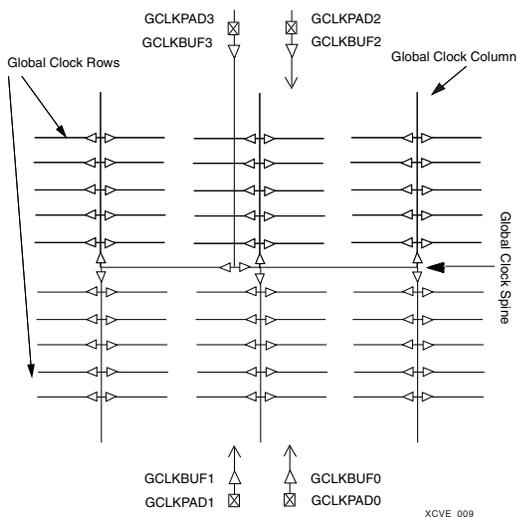
- The local clock routing resources consist of 24 backbone lines, 12 across the top of the chip and 12 across bottom. From these lines, up to 12 unique signals per column can be distributed via the 12 longlines in the column. These local resources are more flexible than the global resources since they are not restricted to routing only to clock pins.

**Global Clock Distribution**

Virtex-E provides high-speed, low-skew clock distribution through the global routing resources described above. A typical clock distribution net is shown in **Figure 9**.

Four global buffers are provided, two at the top center of the device and two at the bottom center. These drive the four global nets that in turn drive any clock pin.

Four dedicated clock pads are provided, one adjacent to each of the global buffers. The input to the global buffer is selected either from these pads or from signals in the general purpose routing.



XCVE\_009

**Figure 9: Global Clock Distribution Network**

**Digital Delay-Locked Loops**

There are eight DLLs (Delay-Locked Loops) per device, with four located at the top and four at the bottom, **Figure 10**. The DLLs can be used to eliminate skew between the clock input pad and the internal clock input pins throughout the device. Each DLL can drive two global clock networks. The DLL monitors the input clock and the distributed clock, and automatically adjusts a clock delay element. Additional delay is introduced such that clock edges arrive at internal flip-flops synchronized with clock edges arriving at the input.

In addition to eliminating clock-distribution delay, the DLL provides advanced control of multiple clock domains. The

DLL provides four quadrature phases of the source clock, and can double the clock or divide the clock by 1.5, 2, 2.5, 3, 4, 5, 8, or 16.

The DLL also operates as a clock mirror. By driving the output from a DLL off-chip and then back on again, the DLL can be used to de-skew a board level clock among multiple devices.

In order to guarantee that the system clock is operating correctly prior to the FPGA starting up after configuration, the DLL can delay the completion of the configuration process until after it has achieved lock.

For more information about DLL functionality, see the Design Consideration section of the data sheet.

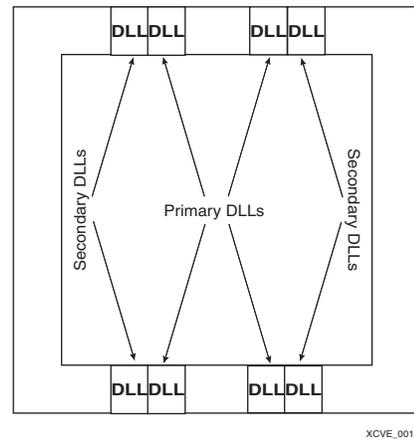


Figure 10: DLL Locations

## Boundary Scan

Virtex-E devices support all the mandatory boundary-scan instructions specified in the IEEE standard 1149.1. A Test Access Port (TAP) and registers are provided that implement the EXTEST, INTEST, SAMPLE/PRELOAD, BYPASS, IDCODE, USERCODE, and HIGHZ instructions. The TAP also supports two internal scan chains and configuration/readback of the device.

The JTAG input pins (TDI, TMS, TCK) do not have a  $V_{CCO}$  requirement, and operate with either 2.5 V or 3.3 V input signalling levels. The output pin (TDO) is sourced from the  $V_{CCO}$  in bank 2, and for proper operation of LVTTTL 3.3 V levels, the bank should be supplied with 3.3 V.

Boundary-scan operation is independent of individual IOB configurations, and unaffected by package type. All IOBs, including un-bonded ones, are treated as independent 3-state bidirectional pins in a single scan chain. Retention of the bidirectional test capability after configuration facilitates

the testing of external interconnections, provided the user design or application is turned off.

Table 6 lists the boundary-scan instructions supported in Virtex-E FPGAs. Internal signals can be captured during EXTEST by connecting them to un-bonded or unused IOBs. They can also be connected to the unused outputs of IOBs defined as unidirectional input pins.

Before the device is configured, all instructions except USER1 and USER2 are available. After configuration, all instructions are available. During configuration, it is recommended that those operations using the boundary-scan register (SAMPLE/PRELOAD, INTEST, EXTEST) not be performed.

In addition to the test instructions outlined above, the boundary-scan circuitry can be used to configure the FPGA, and also to read back the configuration data.

Figure 11 is a diagram of the Virtex-E Series boundary scan logic. It includes three bits of Data Register per IOB, the IEEE 1149.1 Test Access Port controller, and the Instruction Register with decodes.

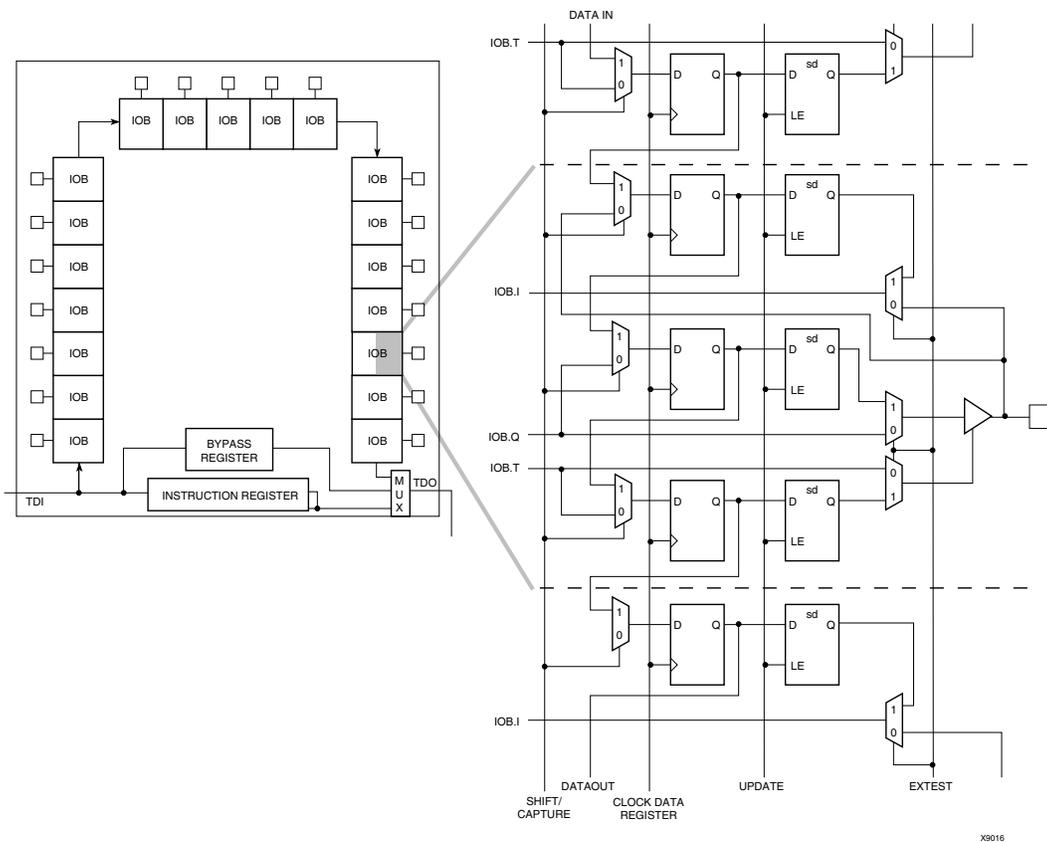


Figure 11: Virtex-E Family Boundary Scan Logic

Table 6: Boundary Scan Instructions

Boundary-Scan Command	Binary Code (4:0)	Description
EXTEST	00000	Enable boundary-scan EXTEST operation.
SAMPLE/PRELOAD	00001	Enable boundary-scan SAMPLE/PRELOAD operation.
USER1	00010	Access user-defined register 1.
USER2	00011	Access user-defined register 2.
CFG_OUT	00100	Access the configuration bus for read operations.
CFG_IN	00101	Access the configuration bus for write operations.
INTEST	00111	Enable boundary-scan INTEST operation.
USERCODE	01000	Enable shifting out USER code.
IDCODE	01001	Enable shifting out of ID Code.
HIGHZ	01010	3-state output pins while enabling the Bypass Register.
JSTART	01100	Clock the start-up sequence when StartupClk is TCK.
BYPASS	11111	Enable BYPASS.
RESERVED	All other codes	Xilinx reserved instructions.

### Instruction Set

The Virtex-E Series boundary scan instruction set also includes instructions to configure the device and read back configuration data (CFG\_IN, CFG\_OUT, and JSTART). The complete instruction set is coded as shown in Table 6.

### Data Registers

The primary data register is the boundary scan register. For each IOB pin in the FPGA, bonded or not, it includes three bits for In, Out, and 3-State Control. Non-IOB pins have appropriate partial bit population if input-only or output-only.

Each EXTEST CAPTURED-OR state captures all In, Out, and 3-state pins.

The other standard data register is the single flip-flop BYPASS register. It synchronizes data being passed through the FPGA to the next downstream boundary scan device.

The FPGA supports up to two additional internal scan chains that can be specified using the BSCAN macro. The macro provides two user pins (SEL1 and SEL2) which are decoded of the USER1 and USER2 instructions respectively. For these instructions, two corresponding pins (TDO1 and TDO2) allow user scan data to be shifted out of TDO.

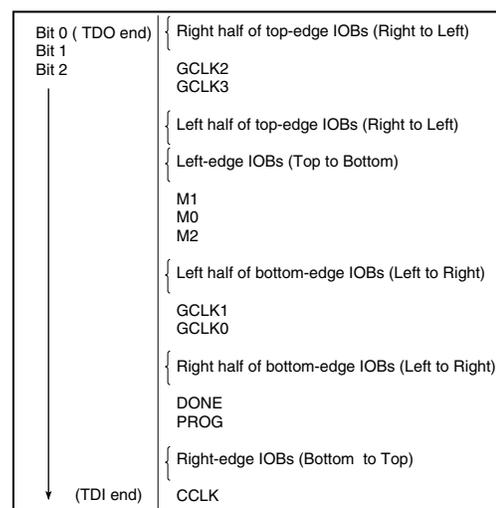
Likewise, there are individual clock pins (DRCK1 and DRCK2) for each user register. There is a common input pin (TDI) and shared output pins that represent the state of the TAP controller (RESET, SHIFT, and UPDATE).

### Bit Sequence

The order within each IOB is: In, Out, 3-State. The input-only pins contribute only the In bit to the boundary scan I/O data register, while the output-only pins contributes all three bits.

From a cavity-up view of the chip (as shown in EPIC), starting in the upper right chip corner, the boundary scan data-register bits are ordered as shown in Figure 12.

BSDL (Boundary Scan Description Language) files for Virtex-E Series devices are available on the Xilinx web site in the File Download area.



990602001

Figure 12: Boundary Scan Bit Sequence

### Identification Registers

The IDCODE register is supported. By using the IDCODE, the device connected to the JTAG port can be determined.

The IDCODE register has the following binary format:

```
vvvv:ffff:ffa:aaaa:aaaa:cccc:cccc:ccc1
```

where

v = the die version number

f = the family code (05 for Virtex-E family)

a = the number of CLB rows (ranges from 16 for XCV50E to 104 for XCV3200E)

c = the company code (49h for Xilinx)

The USERCODE register is supported. By using the USERCODE, a user-programmable identification code can be loaded and shifted out for examination. The identification code (see [Table 7](#)) is embedded in the bitstream during bitstream generation and is valid only after configuration.

## Development System

Virtex-E FPGAs are supported by the Xilinx Foundation and Alliance Series CAE tools. The basic methodology for Virtex-E design consists of three interrelated steps: design entry, implementation, and verification. Industry-standard tools are used for design entry and simulation (for example, Synopsys FPGA Express), while Xilinx provides proprietary architecture-specific tools for implementation.

The Xilinx development system is integrated under the Xilinx Design Manager (XDM™) software, providing designers with a common user interface regardless of their choice of entry and verification tools. The XDM software simplifies the selection of implementation options with pull-down menus and on-line help.

Application programs ranging from schematic capture to Placement and Routing (PAR) can be accessed through the XDM software. The program command sequence is generated prior to execution, and stored for documentation.

Several advanced software features facilitate Virtex-E design. RPMs, for example, are schematic-based macros with relative location constraints to guide their placement. They help ensure optimal implementation of common functions.

For HDL design entry, the Xilinx FPGA Foundation development system provides interfaces to the following synthesis design environments.

- Synopsys (FPGA Compiler, FPGA Express)
- Exemplar (Spectrum)
- Synplicity (Synplify)

For schematic design entry, the Xilinx FPGA Foundation and Alliance development system provides interfaces to the following schematic-capture design environments.

- Mentor Graphics V8 (Design Architect, QuickSim II)
- Viewlogic Systems (Viewdraw)

Third-party vendors support many other environments.

**Table 7: IDCODEs Assigned to Virtex-E FPGAs**

FPGA	IDCODE
XCV405EM	v0C28093h
XCV812EM	v0C38093h

**Note:**

Attempting to load an incorrect bitstream causes configuration to fail and can damage the device.

### Including Boundary Scan in a Design

Since the boundary scan pins are dedicated, no special element needs to be added to the design unless an internal data register (USER1 or USER2) is desired.

If an internal data register is used, insert the boundary scan symbol and connect the necessary pins as appropriate.

A standard interface-file specification, Electronic Design Interchange Format (EDIF), simplifies file transfers into and out of the development system.

Virtex-E FPGAs are supported by a unified library of standard functions. This library contains over 400 primitives and macros, ranging from 2-input AND gates to 16-bit accumulators, and includes arithmetic functions, comparators, counters, data registers, decoders, encoders, I/O functions, latches, Boolean functions, multiplexers, shift registers, and barrel shifters.

The “soft macro” portion of the library contains detailed descriptions of common logic functions, but does not contain any partitioning or placement information. The performance of these macros depends, therefore, on the partitioning and placement obtained during implementation.

RPMs, on the other hand, do contain predetermined partitioning and placement information that permits optimal implementation of these functions. Users can create their own library of soft macros or RPMs based on the macros and primitives in the standard library.

The design environment supports hierarchical design entry, with high-level schematics that comprise major functional blocks, while lower-level schematics define the logic in these blocks. These hierarchical design elements are automatically combined by the implementation tools. Different design entry tools can be combined within a hierarchical design, thus allowing the most convenient entry method to be used for each portion of the design.

## Design Implementation

The place-and-route tools (PAR) automatically provide the implementation flow described in this section. The partitioner takes the EDIF net list for the design and maps the logic into the architectural resources of the FPGA (CLBs and IOBs, for example). The placer then determines the best locations for these blocks based on their intercon-

tions and the desired performance. Finally, the router interconnects the blocks.

The PAR algorithms support fully automatic implementation of most designs. For demanding applications, however, the user can exercise various degrees of control over the process. User partitioning, placement, and routing information is optionally specified during the design-entry process. The implementation of highly structured designs can benefit greatly from basic floor planning.

The implementation software incorporates Timing Wizard® timing-driven placement and routing. Designers specify timing requirements along entire paths during design entry. The timing path analysis routines in PAR then recognize these user-specified requirements and accommodate them.

Timing requirements are entered on a schematic in a form directly relating to the system requirements, such as the targeted clock frequency, or the maximum allowable delay between two registers. In this way, the overall performance of the system along entire signal paths is automatically tailored to user-generated specifications. Specific timing information for individual nets is unnecessary.

## Configuration

Virtex-E devices are configured by loading configuration data into the internal configuration memory. Note that attempting to load an incorrect bitstream causes configuration to fail and can damage the device.

Some of the pins used for configuration are dedicated pins, while others can be re-used as general purpose inputs and outputs once configuration is complete.

The following are dedicated pins:

- Mode pins (M2, M1, M0)
- Configuration clock pin (CCLK)
- PROGRAM pin
- DONE pin
- Boundary-scan pins (TDI, TDO, TMS, TCK)

Depending on the configuration mode chosen, CCLK can be an output generated by the FPGA, or it can be generated externally and provided to the FPGA as an input. The PROGRAM pin must be pulled High prior to reconfiguration. Note that some configuration pins can act as outputs. For correct operation, these pins require a  $V_{CCO}$  of 3.3 V to permit LVTTTL operation. All of the pins affected are in banks 2

Table 8: Configuration Codes

Configuration Mode	M2	M1	M0	CCLK Direction	Data Width	Serial D <sub>out</sub>	Configuration Pull-ups
Master-serial mode	0	0	0	Out	1	Yes	No
Boundary-scan mode	1	0	1	N/A	1	No	No
SelectMAP mode	1	1	0	In	8	No	No
Slave-serial mode	1	1	1	In	1	Yes	No
Master-serial mode	1	0	0	Out	1	Yes	Yes

## Design Verification

In addition to conventional software simulation, FPGA users can use in-circuit debugging techniques. Because Xilinx devices are infinitely reprogrammable, designs can be verified in real time without the need for extensive sets of software simulation vectors.

The development system supports both software simulation and in-circuit debugging techniques. For simulation, the system extracts the post-layout timing information from the design database, and back-annotates this information into the net list for use by the simulator. Alternatively, the user can verify timing-critical portions of the design using the TRCE® static timing analyzer.

For in-circuit debugging, an optional download and read-back cable is available. This cable connects the FPGA in the target system to a PC or workstation. After downloading the design into the FPGA, the designer can single-step the logic, readback the contents of the flip-flops, and so observe the internal logic state. Simple modifications can be downloaded into the system in a matter of minutes.

or 3. The configuration pins needed for SelectMap (CS, Write) are located in bank 1.

## Configuration Modes

Virtex-E supports the following four configuration modes.

- Slave-serial mode
- Master-serial mode
- SelectMAP mode
- Boundary-scan mode (JTAG)

The Configuration mode pins (M2, M1, M0) select among these configuration modes with the option in each case of having the IOB pins either pulled up or left floating prior to configuration. The selection codes are listed in Table 8.

Configuration through the boundary-scan port is always available, independent of the mode selection. Selecting the boundary-scan mode simply turns off the other modes. The three mode pins have internal pull-up resistors, and default to a logic High if left unconnected. However, it is recommended to drive the configuration mode pins externally.

Table 8: Configuration Codes

Configuration Mode	M2	M1	M0	CCLK Direction	Data Width	Serial D <sub>out</sub>	Configuration Pull-ups
Boundary-scan mode	0	0	1	N/A	1	No	Yes
SelectMAP mode	0	1	0	In	8	No	Yes
Slave-serial mode	0	1	1	In	1	Yes	Yes

Table 9 lists the total number of bits required to configure each device.

Table 9: Virtex-E Bitstream Lengths

Device	# of Configuration Bits
XCV405E	3,430,400
XCV812E	6,519,648

### Slave-Serial Mode

In slave-serial mode, the FPGA receives configuration data in bit-serial form from a serial PROM or other source of serial configuration data. The serial bitstream must be set up at the DIN input pin a short time before each rising edge of an externally generated CCLK.

For more detailed information on serial PROMs see the PROM data sheet at <http://www.xilinx.com/bvdocs/publications/ds026.pdf>.

Multiple FPGAs can be daisy-chained for configuration from a single source. After a particular FPGA has been config-

ured, the data for the next device is routed to the DOUT pin. Data on the DOUT pin changes on the rising edge of CCLK.

The change of DOUT on the rising edge of CCLK differs from previous families but does not cause a problem for mixed configuration chains. This change was made to improve serial configuration rates for Virtex and Virtex-E only chains.

Figure 13 shows a full master/slave system. A Virtex-E device in slave-serial mode should be connected as shown in the right-most device.

Slave-serial mode is selected by applying <111> or <011> to the mode pins (M2, M1, M0). A weak pull-up on the mode pins makes slave-serial the default mode if the pins are left unconnected. However, it is recommended to drive the configuration mode pins externally. Figure 14 shows slave-serial mode programming switching characteristics.

Table 10 provides more detail about the characteristics shown in Figure 14. Configuration must be delayed until the INIT pins of all daisy-chained FPGAs are High.

Table 10: Master/Slave Serial Mode Programming Switching

	Description	Figure References	Symbol	Values	Units
CCLK	DIN setup/hold, slave mode	1/2	T <sub>DCC</sub> /T <sub>CCD</sub>	5.0/0.0	ns, min
	DIN setup/hold, master mode	1/2	T <sub>DSCK</sub> /T <sub>CKDS</sub>	5.0/0.0	ns, min
	DOUT	3	T <sub>CCO</sub>	12.0	ns, max
	High time	4	T <sub>CCH</sub>	5.0	ns, min
	Low time	5	T <sub>CCL</sub>	5.0	ns, min
	Maximum Frequency		F <sub>CC</sub>	66	MHz, max
	Frequency Tolerance, master mode with respect to nominal			+45% -30%	

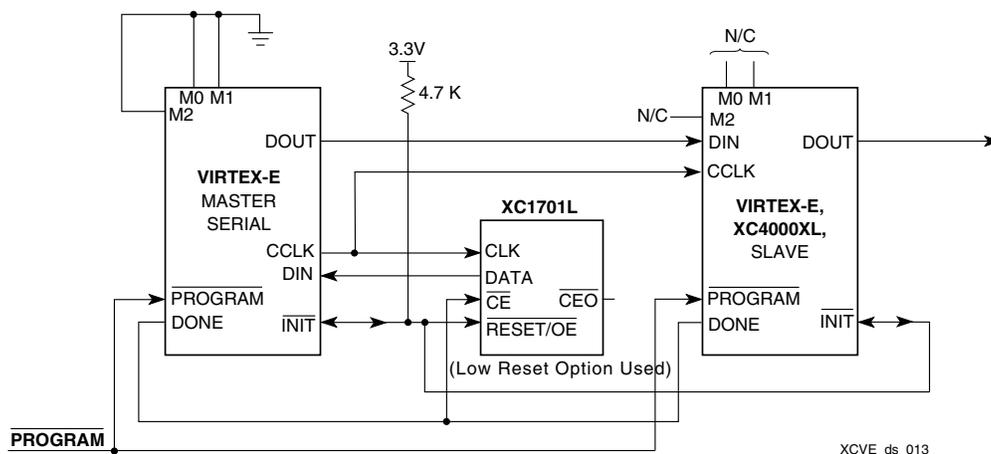


Figure 13: Master/Slave Serial Mode Circuit Diagram

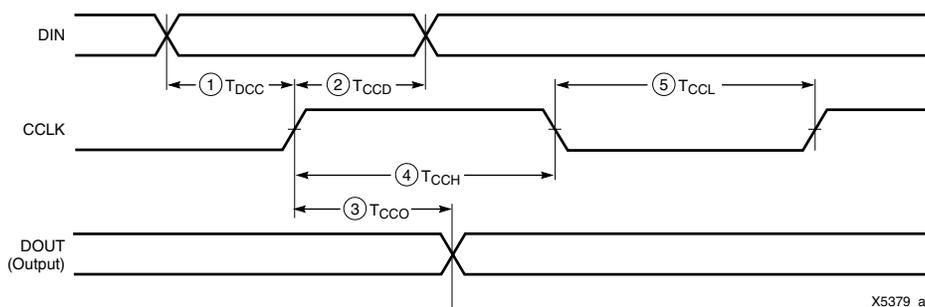


Figure 14: Slave-Serial Mode Programming Switching Characteristics

### Master-Serial Mode

In master-serial mode, the CCLK output of the FPGA drives a Xilinx Serial PROM that feeds bit-serial data to the DIN input. The FPGA accepts this data on each rising CCLK edge. After the FPGA has been loaded, the data for the next device in a daisy-chain is presented on the DOUT pin after the rising CCLK edge.

The interface is identical to slave-serial except that an internal oscillator is used to generate the configuration clock (CCLK). A wide range of frequencies can be selected for CCLK which always starts at a slow default frequency. Configuration bits then switch CCLK to a higher frequency for the remainder of the configuration. Switching to a lower frequency is prohibited.

The CCLK frequency is set using the ConfigRate option in the bitstream generation software. The maximum CCLK fre-

quency that can be selected is 60 MHz. When selecting a CCLK frequency, ensure that the serial PROM and any daisy-chained FPGAs are fast enough to support the clock rate.

On power-up, the CCLK frequency is approximately 2.5 MHz. This frequency is used until the ConfigRate bits have been loaded when the frequency changes to the selected ConfigRate. Unless a different frequency is specified in the design, the default ConfigRate is 4 MHz.

Figure 13 shows a full master/slave system. In this system, the left-most device operates in master-serial mode. The remaining devices operate in slave-serial mode. The SPROM RESET pin is driven by INIT, and the CE input is driven by DONE. There is the potential for contention on the DONE pin, depending on the start-up sequence options chosen.

The sequence of operations necessary to configure a Virtex-E FPGA serially appears in Figure 15.

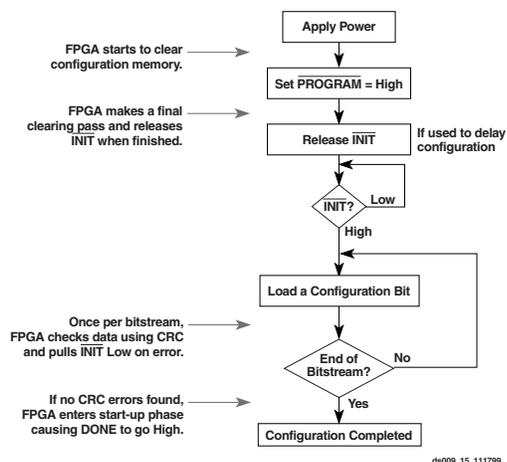


Figure 15: Serial Configuration Flowchart

Figure 16 shows the timing of master-serial configuration. Master-serial mode is selected by a <000> or <100> on the mode pins (M2, M1, M0). Table 10 shows the timing information for Figure 16

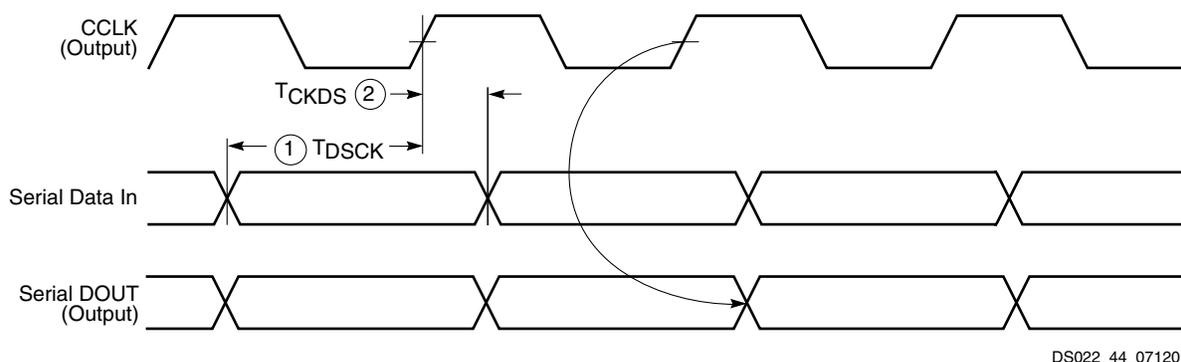


Figure 16: Master-Serial Mode Programming Switching Characteristics

At power-up, V<sub>CC</sub> must rise from 1.0 V to V<sub>CC</sub> min in less than 50 ms, otherwise delay configuration by pulling PROGRAM Low until V<sub>CC</sub> is valid.

### SelectMAP Mode

The SelectMAP mode is the fastest configuration option. Byte-wide data is written into the FPGA with a BUSY flag controlling the flow of data.

An external data source provides a byte stream, CCLK, a Chip Select (CS) signal and a Write signal (WRITE). If BUSY is asserted (High) by the FPGA, the data must be held until BUSY goes Low.

Data can also be read using the SelectMAP mode. If WRITE is not asserted, configuration data is read out of the FPGA as part of a readback operation.

After configuration, the pins of the SelectMAP port can be used as additional user I/O. Alternatively, the port can be retained to permit high-speed 8-bit readback.

Retention of the SelectMAP port is selectable on a design-by-design basis when the bitstream is generated. If

retention is selected, PROHIBIT constraints are required to prevent SelectMAP-port pins from being used as user I/O.

Multiple Virtex-E FPGAs can be configured using the SelectMAP mode, and be made to start-up simultaneously. To configure multiple devices in this way, wire the individual CCLK, Data, WRITE, and BUSY pins of all the devices in parallel. The individual devices are loaded separately by asserting the CS pin of each device in turn and writing the appropriate data. See Table 11 for SelectMAP Write Timing Characteristics.

### Write

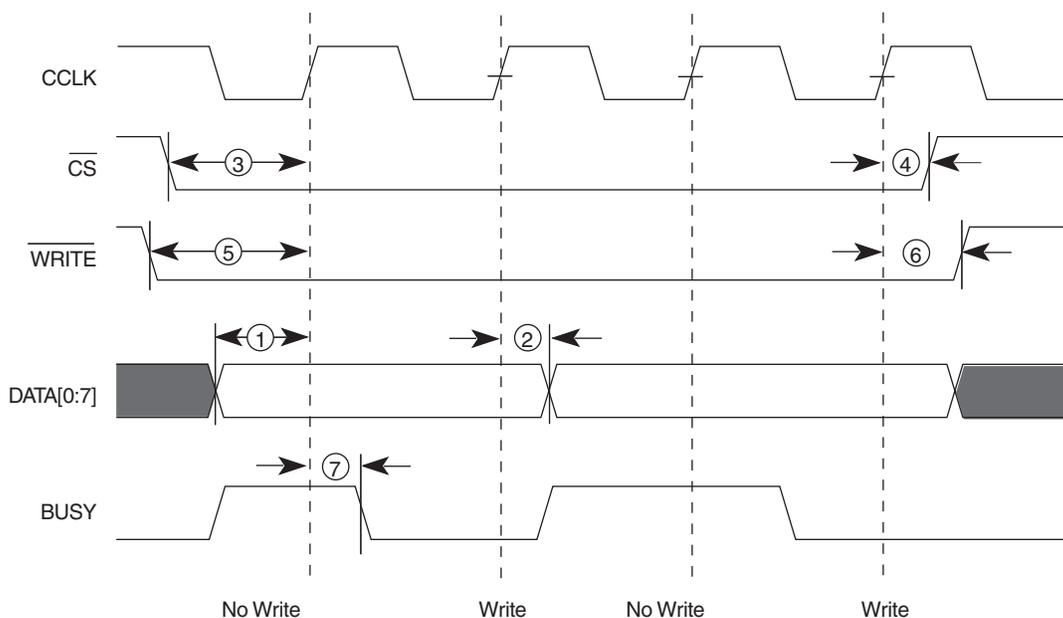
Write operations send packets of configuration data into the FPGA. The sequence of operations for a multi-cycle write operation is shown below. Note that a configuration packet can be split into many such sequences. The packet does not have to complete within one assertion of CS, illustrated in Figure 17.

1. Assert WRITE and CS Low. Note that when CS is asserted on successive CCLKs, WRITE must remain

- either asserted or de-asserted. Otherwise an abort is initiated, as described below.
- Drive data onto D[7:0]. Note that to avoid contention, the data source should not be enabled while  $\overline{CS}$  is Low and  $\overline{WRITE}$  is High. Similarly, while  $\overline{WRITE}$  is High, no more than one  $\overline{CS}$  should be asserted.
  - At the rising edge of CCLK: If BUSY is Low, the data is accepted on this clock. If BUSY is High (from a previous write), the data is not accepted. Acceptance instead occurs on the first clock after BUSY goes Low, and the data must be held until this has happened.
  - Repeat steps 2 and 3 until all the data has been sent.
  - De-assert  $\overline{CS}$  and  $\overline{WRITE}$ .

Table 11: SelectMAP Write Timing Characteristics

	Description		Symbol	Values	Units
CCLK	D <sub>0-7</sub> Setup/Hold	1/2	T <sub>SMDC</sub> /T <sub>SMCCD</sub>	5.0 / 1.7	ns, min
	$\overline{CS}$ Setup/Hold	3/4	T <sub>SMCSC</sub> /T <sub>SMCCS</sub>	7.0 / 1.7	ns, min
	$\overline{WRITE}$ Setup/Hold	5/6	T <sub>SMCCW</sub> /T <sub>SMWCC</sub>	7.0 / 1.7	ns, min
	BUSY Propagation Delay	7	T <sub>SMCKBY</sub>	12.0	ns, max
	Maximum Frequency		F <sub>CC</sub>	66	MHz, max
	Maximum Frequency with no handshake		F <sub>CCNH</sub>	50	MHz, max



DS022\_45\_071702

Figure 17: Write Operations

A flowchart for the write operation appears in [Figure 18](#). Note that if CCLK is slower than  $f_{CCNH}$ , the FPGA never asserts BUSY. In this case, the above handshake is unnecessary, and data can simply be entered into the FPGA every CCLK cycle.

### Abort

During a given assertion of  $\overline{CS}$ , the user cannot switch from a write to a read, or vice-versa. This action causes the cur-

rent packet command to be aborted. The device remains BUSY until the aborted operation has completed. Following an abort, data is assumed to be unaligned to word boundaries, and the FPGA requires a new synchronization word prior to accepting any new packets.

To initiate an abort during a write operation, de-assert  $\overline{WRITE}$ . At the rising edge of CCLK, an abort is initiated, as shown in [Figure 19](#).

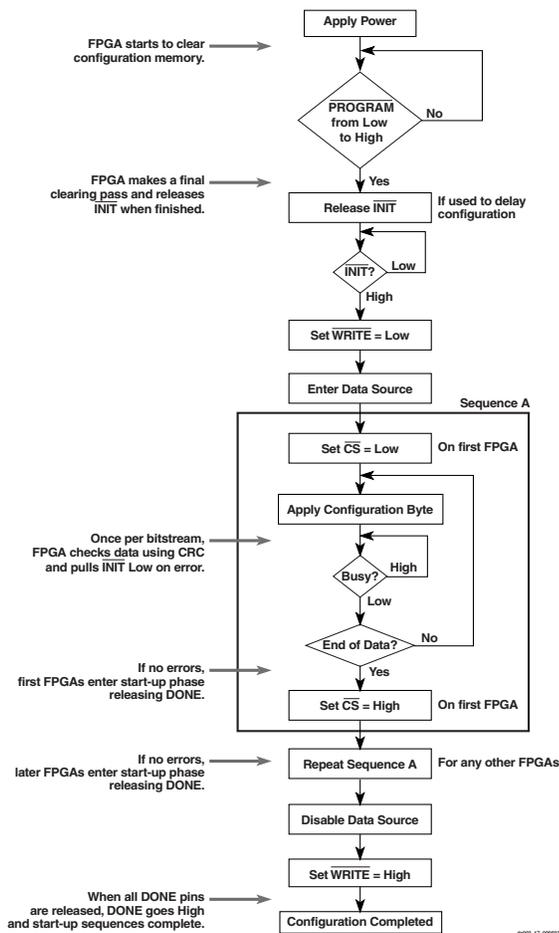


Figure 18: SelectMAP Flowchart for Write Operations

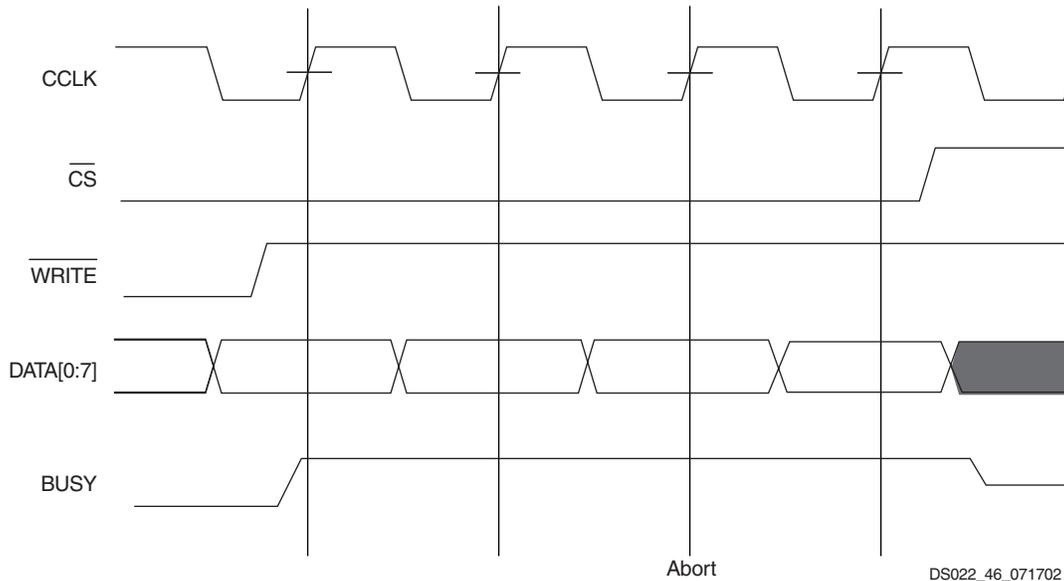


Figure 19: SelectMAP Write Abort Waveforms

**Boundary-Scan Mode**

In the boundary-scan mode, configuration is done through the IEEE 1149.1 Test Access Port. Note that the

PROGRAM pin must be pulled High prior to reconfiguration. A Low on the PROGRAM pin resets the TAP controller and no JTAG operations can be performed.

Configuration through the TAP uses the CFG\_IN instruction. This instruction allows data input on TDI to be converted into data packets for the internal configuration bus.

The following steps are required to configure the FPGA through the boundary-scan port (when using TCK as a start-up clock).

1. Load the CFG\_IN instruction into the boundary-scan instruction register (IR)
2. Enter the Shift-DR (SDR) state
3. Shift a configuration bitstream into TDI
4. Return to Run-Test-Idle (RTI)
5. Load the JSTART instruction into IR
6. Enter the SDR state
7. Clock TCK through the startup sequence
8. Return to RTI

Configuration and readback via the TAP is always available. The boundary-scan mode is selected by a <101> or <001>

on the mode pins (M2, M1, M0). For details on TAP characteristics, refer to XAPP139.

## Configuration Sequence

The configuration of Virtex-E devices is a three-phase process. First, the configuration memory is cleared. Next, configuration data is loaded into the memory, and finally, the logic is activated by a start-up process.

Configuration is automatically initiated on power-up unless it is delayed by the user, as described below. The configuration process can also be initiated by asserting PROGRAM. The end of the memory-clearing phase is signalled by  $\overline{\text{INIT}}$  going High, and the completion of the entire process is signalled by DONE going High.

The power-up timing of configuration signals is shown in Figure 20.

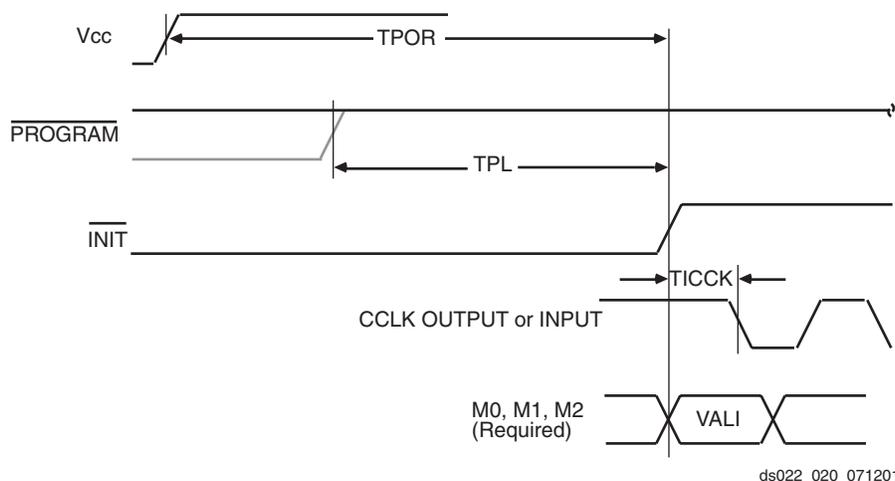


Figure 20: Power-Up Timing Configuration Signals

The corresponding timing characteristics are listed in Table 12.

Table 12: Power-up Timing Characteristics

Description	Symbol	Value	Units
Power-on Reset <sup>1</sup>	$T_{\text{POR}}$	2.0	ms, max
Program Latency	$T_{\text{PL}}$	100.0	$\mu\text{s}$ , max
CCLK (output) Delay	$T_{\text{ICCK}}$	0.5	$\mu\text{s}$ , min
		4.0	$\mu\text{s}$ , max
Program Pulse Width	$T_{\text{PROGRAM}}$	300	ns, min

### Notes:

1.  $T_{\text{POR}}$  delay is the initialization time required after  $V_{\text{CCINT}}$  reaches the recommended operating voltage.

## Delaying Configuration

$\overline{\text{INIT}}$  can be held Low using an open-drain driver. An open-drain is required since  $\overline{\text{INIT}}$  is a bidirectional open-drain pin that is held Low by the FPGA while the configuration memory is being cleared. Extending the time that the pin is Low causes the configuration sequencer to wait. Thus, configuration is delayed by preventing entry into the phase where data is loaded.

## Start-Up Sequence

The default Start-up sequence is that one CCLK cycle after DONE goes High, the global 3-state signal (GTS) is released. This permits device outputs to turn on as necessary.

One CCLK cycle later, the Global Set/Reset (GSR) and Global Write Enable (GWE) signals are released. This permits

the internal storage elements to begin changing state in response to the logic and the user clock.

The relative timing of these events can be changed. In addition, the GTS, GSR, and GWE events can be made depen-

## **Readback**

The configuration data stored in the Virtex-E configuration memory can be readback for verification. Along with the configuration data it is possible to readback the contents all flip-flops/latches, LUT RAMs, and block RAMs. This capa-

bility is used for real-time debugging. For more detailed information, see application note XAPP138 "Virtex FPGA Series Configuration and Readback".

ment on the DONE pins of multiple devices all going High, forcing the devices to start synchronously. The sequence can also be paused at any stage until lock has been achieved on any or all DLLs.

## Design Considerations

This section contains more detailed design information on the following features.

- Delay-Locked Loop . . . see [page 20](#)
- BlockRAM . . . see [page 24](#)
- Select/O . . . see [page 31](#)

## Using DLLs

The Virtex-E FPGA series provides up to eight fully digital dedicated on-chip Delay-Locked Loop (DLL) circuits which provide zero propagation delay, low clock skew between output clock signals distributed throughout the device, and advanced clock domain control. These dedicated DLLs can be used to implement several circuits which improve and simplify system level design.

## Introduction

As FPGAs grow in size, quality on-chip clock distribution becomes increasingly important. Clock skew and clock delay impact device performance and the task of managing clock skew and clock delay with conventional clock trees becomes more difficult in large devices. The Virtex-E series of devices resolve this potential problem by providing up to eight fully digital dedicated on-chip DLL circuits which provide zero propagation delay and low clock skew between output clock signals distributed throughout the device.

Each DLL can drive up to two global clock routing networks within the device. The global clock distribution network minimizes clock skews due to loading differences. By monitoring a sample of the DLL output clock, the DLL can compensate for the delay on the routing network, effectively eliminating the delay from the external input port to the individual clock loads within the device.

In addition to providing zero delay with respect to a user source clock, the DLL can provide multiple phases of the source clock. The DLL can also act as a clock doubler or it can divide the user source clock by up to 16.

Clock multiplication gives the designer a number of design alternatives. For instance, a 50 MHz source clock doubled by the DLL can drive an FPGA design operating at 100 MHz. This technique can simplify board design because the clock path on the board no longer distributes such a high-speed signal. A multiplied clock also provides designers the option of time-domain-multiplexing, using one circuit twice per clock cycle, consuming less area than two copies of the same circuit. Two DLLs in can be connected in series to increase the effective clock multiplication factor to four.

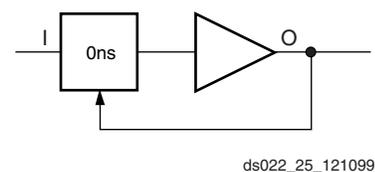
The DLL can also act as a clock mirror. By driving the DLL output off-chip and then back in again, the DLL can be used to de-skew a board level clock between multiple devices.

In order to guarantee the system clock establishes prior to the device “waking up,” the DLL can delay the completion of the device configuration process until after the DLL achieves lock.

By taking advantage of the DLL to remove on-chip clock delay, the designer can greatly simplify and improve system level design involving high-fanout, high-performance clocks.

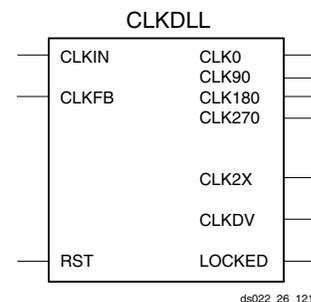
## Library DLL Symbols

**Figure 21** shows the simplified Xilinx library DLL macro symbol, BUFGDLL. This macro delivers a quick and efficient way to provide a system clock with zero propagation delay throughout the device. **Figure 22** and **Figure 23** show the two library DLL primitives. These symbols provide access to the complete set of DLL features when implementing more complex applications.



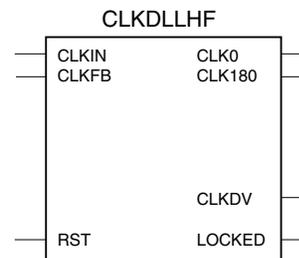
ds022\_25\_121099

**Figure 21: Simplified DLL Macro Symbol BUFGDLL**



ds022\_26\_121099

**Figure 22: Standard DLL Symbol CLKDLL**



ds022\_027\_121099

**Figure 23: High Frequency DLL Symbol**

## BUFGDLL Pin Descriptions

Use the BUFGDLL macro as the simplest way to provide zero propagation delay for a high-fanout on-chip clock from an external input. This macro uses the IBUFG, CLKDLL and BUFG primitives to implement the most basic DLL application as shown in Figure 24.

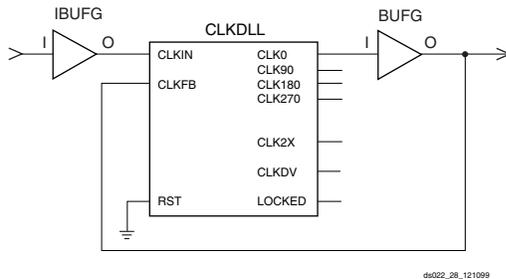


Figure 24: BUFGDLL Schematic

This symbol does not provide access to the advanced clock domain controls or to the clock multiplication or clock division features of the DLL. This symbol also does not provide access to the RST, or LOCKED pins of the DLL. For access to these features, a designer must use the library DLL primitives described in the following sections.

### Source Clock Input — I

The I pin provides the user source clock, the clock signal on which the DLL operates, to the BUFGDLL. For the BUFGDLL macro the source clock frequency must fall in the low frequency range as specified in the data sheet. The BUFGDLL requires an external signal source clock. Therefore, only an external input port can source the signal that drives the BUFGDLL I pin.

### Clock Output — O

The clock output pin O represents a delay-compensated version of the source clock (I) signal. This signal, sourced by a global clock buffer BUFG symbol, takes advantage of the dedicated global clock routing resources of the device.

The output clock has a 50-50 duty cycle unless you deactivate the duty cycle correction property.

## CLKDLL Primitive Pin Descriptions

The library CLKDLL primitives provide access to the complete set of DLL features needed when implementing more complex applications with the DLL.

### Source Clock Input — CLKIN

The CLKIN pin provides the user source clock (the clock signal on which the DLL operates) to the DLL. The CLKIN frequency must fall in the ranges specified in the data sheet. A global clock buffer (BUFG) driven from another CLKDLL, one of the global clock input buffers (IBUFG), or an IO\_LVDS\_DLL pin on the same edge of the device (top or bottom) must source this clock signal. There are four IO\_LVDS\_DLL input pins that can be used as inputs to the

DLLs. This makes a total of eight usable input pins for DLLs in the Virtex-E family.

### Feedback Clock Input — CLKFB

The DLL requires a reference or feedback signal to provide the delay-compensated output. Connect only the CLK0 or CLK2X DLL outputs to the feedback clock input (CLKFB) pin to provide the necessary feedback to the DLL. The feedback clock input can also be provided through one of the following pins.

IBUFG - Global Clock Input Pad

IO\_LVDS\_DLL - the pin adjacent to IBUFG

If an IBUFG sources the CLKFB pin, the following special rules apply.

1. An external input port must source the signal that drives the IBUFG I pin.
2. The CLK2X output must feedback to the device if both the CLK0 and CLK2X outputs are driving off chip devices.
3. That signal must directly drive only OBUFs and nothing else.

These rules enable the software determine which DLL clock output sources the CLKFB pin.

### Reset Input — RST

When the reset pin RST activates the LOCKED signal deactivates within four source clock cycles. The RST pin, active High, must either connect to a dynamic signal or tied to ground. As the DLL delay taps reset to zero, glitches can occur on the DLL clock output pins. Activation of the RST pin can also severely affect the duty cycle of the clock output pins. Furthermore, the DLL output clocks no longer de-skew with respect to one another. For these reasons, rarely use the reset pin unless re-configuring the device or changing the input frequency.

### 2x Clock Output — CLK2X

The output pin CLK2X provides a frequency-doubled clock with an automatic 50/50 duty-cycle correction. Until the CLKDLL has achieved lock, the CLK2X output appears as a 1x version of the input clock with a 25/75 duty cycle. This behavior allows the DLL to lock on the correct edge with respect to source clock. This pin is not available on the CLKDLLHF primitive.

### Clock Divide Output — CLKDV

The clock divide output pin CLKDV provides a lower frequency version of the source clock. The CLKDV\_DIVIDE property controls CLKDV such that the source clock is divided by N where N is either 1.5, 2, 2.5, 3, 4, 5, 8, or 16.

This feature provides automatic duty cycle correction such that the CLKDV output pin always has a 50/50 duty cycle, with the exception of noninteger divides in HF mode, where the duty cycle is 1/3 for N=1.5 and 2/5 for N=2.5.