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

— OBSOLETE — OBSOLETE — OBSOLETE — OBSOLETE —

Spartan-IIE FPGA Family Data Sheet

DS077 August 9, 2013 **Product Specification**

This document includes all four modules of the Spartan®-IIE FPGA data sheet.

Module 1: Introduction and Ordering Information

DS077-1 (v3.0) August 9, 2013

- Introduction
- Features
- General Overview
- Product Availability
- User I/O Chart
- Ordering Information

Module 2: Functional Description

DS077-2 (v3.0) August 9, 2013

- Architectural Description
  - Spartan-IIE Array
  - Input/Output Block
  - Configurable Logic Block
  - Block RAM
  - Clock Distribution: Delay-Locked Loop
  - Boundary Scan
- Development System
- Configuration

Module 3: DC and Switching Characteristics

DS077-3 (v3.0) August 9, 2013

- DC Specifications
  - Absolute Maximum Ratings
  - Recommended Operating Conditions
  - DC Characteristics
  - Power-On Requirements
  - DC Input and Output Levels
- Switching Characteristics
  - Pin-to-Pin Parameters
  - IOB Switching Characteristics
  - Clock Distribution Characteristics
  - DLL Timing Parameters
  - CLB Switching Characteristics
  - Block RAM Switching Characteristics
  - TBUF Switching Characteristics
  - JTAG Switching Characteristics
  - Configuration Switching Characteristics

Module 4: Pinout Tables

DS077-4 (v3.0) August 9, 2013

- Pin Definitions
- Pinout Tables

IMPORTANT NOTE: *The Spartan-IIE FPGA data sheet is in four modules. Each module has its own Revision History at the end. Use the PDF "Bookmarks" for easy navigation in this volume.*

Spartan-IIE FPGA Family: Introduction and Ordering Information

DS077-1 (v3.0) August 9, 2013

Product Specification

Introduction

The Spartan®-IIE Field-Programmable Gate Array family gives users high performance, abundant logic resources, and a rich feature set, all at an exceptionally low price. The seven-member family offers densities ranging from 50,000 to 600,000 system gates, as shown in Table 1. System performance is supported beyond 200 MHz.

Features include block RAM (to 288K bits), distributed RAM (to 221,184 bits), 19 selectable I/O standards, and four DLLs (Delay-Locked Loops). Fast, predictable interconnect means that successive design iterations continue to meet timing requirements.

The Spartan-IIE family is a superior alternative to mask-programmed ASICs. The FPGA avoids the initial cost, lengthy development cycles, and inherent risk of conventional ASICs. Also, FPGA programmability permits design upgrades in the field with no hardware replacement necessary (impossible with ASICs).

Features

- Second generation ASIC replacement technology
  - Densities as high as 15,552 logic cells with up to 600,000 system gates
  - Streamlined features based on Virtex®-E FPGA architecture
  - Unlimited in-system reprogrammability
  - Very low cost
  - Cost-effective 0.15 micron technology
- System level features
  - SelectRAM™ hierarchical memory:
    - 16 bits/LUT distributed RAM
    - Configurable 4K-bit true dual-port block RAM
    - Fast interfaces to external RAM
  - Fully 3.3V PCI compliant to 64 bits at 66 MHz and CardBus compliant
  - Low-power segmented routing architecture
  - Dedicated carry logic for high-speed arithmetic
  - Efficient multiplier support
  - Cascade chain for wide-input functions
  - Abundant registers/latches with enable, set, reset
  - Four dedicated DLLs for advanced clock control
    - Eliminate clock distribution delay
    - Multiply, divide, or phase shift
  - Four primary low-skew global clock distribution nets
  - IEEE 1149.1 compatible boundary scan logic
- Versatile I/O and packaging
  - Pb-free package options
  - Low-cost packages available in all densities
  - Family footprint compatibility in common packages
  - 19 high-performance interface standards
    - LVTTL, LVCMOS, HSTL, SSTL, AGP, CTT, GTL
    - LVDS and LVPECL differential I/O
  - Up to 205 differential I/O pairs that can be input, output, or bidirectional
  - Hot swap I/O (CompactPCI friendly)
- Core logic powered at 1.8V and I/Os powered at 1.5V, 2.5V, or 3.3V
- Fully supported by powerful Xilinx® ISE® development system
  - Fully automatic mapping, placement, and routing
  - Integrated with design entry and verification tools
  - Extensive IP library including DSP functions and soft processors

Table 1: **Spartan-IIE FPGA Family Members**

Device	Logic Cells	Typical System Gate Range (Logic and RAM)	CLB Array (R x C)	Total CLBs	Maximum Available User I/O(1)	Maximum Differential I/O Pairs	Distributed RAM Bits	Block RAM Bits
XC2S50E	1,728	23,000 - 50,000	16 x 24	384	182	83	24,576	32K
XC2S100E	2,700	37,000 - 100,000	20 x 30	600	202	86	38,400	40K
XC2S150E	3,888	52,000 - 150,000	24 x 36	864	265	114	55,296	48K
XC2S200E	5,292	71,000 - 200,000	28 x 42	1,176	289	120	75,264	56K
XC2S300E	6,912	93,000 - 300,000	32 x 48	1,536	329	120	98,304	64K
XC2S400E	10,800	145,000 - 400,000	40 x 60	2,400	410	172	153,600	160K
XC2S600E	15,552	210,000 - 600,000	48 x 72	3,456	514	205	221,184	288K

Notes:

1. User I/O counts include the four global clock/user input pins. See details in Table 2, page 5

General Overview

The Spartan-IIE family of FPGAs have a regular, flexible, programmable architecture of Configurable Logic Blocks (CLBs), surrounded by a perimeter of programmable Input/Output Blocks (IOBs). There are four Delay-Locked Loops (DLLs), one at each corner of the die. Two columns of block RAM lie on opposite sides of the die, between the CLBs and the IOB columns. The XC2S400E has four columns and the XC2S600E has six columns of block RAM. These functional elements are interconnected by a powerful hierarchy of versatile routing channels (see Figure 1).

Spartan-IIE FPGAs are customized by loading configuration data into internal static memory cells. Unlimited reprogramming cycles are possible with this approach. Stored values in these cells determine logic functions and interconnections implemented in the FPGA. Configuration data can be read from an external serial PROM (master serial mode), or written into the FPGA in slave serial, slave parallel, or Boundary Scan modes. Xilinx offers multiple types of low-cost configuration solutions including the Platform Flash in-system programmable configuration PROMs.

Spartan-IIE FPGAs are typically used in high-volume applications where the versatility of a fast programmable solution adds benefits. Spartan-IIE FPGAs are ideal for shortening product development cycles while offering a cost-effective solution for high volume production.

Spartan-IIE FPGAs achieve high-performance, low-cost operation through advanced architecture and semiconductor technology. Spartan-IIE devices provide system clock rates beyond 200 MHz. In addition to the conventional benefits of high-volume programmable logic solutions, Spartan-IIE FPGAs also offer on-chip synchronous single-port and dual-port RAM (block and distributed form), DLL clock drivers, programmable set and reset on all flip-flops, fast carry logic, and many other features.

Spartan-IIE Family Compared to Spartan-II Family

- Higher density and more I/O
- Higher performance
- Unique pinouts in cost-effective packages
- Differential signaling
  - LVDS, Bus LVDS, LVPECL
- V_{CCINT} = 1.8V
  - Lower power
  - 5V tolerance with external resistor
  - 3V tolerance directly
- PCI, LVTTL, and LVCMOS2 input buffers powered by V_{CCO} instead of V_{CCINT}
- Unique larger bitstream

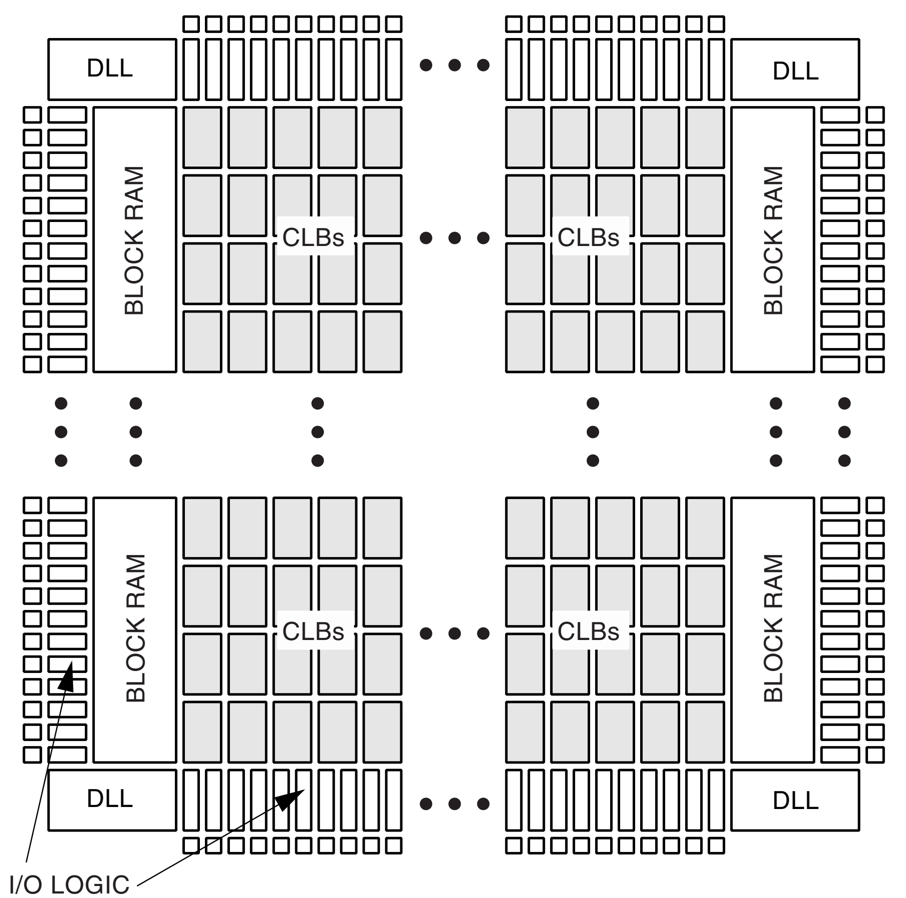


Figure 1: **Basic Spartan-IIE Family FPGA Block Diagram**

Spartan-IIE Product Availability

Table 2 shows the maximum user I/Os available on the device and the number of user I/Os available for each device/package combination.

Table 2: **Spartan-IIE FPGA User I/O Chart**

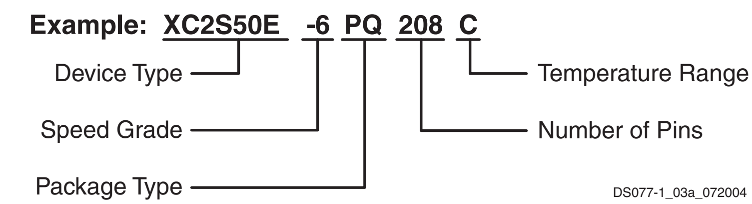
Device	Maximum User I/O	Available User I/O According to Package Type				
		TQ144 TQG144	PQ208 PQG208	FT256 FTG256	FG456 FGG456	FG676 FGG676
XC2S50E	182	102	146	182	-	-
XC2S100E	202	102	146	182	202	-
XC2S150E	265	-	146	182	265	-
XC2S200E	289	-	146	182	289	-
XC2S300E	329	-	146	182	329	-
XC2S400E	410	-	-	182	329	410
XC2S600E	514	-	-	-	329	514

Notes:
1. User I/O counts include the four global clock/user input pins.

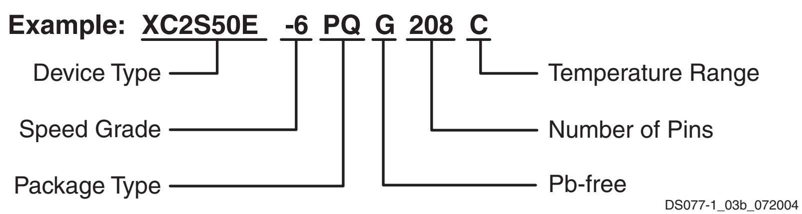
Ordering Information

Spartan-IIE devices are available in both standard and Pb-free packaging options for all device/package combinations. The Pb-free packages include a special "G" character in the ordering code.

Standard Packaging



Pb-Free Packaging



Device Ordering Options

Device
XC2S50E
XC2S100E
XC2S150E
XC2S200E
XC2S300E
XC2S400E
XC2S600E

Speed Grade	
-6	Standard Performance
-7	Higher Performance(1)

Package Type / Number of Pins	
TQ(G)144	144-pin Plastic Thin QFP
PQ(G)208	208-pin Plastic QFP
FT(G)256	256-ball Fine Pitch BGA
FG(G)456	456-ball Fine Pitch BGA
FG(G)676	676-ball Fine Pitch BGA

Temperature Range (T_J)(2)	
C = Commercial	0°C to +85°C
I = Industrial	–40°C to +100°C

Notes:
1. The -7 speed grade is exclusively available in the Commercial temperature range.
2. See www.xilinx.com for information on automotive temperature range devices.

Device Part Marking

Figure 2 is a top marking example for Spartan-IIE FPGAs in the quad-flat packages. The markings for BGA packages are nearly identical to those for the quad-flat packages, except that the marking is rotated with respect to the ball A1 indicator.

The "7C" and "6I" Speed Grade/Temperature Range part combinations may be dual marked as "7C/6I". Devices with the dual mark can be used as either -7C or -6I devices. Devices with a single mark are only guaranteed for the marked speed grade and temperature range.

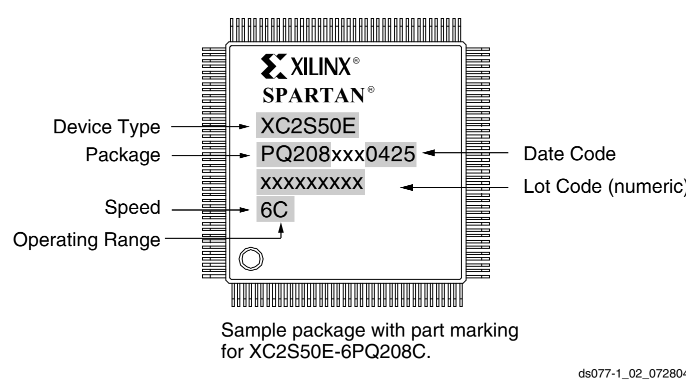


Figure 2: **Spartan-IIE QFP Marking Example**

Revision History

Date	Version	Description
06/27/2002	1.1	Updated -7 availability.
11/18/2002	2.0	Added XC2S400E and XC2S600E. Corrected XC2S150E max I/O count and XC2S50E differential I/O count and updated availability.
07/09/2003	2.1	Noted hot-swap capability. Updated Table 2 to show that all products are available. Clarified device part marking.
07/28/2004	2.2	Added information on Pb-free packaging options.
06/18/2008	2.3	Added dual mark information in Device Part Marking. Updated all modules for continuous page, figure, and table numbering. Updated links. Synchronized all modules to v2.3.
08/09/2013	3.0	This product is obsolete/discontinued per XCN12026.

Spartan-IIE FPGA Family: Functional Description

DS077-2 (v3.0) August 9, 2013

Product Specification

Architectural Description

Spartan-IIE FPGA Array

The Spartan®-IIE user-programmable gate array, shown in Figure 3, is composed of five major configurable elements:

- IOBs provide the interface between the package pins and the internal logic
- CLBs provide the functional elements for constructing most logic
- Dedicated block RAM memories of 4096 bits each
- Clock DLLs for clock-distribution delay compensation and clock domain control
- Versatile multi-level interconnect structure

As can be seen in Figure 3, the CLBs form the central logic structure with easy access to all support and routing structures. The IOBs are located around all the logic and memory elements for easy and quick routing of signals on and off the chip.

Values stored in static memory cells control all 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.

Each of these elements will be discussed in detail in the following sections.

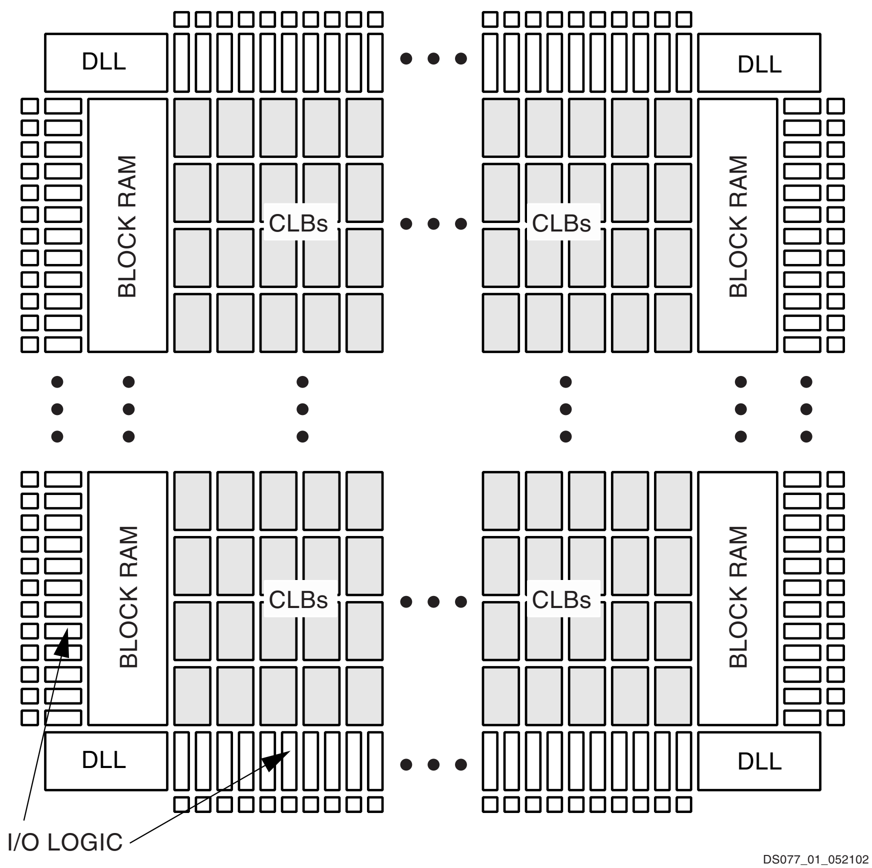


Figure 3: **Basic Spartan-IIE Family FPGA Block Diagram**

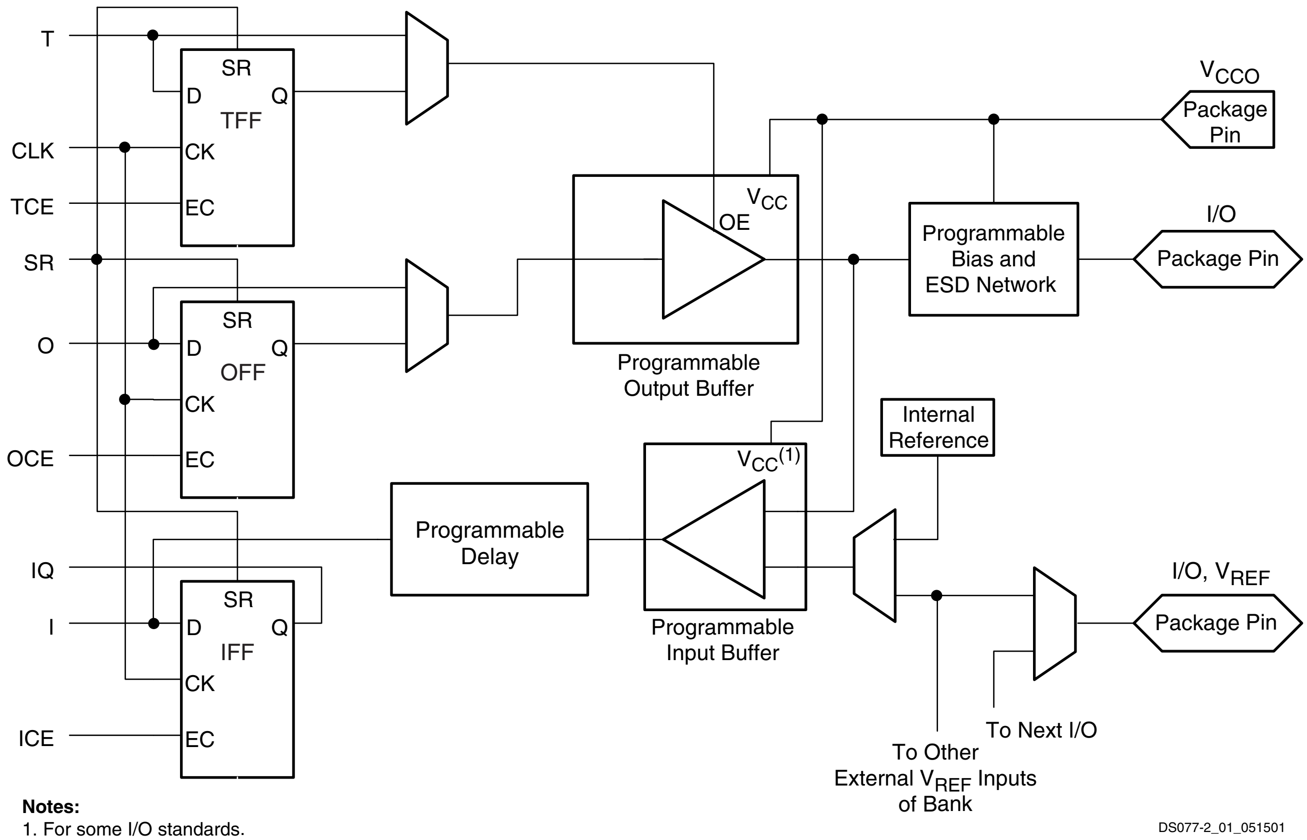


Figure 4: **Spartan-IIE Input/Output Block (IOB)**

Table 3: **Standards Supported by I/O (Typical Values)**

I/O Standard	Input Reference Voltage (V_{REF})	Input Voltage (V_{CCO})	Output Source Voltage (V_{CCO})	Board Termination Voltage (V_{TT})
LVTTL (2-24 mA)	N/A	3.3	3.3	N/A
LVCMOS2	N/A	2.5	2.5	N/A
LVCMOS18	N/A	1.8	1.8	N/A
PCI (3V, 33 MHz/66 MHz)	N/A	3.3	3.3	N/A
GTL	0.8	N/A	N/A	1.2
GTL+	1.0	N/A	N/A	1.5
HSTL Class I	0.75	N/A	1.5	0.75
HSTL Class III	0.9	N/A	1.5	1.5
HSTL Class IV	0.9	N/A	1.5	1.5
SSTL3 Class I and II	1.5	N/A	3.3	1.5
SSTL2 Class I and II	1.25	N/A	2.5	1.25
CTT	1.5	N/A	3.3	1.5
AGP	1.32	N/A	3.3	N/A
LVDS, Bus LVDS	N/A	N/A	2.5	N/A
LVPECL	N/A	N/A	3.3	N/A

Input/Output Block

The Spartan-IIE FPGA IOB, as seen in Figure 4, features inputs and outputs that support a wide variety of I/O signaling standards. These high-speed inputs and outputs are capable of supporting various state of the art memory and bus interfaces. The default standard is LVTTL. Table 3 lists several of the standards which are supported along with the required reference (V_{REF}), output (V_{CCO}) and board termination (V_{TT}) voltages needed to meet the standard. For more details on the I/O standards and termination application examples, see XAPP179, "*Using SelectIO Interfaces in Spartan-II and Spartan-IIE FPGAs*."

The three IOB registers 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 registers and independent Clock Enable (CE) signals for each register.

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

A feature not shown in the block diagram, but controlled by the software, is polarity control. The input and output buffers and all of the IOB control signals have independent polarity controls.

Optional pull-up and pull-down resistors and an optional weak-keeper circuit are attached to each user I/O 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 inputs may 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 will float. 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 pads are protected against damage from electrostatic discharge (ESD) and from over-voltage transients. After configuration, clamping diodes are connected to V_{CCO} for LVTTL, PCI, HSTL, SSTL, CTT, and AGP standards.

All Spartan-IIE FPGA IOBs support IEEE 1149.1-compatible boundary scan testing.

Input Path

A buffer in the IOB input path routes the input signal directly to internal logic and 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 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 signaling 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 used in close proximity to each other. See I/O Banking.

There are optional pull-up and pull-down resistors at each input for use after configuration.

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-flip that provides synchronous enable and disable.

Each output driver can be individually programmed for a wide range of low-voltage signaling 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. The default output driver is LVTTL with 12 mA drive strength and slow slew rate.

In most signaling 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.

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 helps eliminate bus chatter.

Because the weak-keeper circuit uses the IOB input buffer to monitor the input level, an appropriate V_{REF} voltage must be provided if the signaling 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 IOBs, 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 (see Figure 5). The pinout tables show the bank affiliation of each I/O (see Pinout Tables, page 53). Each bank has multiple V_{CCO} pins which must be connected to the same voltage. Voltage requirements are determined by the output standards in use.

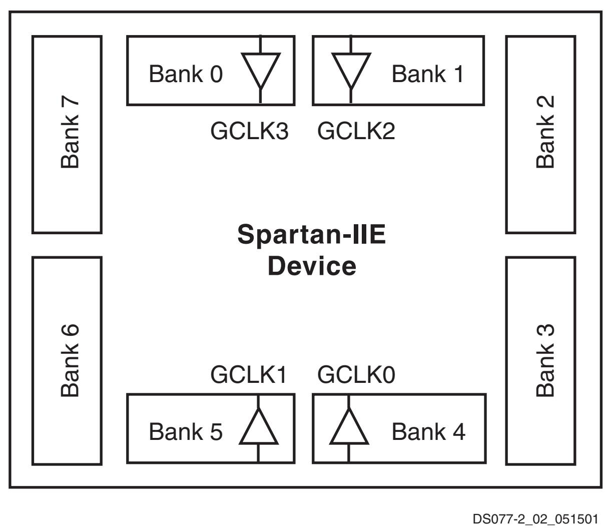


Figure 5: **Spartan-IIE I/O Banks**

In the TQ144 and PQ208 packages, the eight banks have V_{CCO} connected together. Thus, only one V_{CCO} level is allowed in these packages, although different V_{REF} values are allowed in each of the eight banks.

Within a bank, standards may be mixed only if they use the same V_{CCO}. Compatible standards are shown in Table 4. GTL and GTL+ appear under all voltages because their open-drain outputs do not depend on V_{CCO}. Note that V_{CCO}

is required for most output standards and for LVTTL, LVCMOS, and PCI inputs.

Table 4: **Compatible Standards**

V_{CCO}	Compatible Standards
3.3V	PCI, LVTTL, SSTL3 I, SSTL3 II, CTT, AGP, LVPECL, GTL, GTL+
2.5V	SSTL2 I, SSTL2 II, LVCMOS2, LVDS, Bus LVDS, GTL, GTL+
1.8V	LVCMOS18, GTL, GTL+
1.5V	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. About one in six of the I/O pins in the bank assume this role.

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.

In a bank, inputs requiring V_{REF} can be mixed with those that do not but only one V_{REF} voltage may be used within a bank. The V_{CCO} and V_{REF} pins for each bank appear in the device pinout tables.

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 superset of the V_{REF} pins used for smaller devices, it is possible to design a PCB that permits migration to a larger device. All V_{REF} pins for the largest device anticipated must be connected to the V_{REF} voltage, and not used for I/O.

Table 5: **I/O Banking**

Package	TQ144, PQ208	FT256, FG456, FG676
V_{CCO} Banks	Interconnected as 1	8 independent
V_{REF} Banks	8 independent	8 independent

See Xilinx® Application Note XAPP179 for more information on I/O resources.

Hot Swap, Hot Insertion, Hot Socketing Support

The I/O pins support hot swap — also called hot insertion and hot socketing — and are considered CompactPCI Friendly according to the PCI Bus v2.2 Specification. Consequently, an unpowered Spartan-IIE FPGA can be plugged directly into a powered system or backplane without affecting or damaging the system or the FPGA. The hot swap functionality is built into every XC2S150E, XC2S400E, and XC2S600E device. All other Spartan-IIE devices built after Product Change Notice PCN2002-05 also include hot swap functionality.

To support hot swap, Spartan-IIE devices include the following I/O features.

- Signals can be applied to Spartan-IIE FPGA I/O pins before powering the FPGA's V_{CCINT} or V_{CCO} supply inputs.
- Spartan-IIE FPGA I/O pins are high-impedance (i.e., three-stated) before and throughout the power-up and configuration processes when employing a configuration mode that does not enable the preconfiguration weak pull-up resistors (see Table 11, page 22).
- There is no current path from the I/O pin back to the V_{CCINT} or V_{CCO} voltage supplies.
- Spartan-IIE FPGAs are immune to latch-up during hot swap.

Once connected to the system, each pin adds a small amount of capacitance (C_{IN}). Likewise, each I/O consumes a small amount of DC current, equivalent to the input leakage specification (I_L). There also may be a small amount of temporary AC current (I_{HSPO}) when the pin input voltage exceeds V_{CCO} plus 0.4V, which lasts less than 10 ns.

A weak-keeper circuit within each user-I/O pin is enabled during the last frame of configuration data and has no noticeable effect on robust system signals driven by an active driver or a strong pull-up or pull-down resistor. Undriven or floating system signals may be affected. The specific effect depends on how the I/O pin is configured. User-I/O pins configured as outputs or enabled outputs have a weak pull-up resistor to V_{CCO} during the last configuration frame. User-I/O pins configured as inputs or bidirectional I/Os have weak pull-down resistors. The weak-keeper circuit turns off when the DONE pin goes High, provided that it is not used in the configured application.

Configurable Logic Block

The basic building block of the Spartan-IIE FPGA CLB is the logic cell (LC). An LC includes a 4-input function generator, carry logic, and storage element. The output from the function generator in each LC drives the CLB output or the D input of the flip-flop. Each Spartan-IIE FPGA CLB contains four LCs, organized in two similar slices; a single slice is shown in Figure 6.

In addition to the four basic LCs, the Spartan-IIE FPGA CLB contains logic that combines function generators to provide functions of five or six inputs.

Look-Up Tables

Spartan-IIE FPGA 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 combined 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 Spartan-IIE FPGA 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

Storage elements in the Spartan-IIE FPGA 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 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 may be configured to operate asynchronously.

All control signals are independently invertible, and are shared by the two flip-flops within the slice.

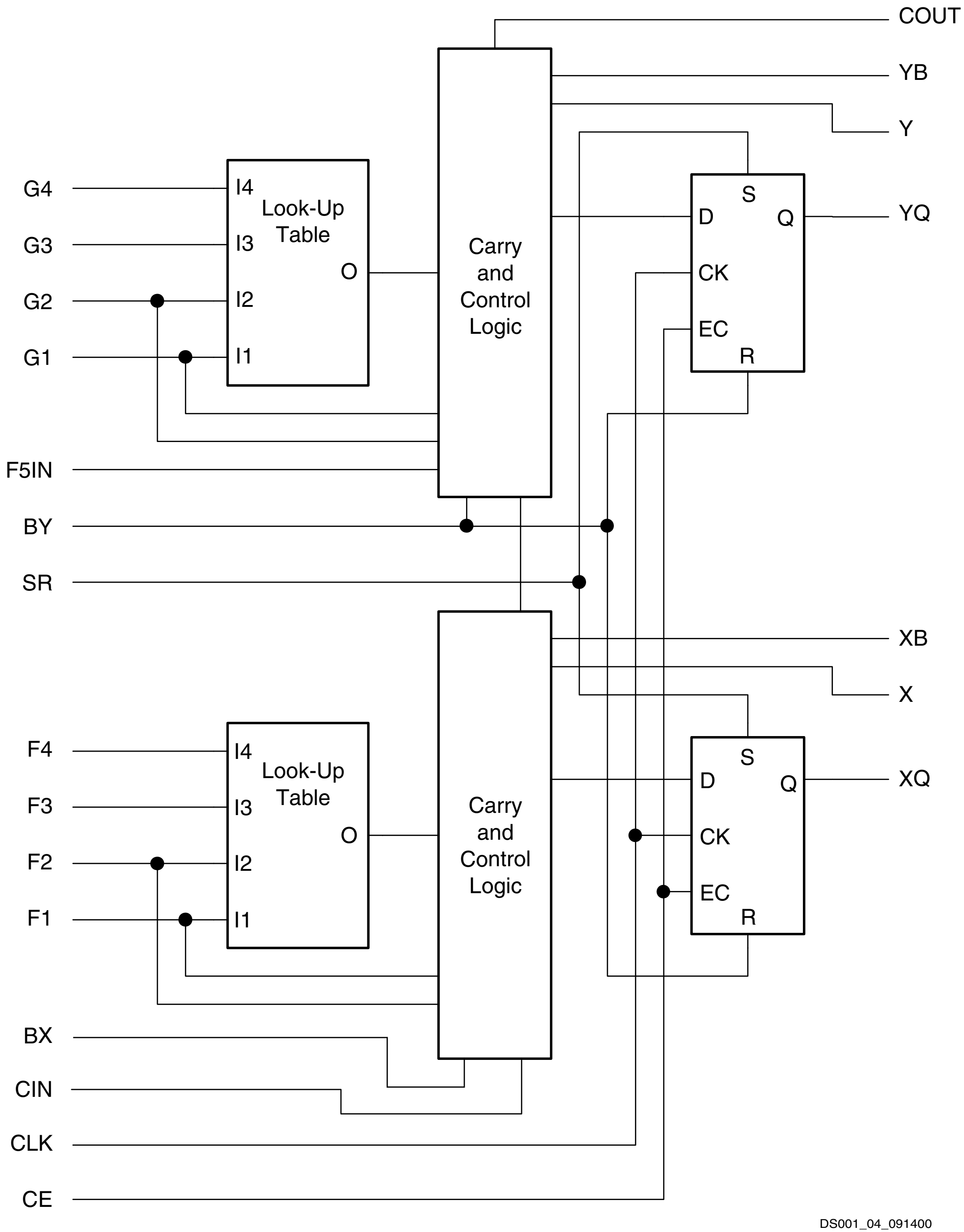


Figure 6: **Spartan-IIE CLB Slice** (two identical slices in each CLB)

Additional Logic

The F5 multiplexer in each slice combines the function generator outputs (Figure 7). 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 two F5-multiplexer outputs. This permits the implementation of any 6-input function, an 8:1 multiplexer, or selected functions of up to 19 inputs.

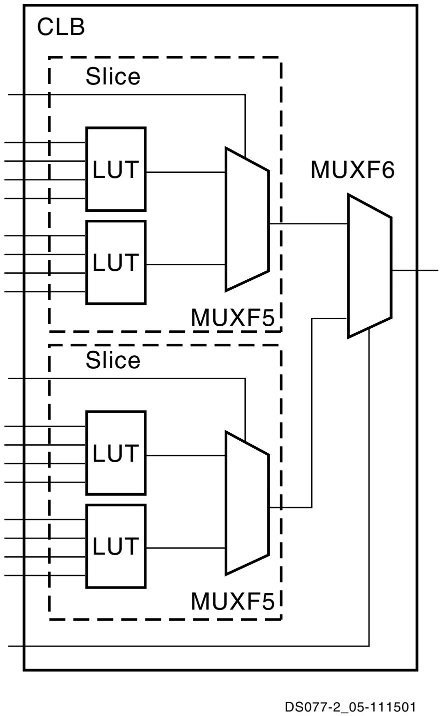


Figure 7: **F5 and F6 Multiplexers**

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

Arithmetic Logic

Dedicated carry logic provides capability for high-speed arithmetic functions. The Spartan-IIE FPGA 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 1-bit full adder to be implemented within an LC. In addition, a dedicated AND gate improves the efficiency of multiplier implementations.

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

BUFTs

Each Spartan-IIE FPGA CLB contains two 3-state drivers (BUFTs) that can drive on-chip busses. The IOBs on the left and right sides can also drive the on-chip busses. See Dedicated Routing, page 17. Each Spartan-IIE FPGA BUFT has an independent 3-state control pin and an independent input pin. The 3-state control pin is an active-Low enable (T). When all BUFTs on a net are disabled, the net is High. There is no need to instantiate a pull-up unless desired for simulation purposes. Simultaneously driving BUFTs onto the same net will not cause contention. If driven both High and Low, the net will be Low.

Block RAM

Spartan-IIE FPGAs incorporate several large block RAM memories. These complement the distributed RAM Look-Up Tables (LUTs) that provide shallow memory structures implemented in CLBs.

Block RAM memory blocks are organized in columns. Most Spartan-IIE devices contain two such columns, one along each vertical edge. The XC2S400E has four block RAM columns and the XC2S600E has six block RAM columns. These columns extend the full height of the chip. Each memory block is four CLBs high, and consequently, a Spartan-IIE device 16 CLBs high will contain four memory blocks per column, and a total of eight blocks.

Table 6: **Spartan-IIE Block RAM Amounts**

Spartan-IIE Device	# of Blocks	Total Block RAM Bits
XC2S50E	8	32K
XC2S100E	10	40K
XC2S150E	12	48K
XC2S200E	14	56K
XC2S300E	16	64K
XC2S400E	40	160K
XC2S600E	72	288K

Each block RAM cell, as illustrated in Figure 8, is a fully synchronous dual-ported 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.

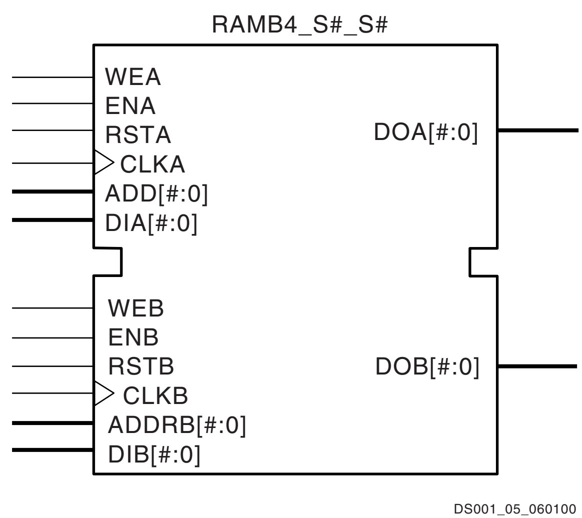


Figure 8: **Dual-Port Block RAM**

Table 7 shows the depth and width aspect ratios for the block RAM.

Table 7: **Block RAM 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>

The Spartan-IIE FPGA block RAM also includes dedicated routing to provide an efficient interface with both CLBs and other block RAMs. See Xilinx Application Note XAPP173 for more information on block RAM.

Programmable Routing

It is the longest delay path that limits the speed of any design. Consequently, the Spartan-IIE FPGA routing architecture and its place-and-route software were defined jointly to minimize long-path delays and yield 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.

The software automatically uses the best available routing based on user timing requirements. The details are provided here for reference.

Local Routing

The local routing resources, as shown in Figure 9, provide the following three types of connections:

- Interconnections among the LUTs, flip-flops, and General Routing Matrix (GRM), described below.
- 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

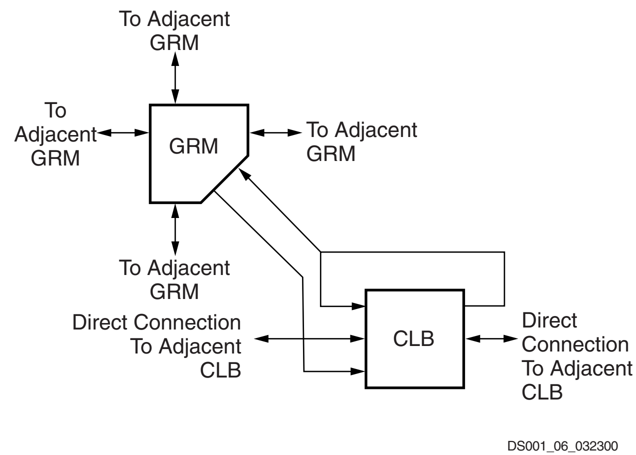


Figure 9: **Spartan-IIE Local Routing**

General Purpose Routing

Most Spartan-IIE FPGA 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 rows and columns of CLBs. 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.
- 96 buffered Hex lines route GRM signals to other GRMs six blocks away in each one of the four directions. Organized in a staggered pattern, Hex lines may be 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 unidirectional.
- 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

Spartan-IIE 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™ routing, 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 classes of signal require dedicated routing resources to maximize performance. In the Spartan-IIE FPGA architecture, dedicated routing resources are provided for two classes of signal.

- 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 10.
- Two dedicated nets per CLB propagate carry signals vertically to the adjacent CLB.

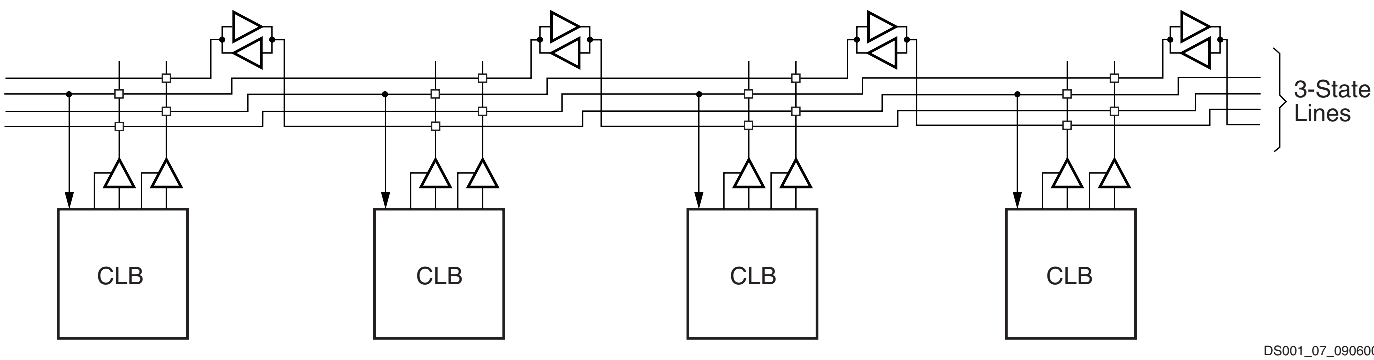


Figure 10: **BUFT Connections to Dedicated Horizontal Bus Lines**

Global Routing

Global Routing resources distribute clocks and other signals with very high fanout throughout the device. Spartan-IIE devices include two tiers of global routing resources referred to as primary and secondary global routing resources.

- The primary 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 primary global nets may only be driven by global buffers. There are four global buffers, one for each global net.
- The secondary global routing resources consist of 24 backbone lines, 12 across the top of the chip and 12 across the bottom. From these lines, up to 12 unique signals per column can be distributed via the 12 longlines in the column. These secondary resources are more flexible than the primary resources since they are not restricted to routing only to clock pins.

Clock Distribution

The Spartan-IIE family provides high-speed, low-skew clock distribution through the primary global routing resources described above. A typical clock distribution net is shown in Figure 11.

Four global buffers are provided, two at the top center of the device and two at the bottom center. These drive the four primary 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.

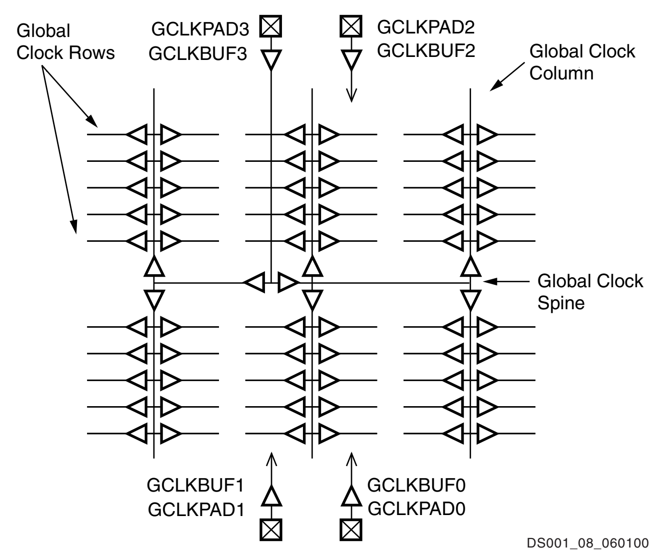


Figure 11: **Global Clock Distribution Network**

Delay-Locked Loop (DLL)

Associated with each global clock input buffer is a fully digital Delay-Locked Loop (DLL) that can eliminate skew between the clock input pad and 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 (Figure 12). Additional delay is introduced such that clock edges reach internal flip-flops exactly one clock period after they arrive at the input. This closed-loop system effectively eliminates clock-distribution delay by ensuring that clock edges arrive at internal flip-flops in synchronism with clock edges arriving at the input.

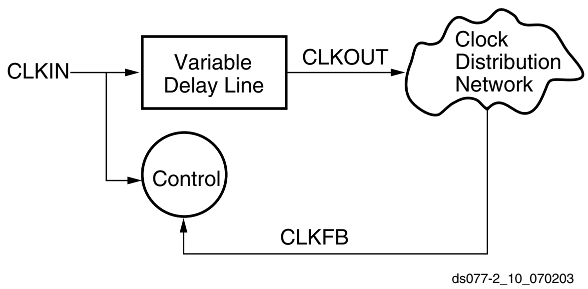


Figure 12: **Delay-Locked Loop Block Diagram**

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, can double the clock, or divide the clock by 1.5, 2, 2.5, 3, 4, 5, 8, or 16. The phase-shifted output have optional duty-cycle correction (Figure 13).

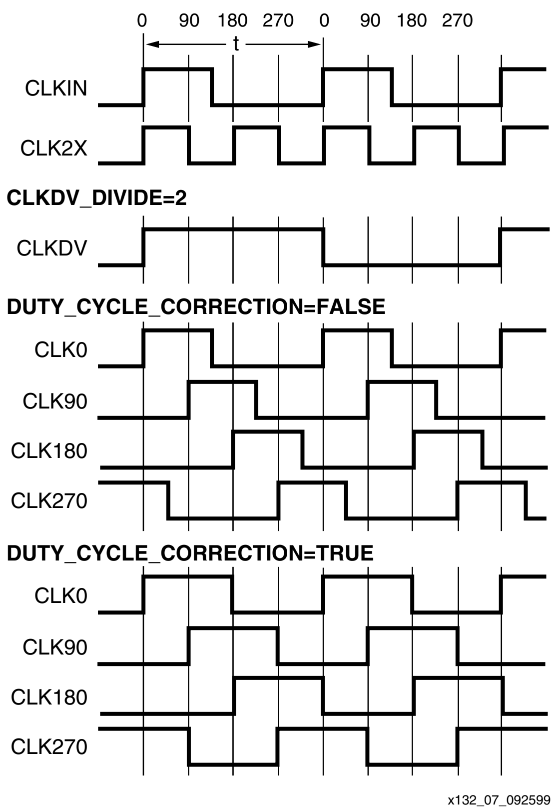


Figure 13: **DLL Output Characteristics**

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 deskew a board level clock among multiple Spartan-IIE 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. If the DLL uses external feedback, apply a reset after startup to ensure consistent locking to the external signal. See Xilinx Application Note XAPP174 for more information on DLLs.

Boundary Scan

Spartan-IIE 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, and HIGHZ instructions. The TAP also supports two USERCODE instructions, internal scan chains, and configuration/readback of the device.

The TAP uses dedicated package pins that always operate using LVTTL. For TDO to operate using LVTTL, the V_{CCO} for Bank 2 must be 3.3V. Otherwise, TDO switches rail-to-rail between ground and V_{CCO}. The boundary-scan input pins (TDI, TMS, TCK) do not have a V_{CCO} requirement and operate with either 2.5V or 3.3V input signaling levels. TDI, TMS, and TCK hava a default internal weak pull-up resistor, and TDO has no default resistor. Bitstream options allow setting any of the four TAP pins to have an internal pull-up, pull-down, or neither.

Boundary-scan operation is independent of individual IOB configurations, and unaffected by package type. All IOBs, including unbonded 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.

Table 8 lists the boundary-scan instructions supported in Spartan-IIE FPGAs. Internal signals can be captured during EXTEST by connecting them to unbonded or unused IOBs. They may also be connected to the unused outputs of IOBs defined as unidirectional input pins.

Table 8: **Boundary-Scan Instructions**

Boundary-Scan Command	Binary Code[4:0]	Description
EXTEST	00000	Enables boundary-scan EXTEST operation
SAMPLE/ PRELOAD	00001	Enables 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 Readback
CFG_IN	00101	Access the configuration bus for Configuration

Table 8: **Boundary-Scan Instructions *(Continued)***

Boundary-Scan Command	Binary Code[4:0]	Description
INTEST	00111	Enables boundary-scan INTEST operation
USERCODE	01000	Enables shifting out USER code
IDCODE	01001	Enables shifting out of ID Code
HIGHZ	01010	Disables output pins while enabling the Bypass Register
JSTART	01100	Clock the start-up sequence when StartupClk is TCK
BYPASS	11111	Enables BYPASS
RESERVED	All other codes	Xilinx reserved instructions

The public boundary-scan instructions are available prior to configuration, except for USER1 and USER2. After configuration, the public instructions remain available together with any USERCODE instructions installed during the configuration. While the SAMPLE/PRELOAD and BYPASS instructions are available during configuration, it is recommended that boundary-scan operations not be performed during this transitional period.

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.

To facilitate internal scan chains, the User Register provides three outputs (Reset, Update, and Shift) that represent the corresponding states in the boundary-scan internal state machine.

Figure 14 is a diagram of the Spartan-IIE family 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.

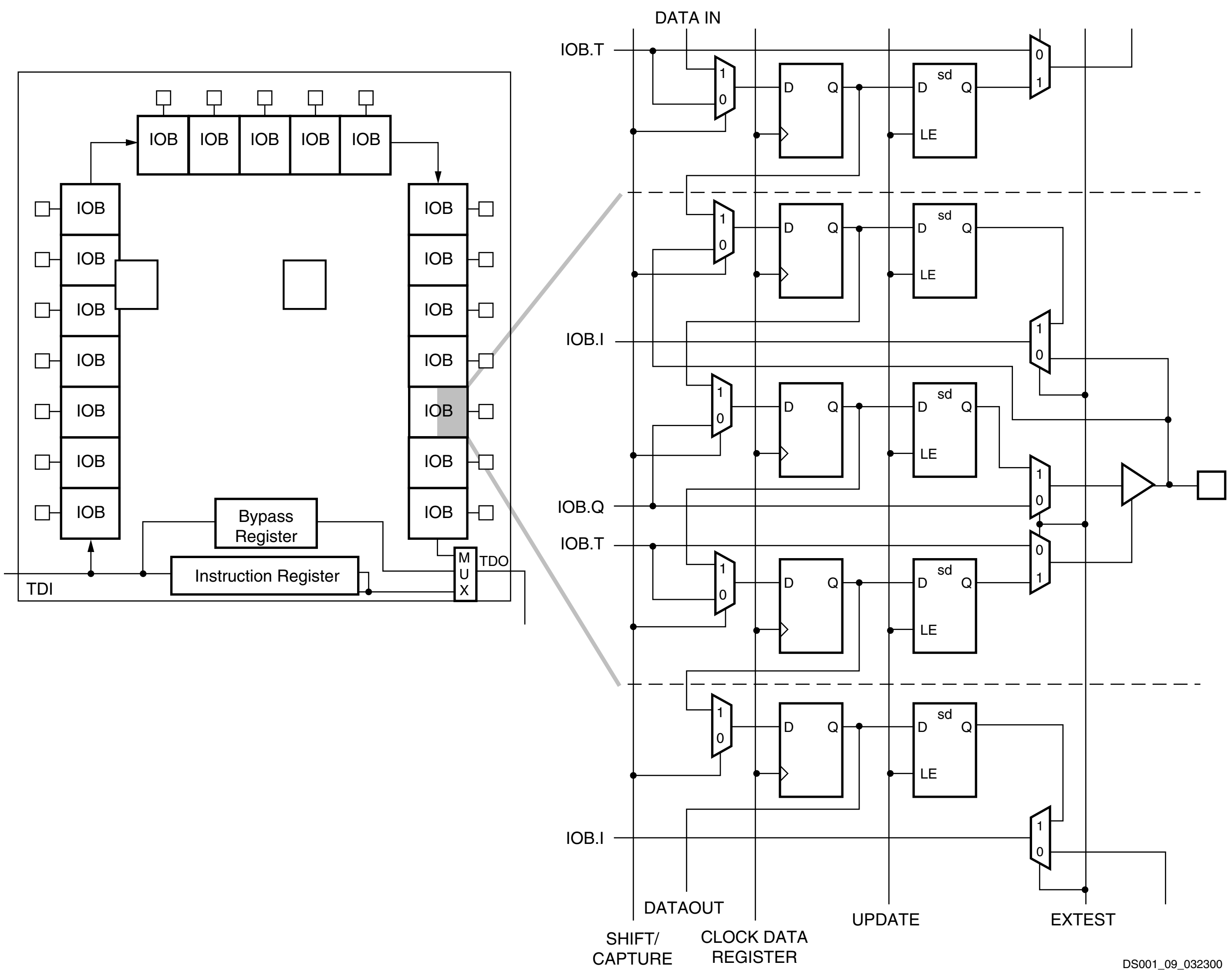


Figure 14: **Spartan-IIE Family Boundary Scan Logic**

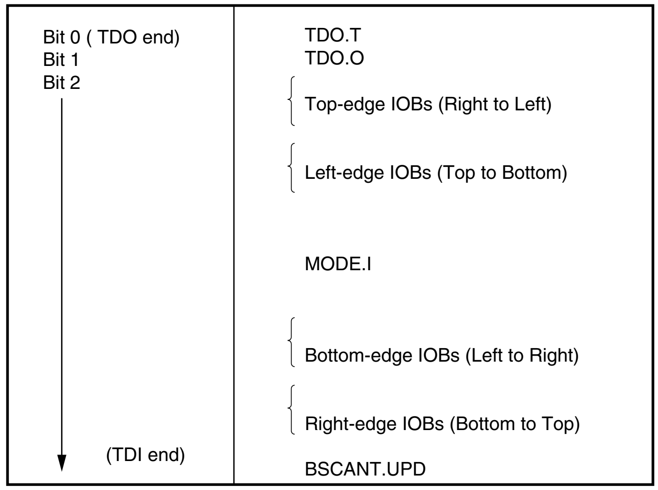
Bit Sequence

The bit sequence 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 the FPGA Editor), starting in the upper right chip corner, the boundary scan data-register bits are ordered as shown in Figure 15.

BSDL (Boundary Scan Description Language) files for Spartan-IIE family devices are available on the Xilinx web site.

Spartan-IIE FPGA boundary scan IDCODE values are shown in Table 9.



DS001_10_032300

Figure 15: **Boundary Scan Bit Sequence**

Table 9: **Spartan-IIE IDCODE Values**

	IDCODE				
Device	**Version**	**Family**	**Array Size**	**Manufacturer**	**Required**
XC2S50E	XXXX	0000 101	0 0001 0000	0000 1001 001	1
XC2S100E	XXXX	0000 101	0 0001 0100	0000 1001 001	1
XC2S150E	XXXX	0000 101	0 0001 1000	0000 1001 001	1
XC2S200E	XXXX	0000 101	0 0001 1100	0000 1001 001	1
XC2S300E	XXXX	0000 101	0 0010 0000	0000 1001 001	1
XC2S400E	XXXX	0000 101	0 0010 1000	0000 1001 001	1
XC2S600E	XXXX	0000 101	0 0011 0000	0000 1001 001	1

Development System

Spartan-IIE FPGAs are supported by the Xilinx ISE® CAE tools. The basic methodology for Spartan-IIE FPGA design consists of three interrelated steps: design entry, implementation, and verification. Industry-standard tools are used for design entry and simulation, while Xilinx provides proprietary architecture-specific tools for implementation.

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

Several advanced software features facilitate Spartan-IIE FPGA design. CORE Generator™ tool functions, for example, include macros with relative location constraints to guide their placement. They help ensure optimal implementation of common functions.

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

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

Spartan-IIE 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 design environment supports hierarchical design entry, with high-level designs that comprise major functional blocks, while lower-level designs 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 automatically provide the implementation flow described in this section. The partitioner takes the EDIF netlist 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 interconnections and the desired performance. Finally, the router interconnects the blocks.

The 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 floorplanning.

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

Timing requirements are entered 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.

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 netlist for use by the simulator. Alternatively, the user can verify timing-critical portions of the design using the static timing analyzer.

For in-circuit debugging, Xilinx offers a download cable, which connects the FPGA in the target system to a PC or workstation. After downloading the design into the FPGA, the designer can read back 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.

Configuration

Configuration is the process by which the bitstream of a design, as generated by the Xilinx development software, is loaded into the internal configuration memory of the FPGA. Spartan-IIE devices support both serial configuration, using the master/slave serial and JTAG modes, as well as byte-wide configuration employing the Slave Parallel mode.

Configuration File

Spartan-IIE devices are configured by sequentially loading frames of data that have been concatenated into a configuration file. Table 10 shows how much nonvolatile storage space is needed for Spartan-IIE devices.

It is important to note that, while a PROM is commonly used to store configuration data before loading them into the FPGA, it is by no means required. Any of a number of different kinds of under populated nonvolatile storage already available either on or off the board (for example, hard drives, FLASH cards, and so on) can be used.

Table 10: **Spartan-IIE Configuration File Size**

Device	Configuration File Size (Bits)
XC2S50E	630,048
XC2S100E	863,840
XC2S150E	1,134,496
XC2S200E	1,442,016
XC2S300E	1,875,648
XC2S400E	2,693,440
XC2S600E	3,961,632

Modes

Spartan-IIE devices support the following four configuration modes:

- Slave Serial mode
- Master Serial mode
- Slave Parallel mode
- Boundary-scan mode

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 the end of configuration. The selection codes are listed in Table 11.

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.

Table 11: **Configuration Modes**

Configuration Mode	Preconfiguration Pull-ups	M0	M1	M2	CCLK Direction	Data Width	Serial D_{OUT}
Master Serial mode	No	0	0	0	Out	1	Yes
	Yes	0	0	1			
Slave Parallel mode (SelectMAP)	Yes	0	1	0	In	8	No
	No	0	1	1			
Boundary-Scan mode	Yes	1	0	0	N/A	1	No
	No	1	0	1			
Slave Serial mode	Yes	1	1	0	In	1	Yes
	No	1	1	1			

Notes:
1. During power-on and throughout configuration, the I/O drivers will be in a high-impedance state. After configuration, all unused I/Os (those not assigned signals) will remain in a high-impedance state. Pins used as outputs may pulse High at the end of configuration (see Answer 10504).
2. If the Mode pins are set for preconfiguration pull-ups, those resistors go into effect once the rising edge of INIT samples the Mode pins. They will stay in effect until GTS is released during startup, after which the UnusedPin bitstream generator option will determine whether the unused I/Os have a pull-up, pull-down, or no resistor.

Signals

There are two kinds of pins that are used to configure Spartan-IIE devices: Dedicated pins perform only specific configuration-related functions; the other pins can serve as general purpose I/Os once user operation has begun.

The dedicated pins comprise the mode pins (M2, M1, M0), the configuration clock pin (CCLK), the $\overline{\text{PROGRAM}}$ pin, the DONE pin and the boundary-scan pins (TDI, TDO, TMS, TCK). Depending on the selected configuration mode, CCLK may be an output generated by the FPGA, or may be generated externally, and provided to the FPGA as an input.

Note that some configuration pins can act as outputs. For correct operation, these pins require a V_{CCO} of 3.3V to drive an LVTTL signal or 2.5V to drive an LVCMOS signal. All the relevant pins fall in banks 2 or 3. The $\overline{\text{CS}}$ and $\overline{\text{WRITE}}$ pins for Slave Parallel mode are located in bank 1.

For a more detailed description than that given below, see ***Module 1*** and XAPP176*, Configuration and Readback of the Spartan-II and Spartan-IIE FPGA Families.*

The Process

The sequence of steps necessary to configure Spartan-IIE devices are shown in Figure 16. The overall flow can be divided into three different phases.

- Initiating configuration
- Configuration memory clear
- Loading data frames
- Start-up

The memory clearing and start-up phases are the same for all configuration modes; however, the steps for the loading of data frames are different. Thus, the details for data frame loading are described separately in the sections devoted to each mode.

Initiating Configuration

There are two different ways to initiate the configuration process: applying power to the device or asserting the $\overline{\text{PROGRAM}}$ input.

Configuration on power-up occurs automatically unless it is delayed by the user, as described in a separate section below. The waveform for configuration on power-up is shown in Configuration Switching Characteristics, page 48. Before configuration can begin, V_{CCO} Bank 2 must be greater than 1.0V. Furthermore, all V_{CCINT} power pins must be connected to a 1.8V supply. For more information on delaying configuration, see Clearing Configuration Memory, page 23.

Once in user operation, the device can be re-configured simply by pulling the $\overline{\text{PROGRAM}}$ pin Low. The device acknowledges the beginning of the configuration process by driving DONE Low, then enters the memory-clearing phase.

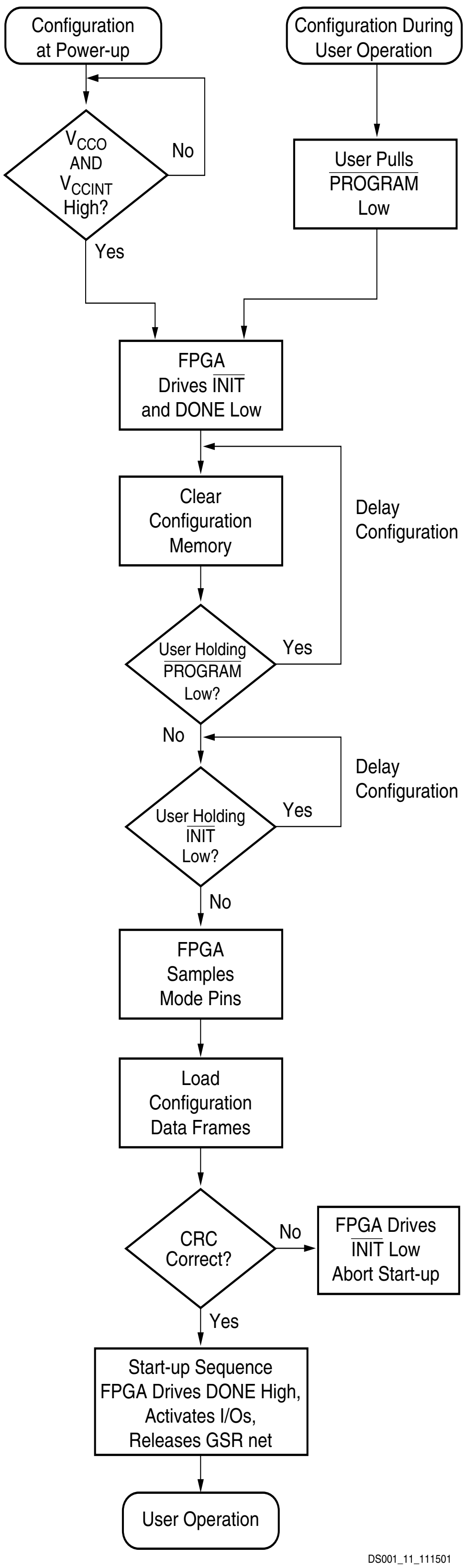


Figure 16: **Configuration Flow Diagram**

Clearing Configuration Memory

The device indicates that clearing the configuration memory is in progress by driving INIT Low.

Delaying Configuration

At this time, the user can delay configuration by holding either PROGRAM or INIT Low, which causes the device to remain in the memory clearing phase. Note that the bidirectional INIT line is driving a Low logic level during memory clearing. Thus, to avoid contention, use an open-drain driver to keep INIT Low.

With no delay in force, the device indicates that the memory is completely clear by driving INIT High. The FPGA samples its mode pins on this Low-to-High transition.

Loading Configuration Data

Once INIT is High, the user can begin loading configuration data frames into the device. The details of loading the configuration data are discussed in the sections treating the configuration modes individually. The sequence of operations necessary to load configuration data using the serial modes is shown in Figure 18. Loading data using the Slave Parallel mode is shown in Figure 21, page 28.

CRC Error Checking

After the loading of configuration data, a CRC value embedded in the configuration file is checked against a CRC value calculated within the FPGA. If the CRC values do not match, the FPGA drives INIT Low to indicate that an error has occurred and configuration is aborted. Note that attempting to load an incorrect bitstream causes configuration to fail and can damage the device.

To reconfigure the device, the PROGRAM pin should be asserted to reset the configuration logic. Recycling power also resets the FPGA for configuration. See Clearing Configuration Memory.

Start-up

The start-up sequence oversees the transition of the FPGA from the configuration state to full user operation. A match of CRC values, indicating a successful loading of the configuration data, initiates the sequence.

During start-up, the device performs four operations:

1. The assertion of DONE. The failure of DONE to go High may indicate the unsuccessful loading of configuration data.
2. The release of the Global Three State (GTS). This activates all the I/Os to which signals are assigned. The remaining I/Os stay in a high-impedance state with internal weak pull-up resistors present.
3. The release of the Global Set Reset (GSR). This allows all flip-flops to change state.
4. The assertion of Global Write Enable (GWE). This allows all RAMs and flip-flops to change state.

By default, these operations are synchronized to CCLK. The entire start-up sequence lasts eight cycles, called C0-C7, after which the loaded design is fully functional. The four operations can be selected to switch on any CCLK cycle C1-C6 through settings in the Xilinx Development Software. The default timing for start-up is shown in the top half of Figure 17; heavy lines show default settings.

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 bottom half of Figure 17 shows another commonly used version of the start-up timing known as Sync-to-DONE. This version makes the GTS, GSR, and GWE events conditional upon the DONE pin going High. This timing is important for a daisy chain of multiple FPGAs in serial mode, since it ensures that all FPGAs go through start-up together, after all their DONE pins have gone High.

Sync-to-DONE timing is selected by setting the GTS, GSR, and GWE cycles to a value of DONE in the configuration options. This causes these signals to transition one clock cycle after DONE externally transitions High.

The sequence can also be paused at any stage until lock has been achieved on any or all DLLs.

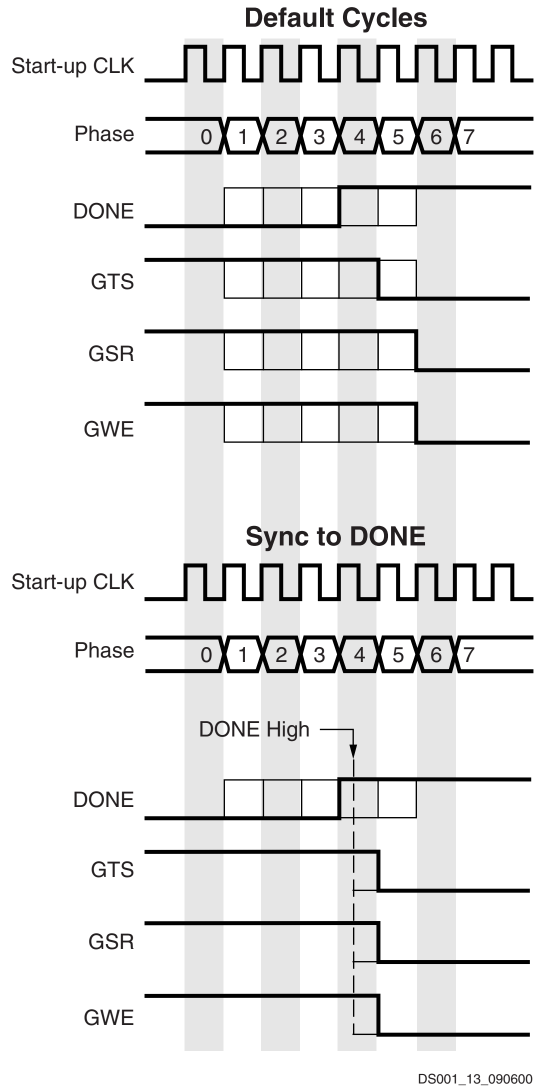


Figure 17: **Start-Up Waveforms**

Serial Modes

There are two serial configuration modes. In Master Serial mode, the FPGA controls the configuration process by driving CCLK as an output. In Slave Serial mode, the FPGA passively receives CCLK as an input from an external agent (e.g., a microprocessor, CPLD, or second FPGA in master mode) that is controlling the configuration process. In both modes, the FPGA is configured by loading one bit per CCLK cycle. The MSB of each configuration data byte is always written to the DIN pin first.

See Figure 18 for the sequence for loading data into the Spartan-IIE FPGA serially. This is an expansion of the "Load Configuration Data Frames" block in Figure 16, page 23. Note that $\overline{CS}$ and $\overline{WRITE}$ are not normally used during serial configuration. To ensure successful loading of the FPGA, do not toggle $\overline{WRITE}$ with $\overline{CS}$ Low during serial configuration.

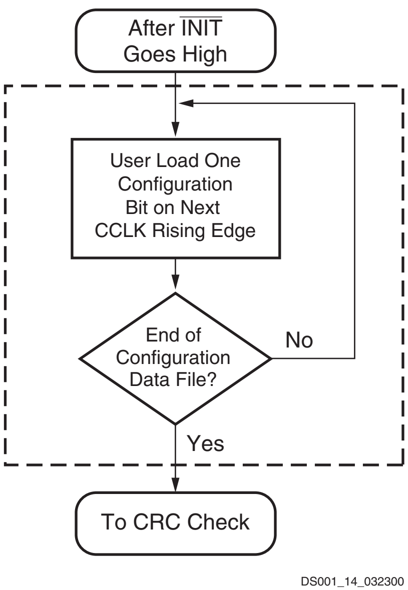


Figure 18: **Loading Serial Mode Configuration Data**

Slave Serial Mode

In Slave Serial mode, the FPGA's CCLK pin is driven by an external source, allowing the FPGA to be configured from other logic devices such as microprocessors or in a daisy-chain configuration. Figure 19 shows connections for a Master Serial FPGA configuring a Slave Serial FPGA from a PROM. A Spartan-IIE device in slave serial mode should be connected as shown for the third device from the left. Slave Serial mode is selected by a <11x> on the mode pins (M0, M1, M2). The weak pull-ups on the mode pins make slave serial the default mode if the pins are left unconnected.

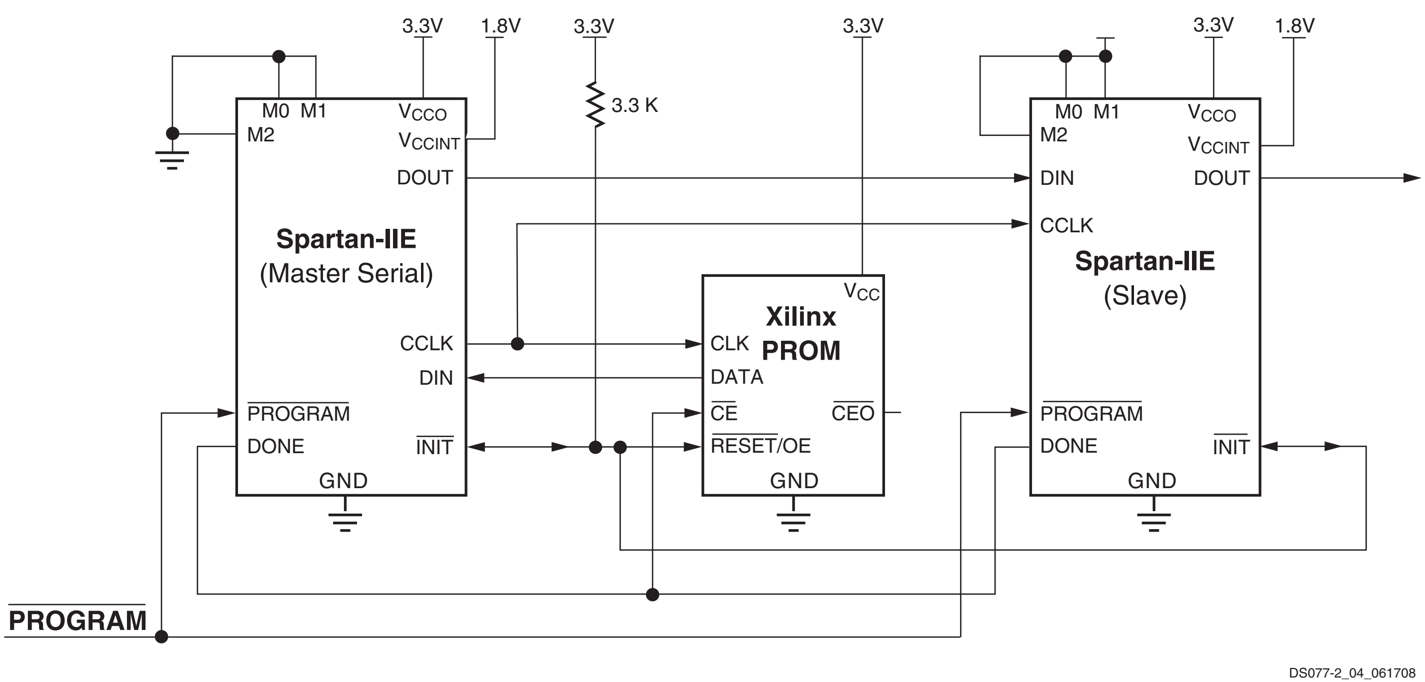
The serial bitstream must be setup at the DIN input pin a short time before each rising edge of an externally generated CCLK.

Timing for Slave Serial mode is shown in Figure 24, page 49.

Daisy Chain

Multiple FPGAs in Slave Serial mode can be daisy-chained for configuration from a single source. After an FPGA is configured, data for the next device is sent to the DOUT pin. Data on the DOUT pin changes on the rising edge of CCLK. Note that DOUT changes on the falling edge of CCLK for some Xilinx families but mixed daisy chains are allowed. Configuration must be delayed until INIT pins of all daisy-chained FPGAs are High. For more information, see Start-up, page 23.

The maximum amount of data that can be sent to the DOUT pin for a serial daisy chain is 2^{20}-1 (1,048,575) 32-bit words, or 33,554,400 bits, which is approximately 8 XC2S600E bitstreams. The configuration bitstream of downstream devices is limited to this size.



Notes:
1. If the DriveDone configuration option is not active for any of the FPGAs, pull up DONE with a 330Ω resistor.

Figure 19: **Master/Slave Serial Configuration Circuit Diagram**