Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details	
Product Status	Active
Number of LABs/CLBs	240
Number of Logic Elements/Cells	2160
Total RAM Bits	73728
Number of I/O	108
Number of Gates	100000
Voltage - Supply	1.14V ~ 1.26V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	144-LQFP
Supplier Device Package	144-TQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xc3s100e-5tqg144c

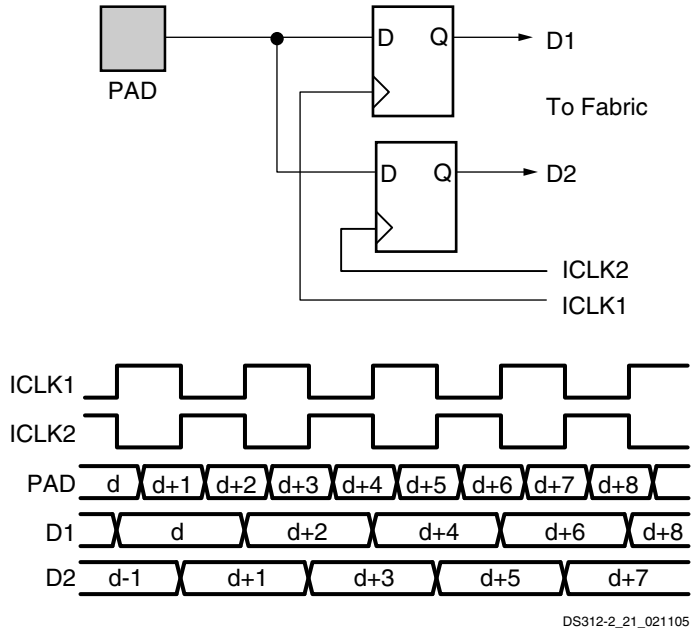

Figure 8: **Input DDR (without Cascade Feature)**

In the Spartan-3E device, the signal D2 can be cascaded into the storage element of the adjacent slave IOB. There it is re-registered to ICLK1, and only then fed to the FPGA fabric where it is now already in the same time domain as D1. Here, the FPGA fabric uses only the clock ICLK1 to process the received data. See Figure 9 for a graphical illustration of this function.

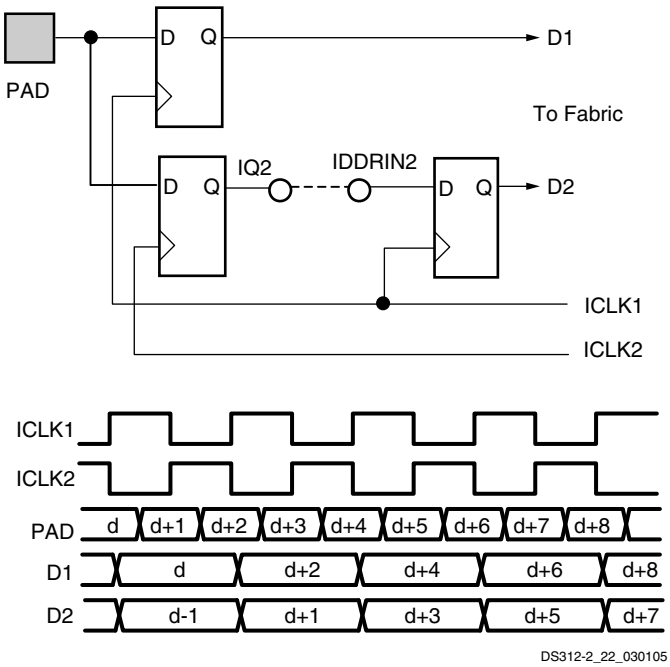

Figure 9: **Input DDR Using Spartan-3E Cascade Feature**

ODDR2

As a DDR output pair, the master IOB registers data coming from the FPGA fabric on the rising edge of OCLK1 (= D1) and the rising edge of OCLK2 (= D2), which is typically the same as the falling edge of OCLK1. These two bits of data are multiplexed by the DDR mux and forwarded to the output pin. The D2 data signal must be re-synchronized from the OCLK1 clock domain to the OCLK2 domain using FPGA slice flip-flops. Placement is critical at high frequencies, because the time available is only one half a clock cycle. See Figure 10 for a graphical illustration of this function.

The C0 or C1 alignment feature of the ODDR2 flip-flop, originally introduced in the Spartan-3E FPGA family, is not recommended or supported in the ISE development software. The ODDR2 flip-flop without the alignment feature remains fully supported. Without the alignment feature, the ODDR2 feature behaves equivalent to the ODDR flip-flop on previous Xilinx FPGA families.

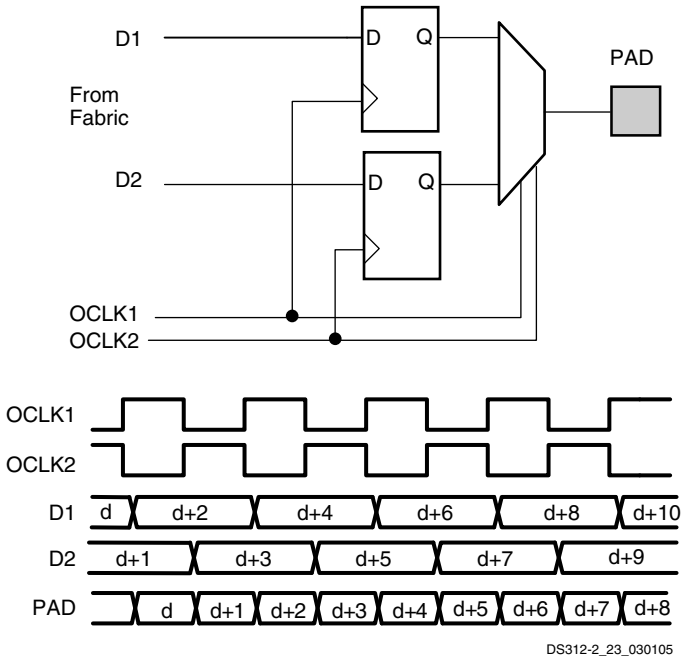

Figure 10: **Output DDR**

SelectIO Signal Standards

The Spartan-3E I/Os feature inputs and outputs that support a wide range of I/O signaling standards (Table 6 and Table 7). The majority of the I/Os also can be used to form differential pairs to support any of the differential signaling standards (Table 7).

To define the I/O signaling standard in a design, set the IOSTANDARD attribute to the appropriate setting. Xilinx provides a variety of different methods for applying the IOSTANDARD for maximum flexibility. For a full description of different methods of applying attributes to control IOSTANDARD, refer to the Xilinx Software Manuals and Help.

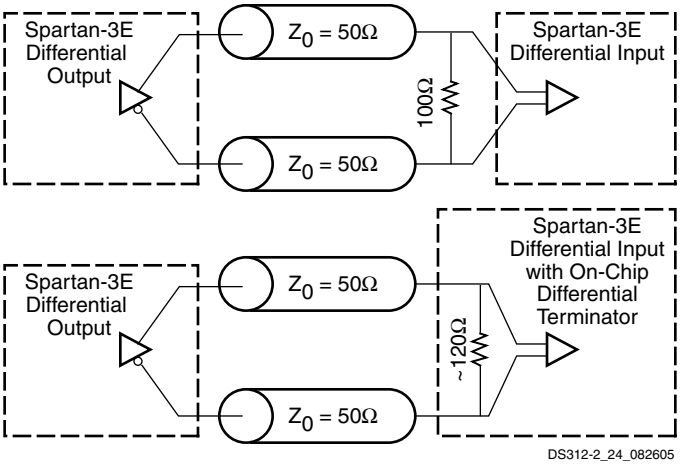

Figure 11: **Differential Inputs and Outputs**

Pull-Up and Pull-Down Resistors

Pull-up and pull-down resistors inside each IOB optionally force a floating I/O or Input-only pin to a determined state. Pull-up and pull-down resistors are commonly applied to unused I/Os, inputs, and three-state outputs, but can be used on any I/O or Input-only pin. The pull-up resistor connects an IOB to V_{CCO} through a resistor. The resistance value depends on the V_{CCO} voltage (see Module 3, DC and Switching Characteristics for the specifications). The pull-down resistor similarly connects an IOB to ground with a resistor. The PULLUP and PULLDOWN attributes and library primitives turn on these optional resistors.

By default, PULLDOWN resistors terminate all unused I/O and Input-only pins. Unused I/O and Input-only pins can alternatively be set to PULLUP or FLOAT. To change the unused I/O Pad setting, set the Bitstream Generator (BitGen) option ***UnusedPin*** to PULLUP, PULLDOWN, or FLOAT. The ***UnusedPin*** option is accessed through the Properties for Generate Programming File in ISE. See Bitstream Generator (BitGen) Options.

During configuration a Low logic level on the HSWAP pin activates pull-up resistors on all I/O and Input-only pins not actively used in the selected configuration mode.

Keeper Circuit

Each I/O has an optional keeper circuit (see Figure 12) that keeps bus lines from floating when not being actively driven. The KEEPER circuit retains the last logic level on a line after all drivers have been turned off. Apply the KEEPER attribute or use the KEEPER library primitive to use the KEEPER circuitry. Pull-up and pull-down resistors override the KEEPER settings.

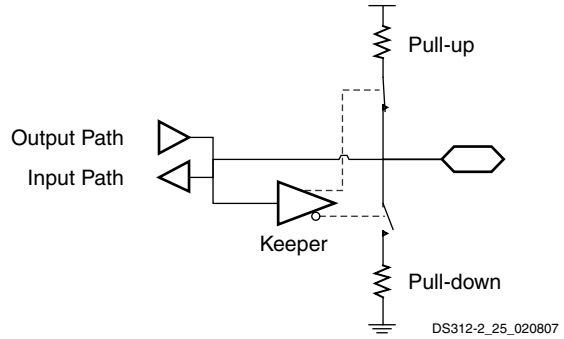

Figure 12: **Keeper Circuit**

Slew Rate Control and Drive Strength

Each IOB has a slew-rate control that sets the output switching edge-rate for LVCMOS and LVTTL outputs. The SLEW attribute controls the slew rate and can either be set to SLOW (default) or FAST.

Each LVCMOS and LVTTL output additionally supports up to six different drive current strengths as shown in Table 8. To adjust the drive strength for each output, the DRIVE attribute is set to the desired drive strength: 2, 4, 6, 8, 12, and 16. Unless otherwise specified in the FPGA application, the software default IOSTANDARD is LVCMOS25, SLOW slew rate, and 12 mA output drive.

Table 8: **Programmable Output Drive Current**

IOSTANDARD	Output Drive Current (mA)					
	2	4	6	8	12	16
LVTTL	✔	✔	✔	✔	✔	✔
LVCMOS33	✔	✔	✔	✔	✔	✔
LVCMOS25	✔	✔	✔	✔	✔	-
LVCMOS18	✔	✔	✔	✔	-	-
LVCMOS15	✔	✔	✔	-	-	-
LVCMOS12	✔	-	-	-	-	-

High output current drive strength and FAST output slew rates generally result in fastest I/O performance. However, these same settings generally also result in transmission line effects on the printed circuit board (PCB) for all but the shortest board traces. Each IOB has independent slew rate and drive strength controls. Use the slowest slew rate and lowest output drive current that meets the performance requirements for the end application.

Likewise, due to lead inductance, a given package supports a limited number of simultaneous switching outputs (SSOs) when using fast, high-drive outputs. Only use fast, high-drive outputs when required by the application.

The SLICEM pair supports two additional functions:

- Two 16x1 distributed RAM blocks, RAM16
- Two 16-bit shift registers, SRL16

Each of these elements is described in more detail in the following sections.

Logic Cells

The combination of a LUT and a storage element is known as a "Logic Cell". The additional features in a slice, such as the wide multiplexers, carry logic, and arithmetic gates, add to the capacity of a slice, implementing logic that would otherwise require additional LUTs. Benchmarks have shown that the overall slice is equivalent to 2.25 simple logic cells. This calculation provides the equivalent logic cell count shown in Table 9.

Slice Details

Figure 15 is a detailed diagram of the SLICEM. It represents a superset of the elements and connections to be found in all slices. The dashed and gray lines (blue when viewed in color) indicate the resources found only in the SLICEM and not in the SLICEL.

Each slice has two halves, which are differentiated as top and bottom to keep them distinct from the upper and lower slices in a CLB. The control inputs for the clock (CLK), Clock Enable (CE), Slice Write Enable (SLICEWE1), and Reset/Set (RS) are shared in common between the two halves.

The LUTs located in the top and bottom portions of the slice are referred to as "G" and "F", respectively, or the "G-LUT" and the "F-LUT". The storage elements in the top and bottom portions of the slice are called FFY and FFX, respectively.

Each slice has two multiplexers with F5MUX in the bottom portion of the slice and FiMUX in the top portion. Depending on the slice, the FiMUX takes on the name F6MUX, F7MUX, or F8MUX, according to its position in the multiplexer chain. The lower SLICEL and SLICEM both have an F6MUX. The upper SLICEM has an F7MUX, and the upper SLICEL has an F8MUX.

The carry chain enters the bottom of the slice as CIN and exits at the top as COUT. Five multiplexers control the chain: CYINIT, CY0F, and CYMUXF in the bottom portion and CY0G and CYMUXG in the top portion. The dedicated arithmetic logic includes the exclusive-OR gates XORF and XORG (bottom and top portions of the slice, respectively) as well as the AND gates FAND and GAND (bottom and top portions, respectively).

See Table 10 for a description of all the slice input and output signals.

Table 10: **Slice Inputs and Outputs**

Name	Location	Direction	Description
F[4:1]	SLICEL/M Bottom	Input	F-LUT and FAND inputs
G[4:1]	SLICEL/M Top	Input	G-LUT and GAND inputs or Write Address (SLICEM)
BX	SLICEL/M Bottom	Input	Bypass to or output (SLICEM) or storage element, or control input to F5MUX, input to carry logic, or data input to RAM (SLICEM)
BY	SLICEL/M Top	Input	Bypass to or output (SLICEM) or storage element, or control input to FiMUX, input to carry logic, or data input to RAM (SLICEM)
BXOUT	SLICEM Bottom	Output	BX bypass output
BYOUT	SLICEM Top	Output	BY bypass output
ALTDIG	SLICEM Top	Input	Alternate data input to RAM
DIG	SLICEM Top	Output	ALTDIG or SHIFTIN bypass output
SLICEWE1	SLICEM Common	Input	RAM Write Enable
F5	SLICEL/M Bottom	Output	Output from F5MUX; direct feedback to FiMUX
FXINA	SLICEL/M Top	Input	Input to FiMUX; direct feedback from F5MUX or another FiMUX
FXINB	SLICEL/M Top	Input	Input to FiMUX; direct feedback from F5MUX or another FiMUX
Fi	SLICEL/M Top	Output	Output from FiMUX; direct feedback to another FiMUX
CE	SLICEL/M Common	Input	FFX/Y Clock Enable
SR	SLICEL/M Common	Input	FFX/Y Set or Reset or RAM Write Enable (SLICEM)
CLK	SLICEL/M Common	Input	FFX/Y Clock or RAM Clock (SLICEM)
SHIFTIN	SLICEM Top	Input	Data input to G-LUT RAM

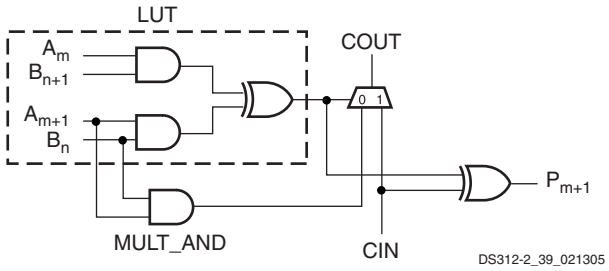

Figure 24: **Using the MULT_AND for Multiplication in Carry Logic**

The MULT_AND is useful for small multipliers. Larger multipliers can be built using the dedicated 18x18 multiplier blocks (see Dedicated Multipliers).

Storage Elements

The storage element, which is programmable as either a D-type flip-flop or a level-sensitive transparent latch, provides a means for synchronizing data to a clock signal, among other uses. The storage elements in the top and bottom portions of the slice are called FFY and FFX, respectively. FFY has a fixed multiplexer on the D input selecting either the combinatorial output Y or the bypass signal BY. FFX selects between the combinatorial output X or the bypass signal BX.

The functionality of a slice storage element is identical to that described earlier for the I/O storage elements. All signals have programmable polarity; the default active-High function is described.

Table 15: **Storage Element Signals**

Signal	Description
D	Input. For a flip-flop data on the D input is loaded when R and S (or CLR and PRE) are Low and CE is High during the Low-to-High clock transition. For a latch, Q reflects the D input while the gate (G) input and gate enable (GE) are High and R and S (or CLR and PRE) are Low. The data on the D input during the High-to-Low gate transition is stored in the latch. The data on the Q output of the latch remains unchanged as long as G or GE remains Low.
Q	Output. Toggles after the Low-to-High clock transition for a flip-flop and immediately for a latch.
C	Clock for edge-triggered flip-flops.
G	Gate for level-sensitive latches.
CE	Clock Enable for flip-flops.
GE	Gate Enable for latches.
S	Synchronous Set (Q = High). When the S input is High and R is Low, the flip-flop is set, output High, during the Low-to-High clock (C) transition. A latch output is immediately set, output High.
R	Synchronous Reset (Q = Low); has precedence over Set.
PRE	Asynchronous Preset (Q = High). When the PRE input is High and CLR is Low, the flip-flop is set, output High, during the Low-to-High clock (C) transition. A latch output is immediately set, output High.
CLR	Asynchronous Clear (Q = Low); has precedence over Preset to reset Q output Low
SR	CLB input for R, S, CLR, or PRE
REV	CLB input for opposite of SR. Must be asynchronous or synchronous to match SR.

The control inputs R, S, CE, and C are all shared between the two flip-flops in a slice.

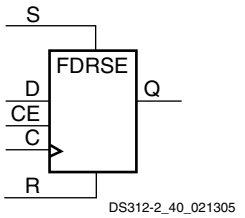

Figure 25: **FD Flip-Flop Component with Synchronous Reset, Set, and Clock Enable**

Table 16: **FD Flip-Flop Functionality with Synchronous Reset, Set, and Clock Enable**

Inputs					Outputs
R	S	CE	D	C	Q
1	X	X	X	↑	0
0	1	X	X	↑	1
0	0	0	X	X	No Change
0	0	1	1	↑	1
0	0	1	0	↑	0

Accommodating Input Frequencies Beyond Specified Maximums

If the CLKIN input frequency exceeds the maximum permitted, divide it down to an acceptable value using the CLKIN_DIVIDE_BY_2 attribute. When this attribute is set to "TRUE", the CLKIN frequency is divided by a factor of two as it enters the DCM. In addition, the CLKIN_DIVIDE_BY_2 option produces a 50% duty-cycle on the input clock, although at half the CLKIN frequency.

Quadrant and Half-Period Phase Shift Outputs

In addition to CLK0 for zero-phase alignment to the CLKIN signal, the DLL also provides the CLK90, CLK180, and CLK270 outputs for 90°, 180°, and 270° phase-shifted signals, respectively. These signals are described in Table 28, page 47 and their relative timing is shown in Figure 43. For control in finer increments than 90°, see Phase Shifter (PS).

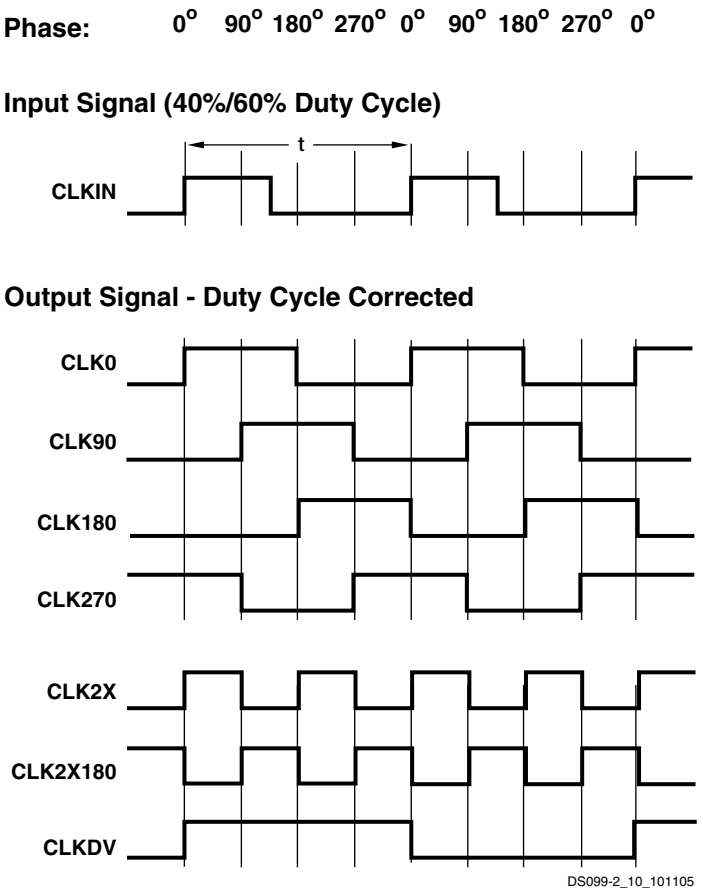

Figure 43: **Characteristics of the DLL Clock Outputs**

Basic Frequency Synthesis Outputs

The DLL component provides basic options for frequency multiplication and division in addition to the more flexible synthesis capability of the DFS component, described in a later section. These operations result in output clock signals with frequencies that are either a fraction (for division) or a multiple (for multiplication) of the incoming clock frequency. The CLK2X output produces an in-phase signal that is twice the frequency of CLKIN. The CLK2X180 output also doubles the frequency, but is 180° out-of-phase with respect to CLKIN. The CLKDIV output generates a clock frequency that is a predetermined fraction of the CLKIN frequency. The CLKDV_DIVIDE attribute determines the factor used to divide the CLKIN frequency. The attribute can be set to various values as described in Table 29. The basic frequency synthesis outputs are described in Table 28.

Duty Cycle Correction of DLL Clock Outputs

The DLL output signals exhibit a 50% duty cycle, even if the incoming CLKIN signal has a different duty cycle. Fifty-percent duty cycle means that the High and Low times of each clock cycle are equal.

DLL Performance Differences Between Steppings

As indicated in Digital Clock Manager (DCM) Timing (Module 3), the Stepping 1 revision silicon supports higher maximum input and output frequencies. Stepping 1 devices are backwards compatible with Stepping 0 devices.

Digital Frequency Synthesizer (DFS)

The DFS unit generates clock signals where the output frequency is a product of the CLKIN input clock frequency and a ratio of two user-specified integers. The two dedicated outputs from the DFS unit, CLKFX and CLKFX180, are defined in Table 33.

Table 33: **DFS Signals**

Signal	Direction	Description
CLKFX	Output	Multiplies the CLKIN frequency by the attribute-value ratio (CLKFX_MULTIPLY/ CLKFX_DIVIDE) to generate a clock signal with a new target frequency.
CLKFX180	Output	Generates a clock signal with the same frequency as CLKFX, but shifted 180° out-of-phase.

The signal at the CLKFX180 output is essentially an inversion of the CLKFX signal. These two outputs always exhibit a 50% duty cycle, even when the CLKIN signal does not. The DFS clock outputs are active coincident with the seven DLL outputs and their output phase is controlled by the Phase Shifter unit (PS).

The output frequency (f_{CLKFX}) of the DFS is a function of the incoming clock frequency (f_{CLKIN}) and two integer attributes, as follows.

$$f_{CLKFX} = f_{CLKIN} \bullet \left(\frac{\text{CLKFX\_MULTIPLY}}{\text{CLKFX\_DIVIDE}} \right) \qquad \text{Eq 1}$$

The CLKFX_MULTIPLY attribute is an integer ranging from 2 to 32, inclusive, and forms the numerator in Equation 1.

Configuration

For additional information on configuration, refer to UG332: *Spartan-3 Generation Configuration User Guide*.

Differences from Spartan-3 FPGAs

In general, Spartan-3E FPGA configuration modes are a superset to those available in Spartan-3 FPGAs. Two new modes added in Spartan-3E FPGAs provide a glueless configuration interface to industry-standard parallel NOR Flash and SPI serial Flash memories.

Configuration Process

The function of a Spartan-3E FPGA is defined by loading application-specific configuration data into the FPGA's internal, reprogrammable CMOS configuration latches (CCLs), similar to the way a microprocessor's function is defined by its application program. For FPGAs, this configuration process uses a subset of the device pins, some of which are dedicated to configuration; other pins are merely borrowed and returned to the application as general-purpose user I/Os after configuration completes.

Spartan-3E FPGAs offer several configuration options to minimize the impact of configuration on the overall system design. In some configuration modes, the FPGA generates a clock and loads itself from an external memory source, either serially or via a byte-wide data path. Alternatively, an external host such as a microprocessor downloads the FPGA's configuration data using a simple synchronous serial interface or via a byte-wide peripheral-style interface. Furthermore, multiple-FPGA designs share a single configuration memory source, creating a structure called a daisy chain.

Three FPGA pins—M2, M1, and M0—select the desired configuration mode. The mode pin settings appear in Table 44. The mode pin values are sampled during the start of configuration when the FPGA's INIT_B output goes High. After the FPGA completes configuration, the mode pins are available as user I/Os.

Table 44: **Spartan-3E Configuration Mode Options and Pin Settings**

	Master Serial	SPI	BPI	Slave Parallel	Slave Serial	JTAG
M[2:0] mode pin settings	<0:0:0>	<0:0:1>	<0:1:0>=Up <0:1:1>=Down	<1:1:0>	<1:1:1>	<1:0:1>
Data width	Serial	Serial	Byte-wide	Byte-wide	Serial	Serial
Configuration memory source	Xilinx Platform Flash	Industry-standard SPI serial Flash	Industry-standard parallel NOR Flash or Xilinx parallel Platform Flash	Any source via microcontroller, CPU, Xilinx parallel Platform Flash, etc.	Any source via microcontroller, CPU, Xilinx Platform Flash, etc.	Any source via microcontroller, CPU, System ACE™ CF, etc.
Clock source	Internal oscillator	Internal oscillator	Internal oscillator	External clock on CCLK pin	External clock on CCLK pin	External clock on TCK pin
Total I/O pins borrowed during configuration	8	13	46	21	8	0
Configuration mode for downstream daisy-chained FPGAs	Slave Serial	Slave Serial	Slave Parallel	Slave Parallel or Memory Mapped	Slave Serial	JTAG
Stand-alone FPGA applications (no external download host)	✓	✓	✓	Possible using XCFxxP Platform Flash, which optionally generates CCLK	Possible using XCFxxP Platform Flash, which optionally generates CCLK	
Uses low-cost, industry-standard Flash		✓	✓			
Supports optional MultiBoot, multi-configuration mode			✓			

This addressing flexibility allows the FPGA to share the parallel Flash PROM with an external or embedded processor. Depending on the specific processor architecture, the processor boots either from the top or bottom of memory. The FPGA is flexible and boots from the opposite end of memory from the processor. Only the processor or the FPGA can boot at any given time. The FPGA can configure first, holding the processor in reset or the processor can boot first, asserting the FPGA's PROG_B pin.

The mode select pins, M[2:0], are sampled when the FPGA's INIT_B output goes High and must be at defined logic levels during this time. After configuration, when the FPGA's DONE output goes High, the mode pins are available as full-featured user-I/O pins.

(P) Similarly, the FPGA's HSWAP pin must be Low to enable pull-up resistors on all user-I/O pins or High to disable the pull-up resistors. The HSWAP control must remain at a constant logic level throughout FPGA configuration. After configuration, when the FPGA's DONE output goes High, the HSWAP pin is available as full-featured user-I/O pin and is powered by the VCCO_0 supply.

The RDWR_B and CSI_B must be Low throughout the configuration process. After configuration, these pins also become user I/O.

In a single-FPGA application, the FPGA's CSO_B and CCLK pins are not used but are actively driving during the configuration process. The BUSY pin is not used but also actively drives during configuration and is available as a user I/O after configuration.

After configuration, all of the interface pins except DONE and PROG_B are available as user I/Os. Furthermore, the bidirectional SelectMAP configuration peripheral interface (see Slave Parallel Mode) is available after configuration. To continue using SelectMAP mode, set the ***Persist*** bitstream generator option to ***Yes***. An external host can then read and verify configuration data.

The Persist option will maintain A20-A23 as configuration pins although they are not used in SelectMAP mode.

Table 59: **Byte-Wide Peripheral Interface (BPI) Connections**

Pin Name	FPGA Direction	Description	During Configuration	After Configuration
HSWAP (P)	Input	**User I/O Pull-Up Control**. When Low during configuration, enables pull-up resistors in all I/O pins to respective I/O bank V_{CCO} input. 0: Pull-ups during configuration 1: No pull-ups	Drive at valid logic level throughout configuration.	User I/O
M[2:0] (A)	Input	**Mode Select**. Selects the FPGA configuration mode. See Design Considerations for the HSWAP, M[2:0], and VS[2:0] Pins.	M2 = 0, M1 = 1. Set M0 = 0 to start at address 0, increment addresses. Set M0 = 1 to start at address 0xFFFFFF and decrement addresses. Sampled when INIT_B goes High.	User I/O
CSI_B	Input	**Chip Select Input**. Active Low.	Must be Low throughout configuration.	User I/O. If bitstream option ***Persist=Yes***, becomes part of SelectMap parallel peripheral interface.
RDWR_B	Input	**Read/Write Control**. Active Low write enable. Read functionality typically only used after configuration, if bitstream option ***Persist=Yes***.	Must be Low throughout configuration.	User I/O. If bitstream option ***Persist=Yes***, becomes part of SelectMap parallel peripheral interface.
LDC0	Output	PROM Chip Enable	Connect to PROM chip-select input (CE#). FPGA drives this signal Low throughout configuration.	User I/O. If the FPGA does not access the PROM after configuration, drive this pin High to deselect the PROM. A[23:0], D[7:0], LDC[2:1], and HDC then become available as user I/O.
LDC1	Output	PROM Output Enable	Connect to the PROM output-enable input (OE#). The FPGA drives this signal Low throughout configuration.	User I/O

After the FPGA configures itself using BPI mode from one end of the parallel Flash PROM, then the FPGA can trigger a MultiBoot event and reconfigure itself from the opposite end of the parallel Flash PROM. MultiBoot is only available when using BPI mode and only for applications with a single Spartan-3E FPGA.

By default, MultiBoot mode is disabled. To trigger a MultiBoot event, assert a Low pulse lasting at least 300 ns on the MultiBoot Trigger (MBT) input to the STARTUP_SPARTAN3E library primitive. When the MBT signal returns High after the 300 ns or longer pulse, the FPGA automatically reconfigures from the opposite end of the parallel Flash memory.

Figure 60 shows an example usage. At power up, the FPGA loads itself from the attached parallel Flash PROM. In this example, the M0 mode pin is Low so the FPGA starts at address 0 and increments through the Flash PROM memory locations. After the FPGA completes configuration, the application initially loaded into the FPGA performs a board-level or system test using FPGA logic. If the test is successful, the FPGA then triggers a MultiBoot event, causing the FPGA to reconfigure from the opposite end of the Flash PROM memory. This second configuration contains the FPGA application for normal operation.

Similarly, the general FPGA application could trigger another MultiBoot event at any time to reload the diagnostics design, and so on.

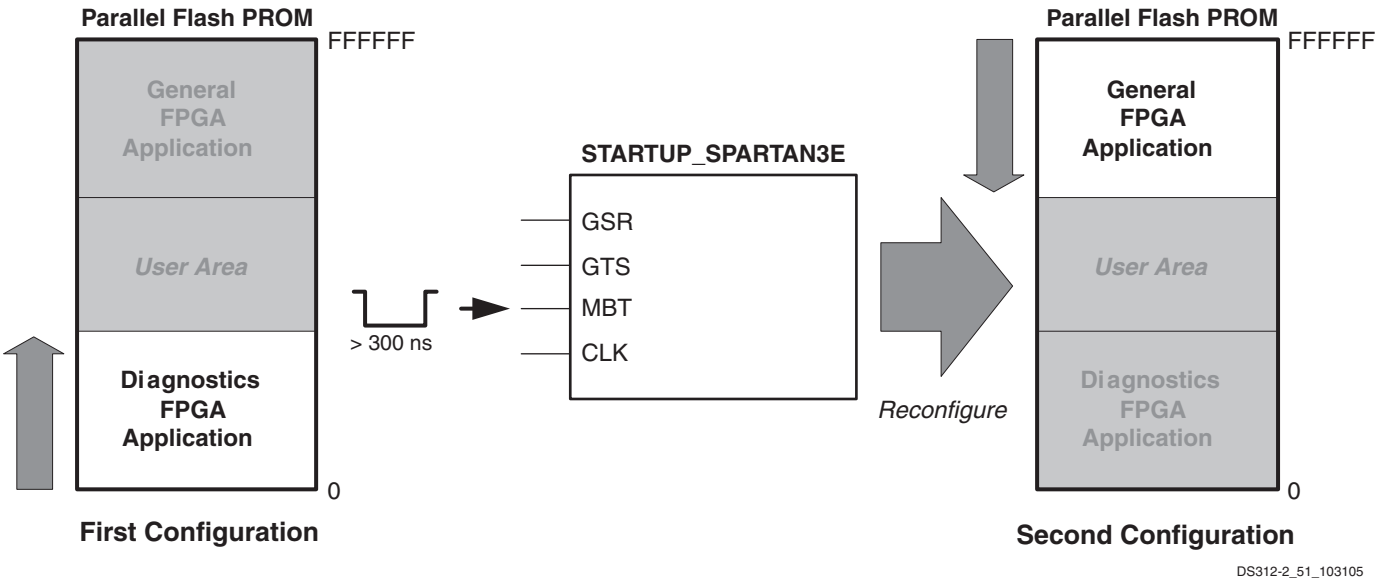

Figure 60: **Use MultiBoot to Load Alternate Configuration Images**

In another potential application, the initial design loaded into the FPGA image contains a “golden” or “fail-safe” configuration image, which then communicates with the outside world and checks for a newer image. If there is a new configuration revision and the new image verifies as good, the “golden” configuration triggers a MultiBoot event to load the new image.

When a MultiBoot event is triggered, the FPGA then again drives its configuration pins as described in Table 59. However, the FPGA does not assert the PROG_B pin. The system design must ensure that no other device drives on these same pins during the reconfiguration process. The FPGA’s DONE, LDC[2:0], or HDC pins can temporarily disable any conflicting drivers during reconfiguration.

Asserting the PROG_B pin Low overrides the MultiBoot feature and forces the FPGA to reconfigure starting from the end of memory defined by the mode pins, shown in Table 58.

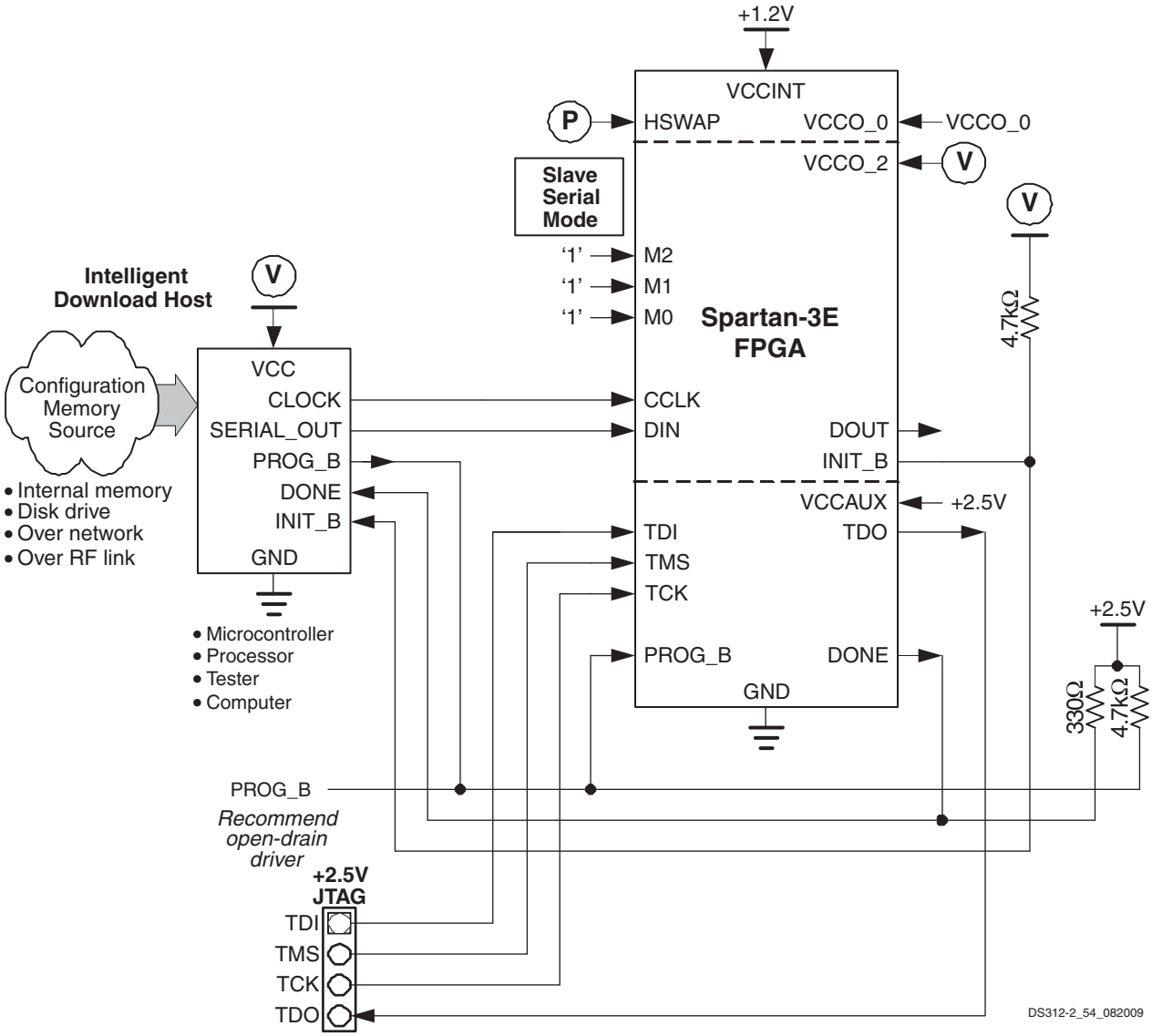

Figure 63: **Slave Serial Configuration**

The mode select pins, M[2:0], are sampled when the FPGA's INIT_B output goes High and must be at defined logic levels during this time. After configuration, when the FPGA's DONE output goes High, the mode pins are available as full-featured user-I/O pins.

(P) Similarly, the FPGA's HSWAP pin must be Low to enable pull-up resistors on all user-I/O pins or High to disable the pull-up resistors. The HSWAP control must remain at a constant logic level throughout FPGA configuration. After configuration, when the FPGA's DONE output goes High, the HSWAP pin is available as full-featured user-I/O pin and is powered by the VCCO_0 supply.

Voltage Compatibility

(V) Most Slave Serial interface signals are within the FPGA's I/O Bank 2, supplied by the VCCO_2 supply input. The VCCO_2 voltage can be 3.3V, 2.5V, or 1.8V to match the requirements of the external host, ideally 2.5V. Using 3.3V or 1.8V requires additional design considerations as the DONE and PROG_B pins are powered by the FPGA's 2.5V V_{CCAUX} supply. See XAPP453: *The 3.3V Configuration of Spartan-3 FPGAs* for additional information.

Daisy-Chaining

If the application requires multiple FPGAs with different configurations, then configure the FPGAs using a daisy chain, as shown in Figure 64. Use Slave Serial mode (M[2:0] = <1:1:1>) for all FPGAs in the daisy-chain. After the lead FPGA is filled with its configuration data, the lead

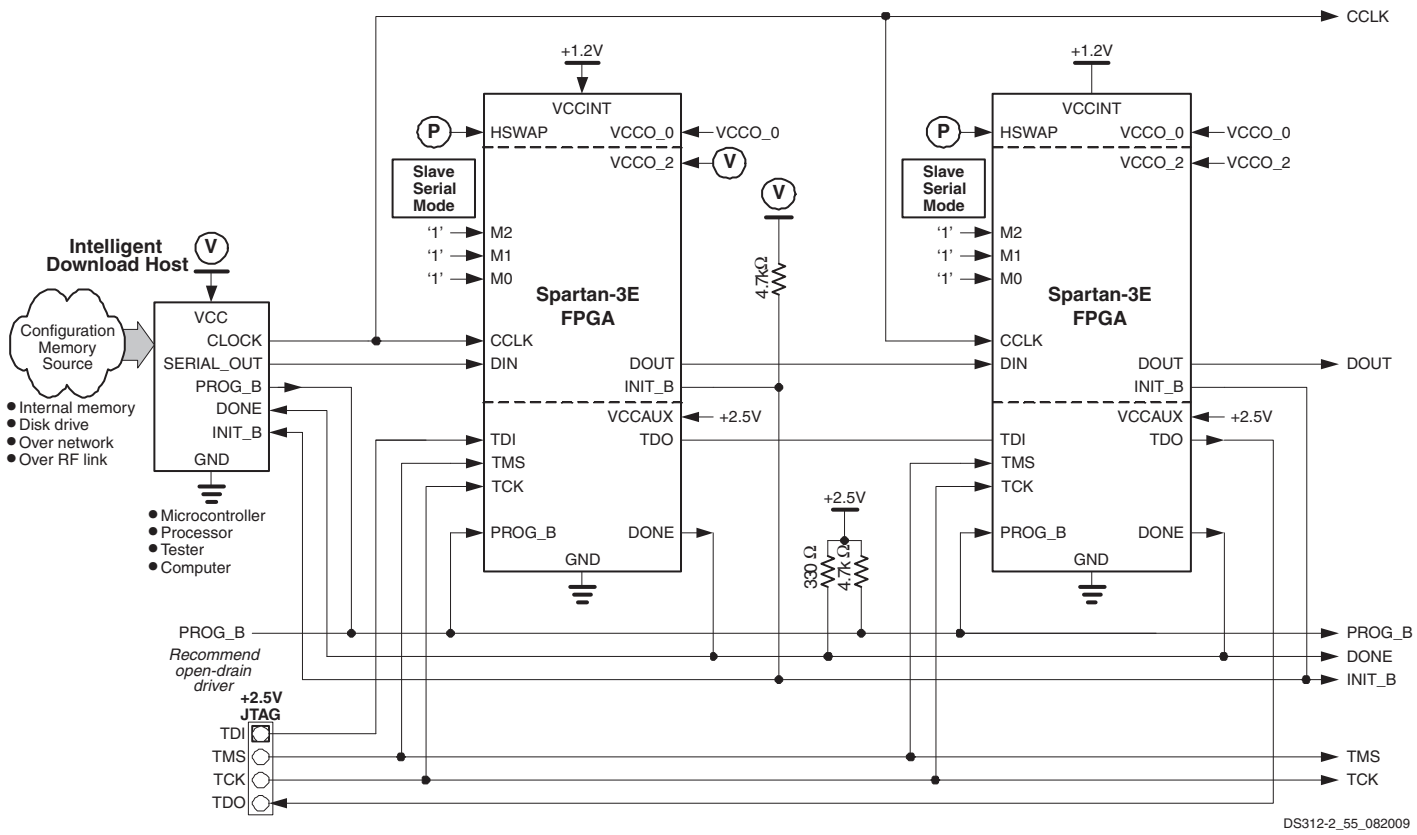

Figure 64: **Daisy-Chaining using Slave Serial Mode**

JTAG Mode

For additional information, refer to the "JTAG Configuration Mode and Boundary-Scan" chapter in UG332.

The Spartan-3E FPGA has a dedicated four-wire IEEE 1149.1/1532 JTAG port that is always available any time the FPGA is powered and regardless of the mode pin settings. However, when the FPGA mode pins are set for JTAG mode (M[2:0] = <1:0:1>), the FPGA waits to be configured via the JTAG port after a power-on event or when PROG_B is asserted. Selecting the JTAG mode simply disables the other configuration modes. No other pins are required as part of the configuration interface.

Figure 65 illustrates a JTAG-only configuration interface. The JTAG interface is easily cascaded to any number of FPGAs by connecting the TDO output of one device to the TDI input of the next device in the chain. The TDO output of the last device in the chain loops back to the port connector.

Design Note

If using software versions prior to ISE 9.1.01i, avoid configuring the FPGA using JTAG if...

- the mode pins are set for a Master mode
- the attached Master mode PROM contains a valid FPGA configuration bitstream.

The FPGA bitstream may be corrupted and the DONE pin may go High. The following Answer Record contains additional information.

http://www.xilinx.com/support/answers/22255.htm

Configurable Logic Block (CLB) Timing

Table 98: **CLB (SLICEM) Timing**

Symbol	Description	Speed Grade				Units
		-5		-4		
		Min	Max	Min	Max	
Clock-to-Output Times						
T_{CKO}	When reading from the FFX (FFY) Flip-Flop, the time from the active transition at the CLK input to data appearing at the XQ (YQ) output	-	0.52	-	0.60	ns
Setup Times						
T_{AS}	Time from the setup of data at the F or G input to the active transition at the CLK input of the CLB	0.46	-	0.52	-	ns
T_{DICK}	Time from the setup of data at the BX or BY input to the active transition at the CLK input of the CLB	1.58	-	1.81	-	ns
Hold Times						
T_{AH}	Time from the active transition at the CLK input to the point where data is last held at the F or G input	0	-	0	-	ns
T_{CKDI}	Time from the active transition at the CLK input to the point where data is last held at the BX or BY input	0	-	0	-	ns
Clock Timing						
T_{CH}	The High pulse width of the CLB's CLK signal	0.70	-	0.80	-	ns
T_{CL}	The Low pulse width of the CLK signal	0.70	-	0.80	-	ns
F_{TOG}	Toggle frequency (for export control)	0	657	0	572	MHz
Propagation Times						
T_{ILO}	The time it takes for data to travel from the CLB's F (G) input to the X (Y) output	-	0.66	-	0.76	ns
Set/Reset Pulse Width						
T_{RPW_CLB}	The minimum allowable pulse width, High or Low, to the CLB's SR input	1.57	-	1.80	-	ns

Notes:

1. The numbers in this table are based on the operating conditions set forth in Table 77.

Block RAM Timing

Table 103: **Block RAM Timing**

Symbol	Description	Speed Grade				Units
		-5		-4		
		Min	Max	Min	Max	
Clock-to-Output Times						
T_{BCKO}	When reading from block RAM, the delay from the active transition at the CLK input to data appearing at the DOUT output	-	2.45	-	2.82	ns
Setup Times						
T_{BACK}	Setup time for the ADDR inputs before the active transition at the CLK input of the block RAM	0.33	-	0.38	-	ns
T_{BDCK}	Setup time for data at the DIN inputs before the active transition at the CLK input of the block RAM	0.23	-	0.23	-	ns
T_{BECK}	Setup time for the EN input before the active transition at the CLK input of the block RAM	0.67	-	0.77	-	ns
T_{BWCK}	Setup time for the WE input before the active transition at the CLK input of the block RAM	1.09	-	1.26	-	ns
Hold Times						
T_{BCKA}	Hold time on the ADDR inputs after the active transition at the CLK input	0.12	-	0.14	-	ns
T_{BCKD}	Hold time on the DIN inputs after the active transition at the CLK input	0.12	-	0.13	-	ns
T_{BCKE}	Hold time on the EN input after the active transition at the CLK input	0	-	0	-	ns
T_{BCKW}	Hold time on the WE input after the active transition at the CLK input	0	-	0	-	ns
Clock Timing						
T_{BPWH}	High pulse width of the CLK signal	1.39	-	1.59	-	ns
T_{BPWL}	Low pulse width of the CLK signal	1.39	-	1.59	-	ns
Clock Frequency						
F_{BRAM}	Block RAM clock frequency. RAM read output value written back into RAM, for shift-registers and circular buffers. Write-only or read-only performance is faster.	0	270	0	230	MHz

Notes:

1. The numbers in this table are based on the operating conditions set forth in Table 77.

Table 124: **Types of Pins on Spartan-3E FPGAs *(Cont'd)***

Type / Color Code	Description	Pin Name(s) in Type[1]
VCCAUX	Dedicated auxiliary power supply pin. The number of VCCAUX pins depends on the package used. All must be connected to +2.5V. See the Powering Spartan-3E FPGAs section in Module 2 for details.	VCCAUX
VCCINT	Dedicated internal core logic power supply pin. The number of VCCINT pins depends on the package used. All must be connected to +1.2V. See the Powering Spartan-3E FPGAs section in Module 2 for details.	VCCINT
VCCO	Along with all the other VCCO pins in the same bank, this pin supplies power to the output buffers within the I/O bank and sets the input threshold voltage for some I/O standards. See the Powering Spartan-3E FPGAs section in Module 2 for details.	VCCO_#
N.C.	This package pin is not connected in this specific device/package combination but may be connected in larger devices in the same package.	N.C.

Notes:

1. # = I/O bank number, an integer between 0 and 3.
2. IRDY/TRDY designations are for PCI designs; refer to PCI documentation for details.

Differential Pair Labeling

I/Os with Lxxy_# are part of a differential pair. 'L' indicates differential capability. The 'xx' field is a two-digit integer, unique to each bank that identifies a differential pin-pair. The 'y' field is either 'P' for the true signal or 'N' for the inverted signal in the differential pair. The '#' field is the I/O bank number.

The pin name suffix has the following significance. Figure 79 provides a specific example showing a differential input to and a differential output from Bank 1.

'L' indicates that the pin is part of a differential pair.

'xx' is a two-digit integer, unique for each bank, that identifies a differential pin-pair.

'y' is replaced by 'P' for the true signal or 'N' for the inverted. These two pins form one differential pin-pair.

'#' is an integer, 0 through 3, indicating the associated I/O bank.

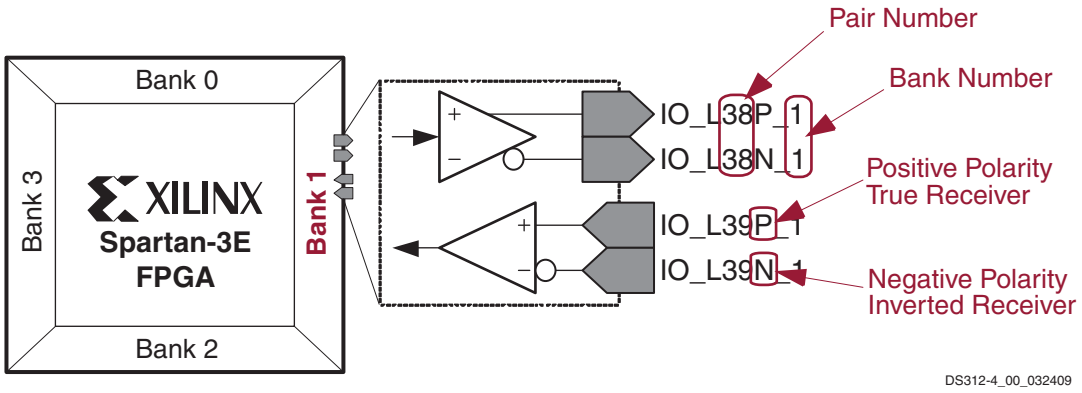

Figure 79: **Differential Pair Labeling**

Table 133: **CP132 Package Pinout *(Cont'd)***

Bank	XC3S100E Pin Name	XC3S250E XC3S500E Pin Name	CP132 Ball	Type
0	VCCO_0	VCCO_0	B10	VCCO
1	IO/A0	IO/A0	F12	DUAL
1	IO/VREF_1	IO/VREF_1	K13	VREF
1	IO_L01N_1/A15	IO_L01N_1/A15	N14	DUAL
1	IO_L01P_1/A16	IO_L01P_1/A16	N13	DUAL
1	IO_L02N_1/A13	IO_L02N_1/A13	M13	DUAL
1	IO_L02P_1/A14	IO_L02P_1/A14	M12	DUAL
1	IO_L03N_1/A11	IO_L03N_1/A11	L14	DUAL
1	IO_L03P_1/A12	IO_L03P_1/A12	L13	DUAL
1	IO_L04N_1/A9/RHCLK1	IO_L04N_1/A9/RHCLK1	J12	RHCLK/DUAL
1	IO_L04P_1/A10/RHCLK0	IO_L04P_1/A10/RHCLK0	K14	RHCLK/DUAL
1	IO_L05N_1/A7/RHCLK3/TRDY1	IO_L05N_1/A7/RHCLK3/TRDY1	J14	RHCLK/DUAL
1	IO_L05P_1/A8/RHCLK2	IO_L05P_1/A8/RHCLK2	J13	RHCLK/DUAL
1	IO_L06N_1/A5/RHCLK5	IO_L06N_1/A5/RHCLK5	H12	RHCLK/DUAL
1	IO_L06P_1/A6/RHCLK4/IRDY1	IO_L06P_1/A6/RHCLK4/IRDY1	H13	RHCLK/DUAL
1	IO_L07N_1/A3/RHCLK7	IO_L07N_1/A3/RHCLK7	G13	RHCLK/DUAL
1	IO_L07P_1/A4/RHCLK6	IO_L07P_1/A4/RHCLK6	G14	RHCLK/DUAL
1	IO_L08N_1/A1	IO_L08N_1/A1	F13	DUAL
1	IO_L08P_1/A2	IO_L08P_1/A2	F14	DUAL
1	IO_L09N_1/LDC0	IO_L09N_1/LDC0	D12	DUAL
1	IO_L09P_1/HDC	IO_L09P_1/HDC	D13	DUAL
1	IO_L10N_1/LDC2	IO_L10N_1/LDC2	C13	DUAL
1	IO_L10P_1/LDC1	IO_L10P_1/LDC1	C14	DUAL
1	IP/VREF_1	IP/VREF_1	G12	VREF
1	VCCO_1	VCCO_1	E13	VCCO
1	VCCO_1	VCCO_1	M14	VCCO
2	IO/D5	IO/D5	P4	DUAL
2	IO/M1	IO/M1	N7	DUAL
2	IP/VREF_2	IO/VREF_2	P11	***100E:*** VREF(INPUT) ***Others:*** VREF(I/O)
2	IO_L01N_2/INIT_B	IO_L01N_2/INIT_B	N1	DUAL
2	IO_L01P_2/CSO_B	IO_L01P_2/CSO_B	M2	DUAL
2	IO_L02N_2/MOSI/CSI_B	IO_L02N_2/MOSI/CSI_B	N2	DUAL
2	IO_L02P_2/DOUT/BUSY	IO_L02P_2/DOUT/BUSY	P1	DUAL
2	IO_L03N_2/D6/GCLK13	IO_L03N_2/D6/GCLK13	N4	DUAL/GCLK
2	IO_L03P_2/D7/GCLK12	IO_L03P_2/D7/GCLK12	M4	DUAL/GCLK
2	IO_L04N_2/D3/GCLK15	IO_L04N_2/D3/GCLK15	N5	DUAL/GCLK
2	IO_L04P_2/D4/GCLK14	IO_L04P_2/D4/GCLK14	M5	DUAL/GCLK
2	IO_L06N_2/D1/GCLK3	IO_L06N_2/D1/GCLK3	P7	DUAL/GCLK
2	IO_L06P_2/D2/GCLK2	IO_L06P_2/D2/GCLK2	P6	DUAL/GCLK
2	IO_L07N_2/DIN/D0	IO_L07N_2/DIN/D0	N8	DUAL
2	IO_L07P_2/M0	IO_L07P_2/M0	P8	DUAL
2	N.C. (◆)	IO_L08N_2/A22	M9	***100E:*** N.C. ***Others:*** DUAL

PQ208 Footprint (Left)

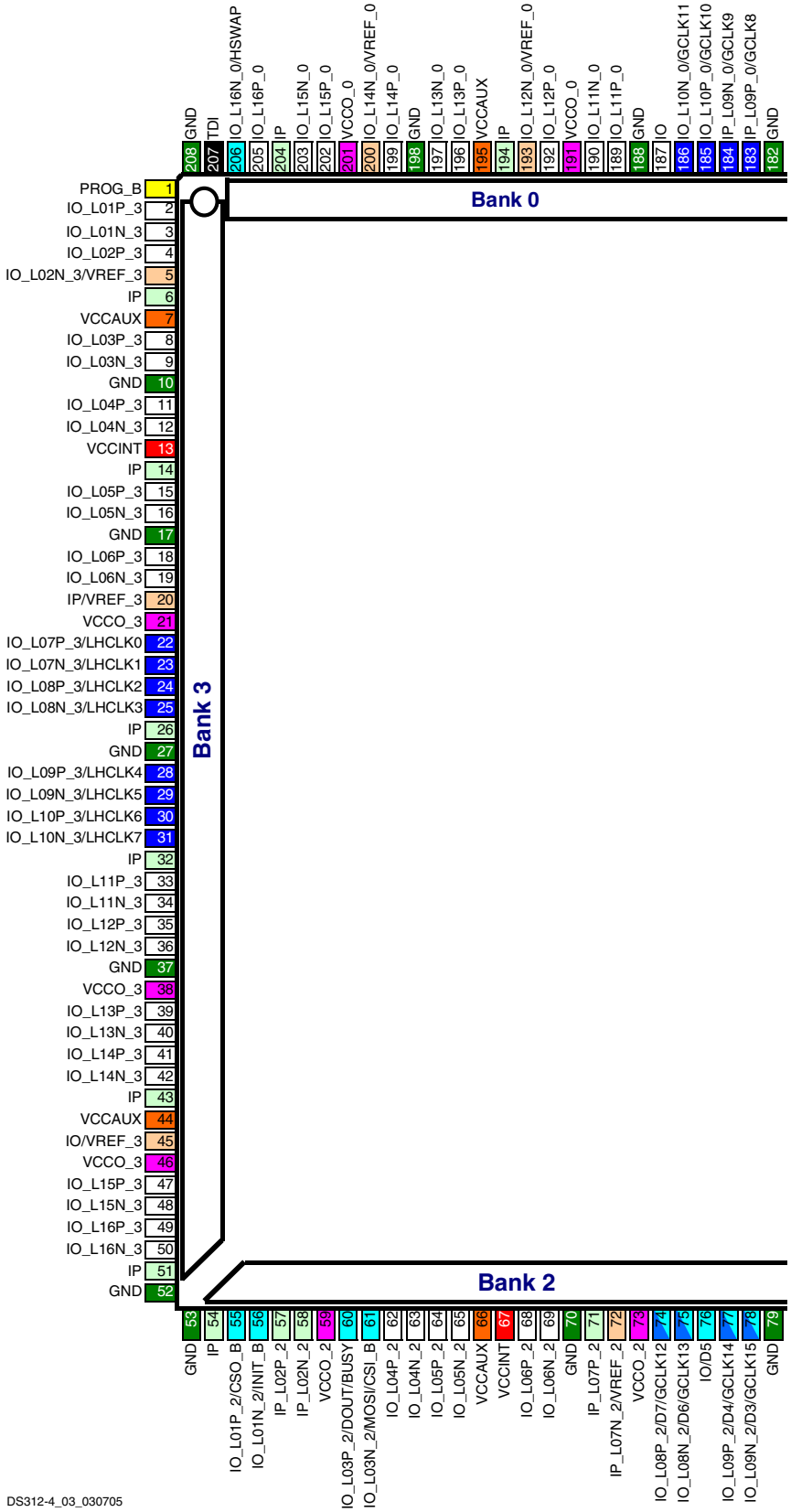

Figure 83: **PQ208 Footprint (Left)**

PQ208 Footprint (Right)

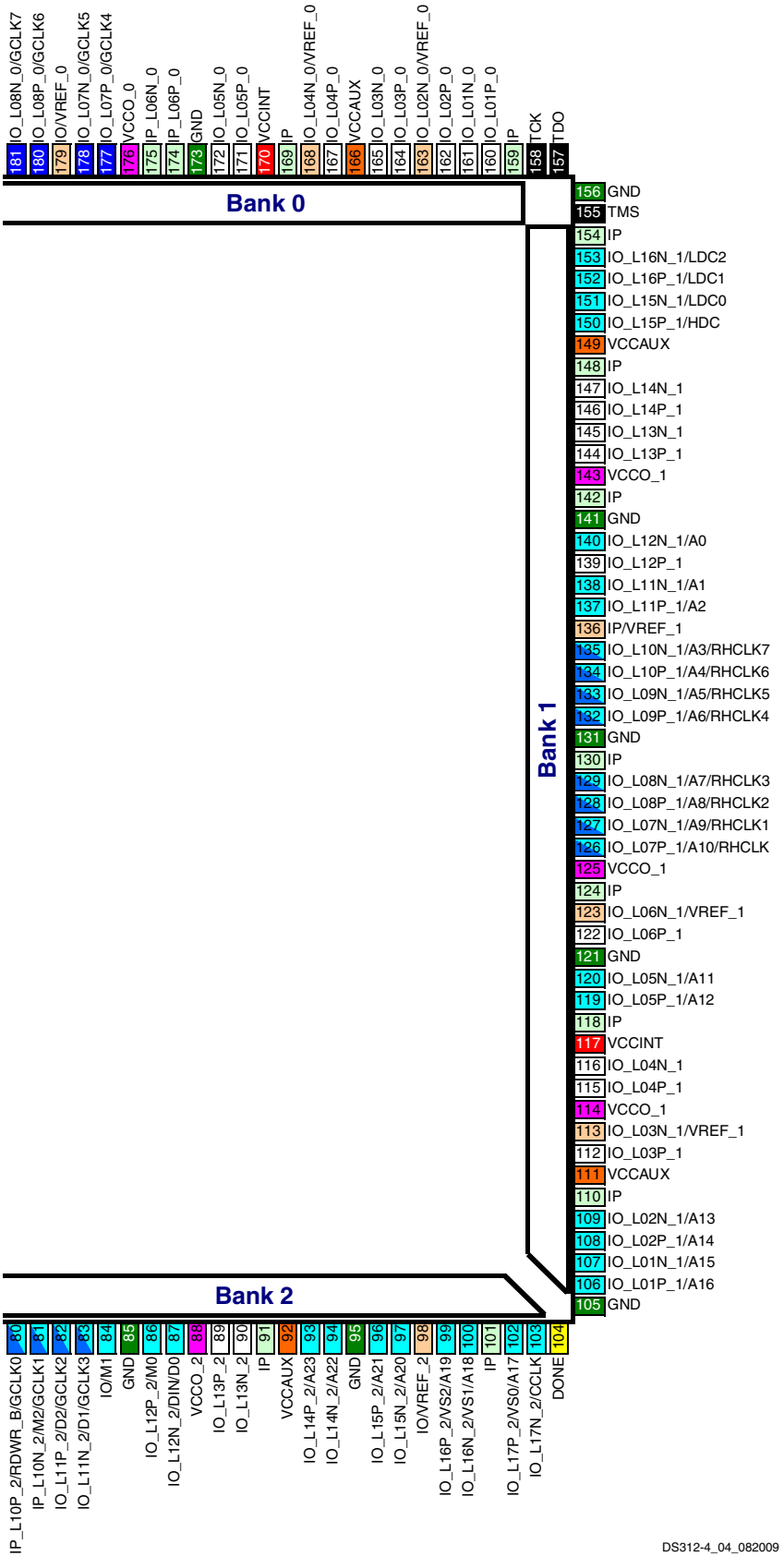

Figure 84: **PQ208 Footprint (Right)**

User I/Os by Bank

Table 149 and Table 150 indicate how the available user-I/O pins are distributed between the four I/O banks on the FG320 package.

Table 149: **User I/Os Per Bank for XC3S500E in the FG320 Package**

Package Edge	I/O Bank	Maximum I/O	All Possible I/O Pins by Type				
			I/O	INPUT	DUAL	VREF(1)	CLK(2)
Top	0	58	29	14	1	6	8
Right	1	58	22	10	21	5	0(2)
Bottom	2	58	17	13	24	4	0(2)
Left	3	58	34	11	0	5	8
TOTAL		**232**	**102**	**48**	**46**	**20**	**16**

Notes:

1. Some VREF and CLK pins are on INPUT pins.
2. The eight global clock pins in this bank have optional functionality during configuration and are counted in the DUAL column.

Table 150: **User I/Os Per Bank for XC3S1200E and XC3S1600E in the FG320 Package**

Package Edge	I/O Bank	Maximum I/O	All Possible I/O Pins by Type				
			I/O	INPUT	DUAL	VREF(1)	CLK(2)
Top	0	61	34	12	1	6	8
Right	1	63	25	12	21	5	0(2)
Bottom	2	63	23	11	24	5	0(2)
Left	3	63	38	12	0	5	8
TOTAL		**250**	**120**	**47**	**46**	**21**	**16**

Notes:

1. Some VREF and CLK pins are on INPUT pins.
2. The eight global clock pins in this bank have optional functionality during configuration and are counted in the DUAL column.

Table 152: **FG400 Package Pinout *(Cont'd)***

Bank	XC3S1200E XC3S1600E Pin Name	FG400 Ball	Type
0	IP_L17P_0/GCLK8	H10	GCLK
0	IP_L20N_0	G9	INPUT
0	IP_L20P_0	G8	INPUT
0	IP_L23N_0	C8	INPUT
0	IP_L23P_0	D8	INPUT
0	IP_L26N_0	E6	INPUT
0	IP_L26P_0	E7	INPUT
0	IP_L29N_0	A4	INPUT
0	IP_L29P_0	A5	INPUT
0	VCCO_0	B4	VCCO
0	VCCO_0	B10	VCCO
0	VCCO_0	B16	VCCO
0	VCCO_0	D7	VCCO
0	VCCO_0	D13	VCCO
0	VCCO_0	F10	VCCO
1	IO_L01N_1/A15	U18	DUAL
1	IO_L01P_1/A16	U17	DUAL
1	IO_L02N_1/A13	T18	DUAL
1	IO_L02P_1/A14	T17	DUAL
1	IO_L03N_1/VREF_1	V19	VREF
1	IO_L03P_1	U19	I/O
1	IO_L04N_1	W20	I/O
1	IO_L04P_1	V20	I/O
1	IO_L05N_1	R18	I/O
1	IO_L05P_1	R17	I/O
1	IO_L06N_1	T20	I/O
1	IO_L06P_1	U20	I/O
1	IO_L07N_1	P18	I/O
1	IO_L07P_1	P17	I/O
1	IO_L08N_1/VREF_1	P20	VREF
1	IO_L08P_1	R20	I/O
1	IO_L09N_1	P16	I/O
1	IO_L09P_1	N16	I/O
1	IO_L10N_1	N19	I/O
1	IO_L10P_1	N18	I/O
1	IO_L11N_1	N15	I/O
1	IO_L11P_1	M15	I/O
1	IO_L12N_1/A11	M18	DUAL
1	IO_L12P_1/A12	M17	DUAL
1	IO_L13N_1/VREF_1	L19	VREF
1	IO_L13P_1	M19	I/O
1	IO_L14N_1/A9/RHCLK1	L16	RHCLK/ DUAL

Table 152: **FG400 Package Pinout *(Cont'd)***

Bank	XC3S1200E XC3S1600E Pin Name	FG400 Ball	Type
1	IO_L14P_1/A10/RHCLK0	M16	RHCLK/ DUAL
1	IO_L15N_1/A7/RHCLK3/ TRDY1	L14	RHCLK/ DUAL
1	IO_L15P_1/A8/RHCLK2	L15	RHCLK/ DUAL
1	IO_L16N_1/A5/RHCLK5	K14	RHCLK/ DUAL
1	IO_L16P_1/A6/RHCLK4/ IRDY1	K13	RHCLK/ DUAL
1	IO_L17N_1/A3/RHCLK7	J20	RHCLK/ DUAL
1	IO_L17P_1/A4/RHCLK6	K20	RHCLK/ DUAL
1	IO_L18N_1/A1	K16	DUAL
1	IO_L18P_1/A2	J16	DUAL
1	IO_L19N_1/A0	J13	DUAL
1	IO_L19P_1	J14	I/O
1	IO_L20N_1	J17	I/O
1	IO_L20P_1	J18	I/O
1	IO_L21N_1	H19	I/O
1	IO_L21P_1	J19	I/O
1	IO_L22N_1	H15	I/O
1	IO_L22P_1	H16	I/O
1	IO_L23N_1	H18	I/O
1	IO_L23P_1	H17	I/O
1	IO_L24N_1/VREF_1	H20	VREF
1	IO_L24P_1	G20	I/O
1	IO_L25N_1	G16	I/O
1	IO_L25P_1	F16	I/O
1	IO_L26N_1	F19	I/O
1	IO_L26P_1	F20	I/O
1	IO_L27N_1	F18	I/O
1	IO_L27P_1	F17	I/O
1	IO_L28N_1	D20	I/O
1	IO_L28P_1	E20	I/O
1	IO_L29N_1/LDC0	D18	DUAL
1	IO_L29P_1/HDC	E18	DUAL
1	IO_L30N_1/LDC2	C19	DUAL
1	IO_L30P_1/LDC1	C20	DUAL
1	IP	B20	INPUT
1	IP	G15	INPUT
1	IP	G18	INPUT
1	IP	H14	INPUT
1	IP	J15	INPUT

Table 152: **FG400 Package Pinout *(Cont'd)***

Bank	XC3S1200E XC3S1600E Pin Name	FG400 Ball	Type
2	IP_L14P_2	T9	INPUT
2	IP_L17N_2/M2/GCLK1	P12	DUAL/ GCLK
2	IP_L17P_2/RDWR_B/ GCLK0	P11	DUAL/ GCLK
2	IP_L20N_2	T12	INPUT
2	IP_L20P_2	R12	INPUT
2	IP_L23N_2/VREF_2	T13	VREF
2	IP_L23P_2	T14	INPUT
2	IP_L26N_2	V14	INPUT
2	IP_L26P_2	V15	INPUT
2	IP_L29N_2	W16	INPUT
2	IP_L29P_2	Y16	INPUT
2	VCCO_2	R11	VCCO
2	VCCO_2	U8	VCCO
2	VCCO_2	U14	VCCO
2	VCCO_2	W5	VCCO
2	VCCO_2	W11	VCCO
2	VCCO_2	W17	VCCO
3	IO_L01N_3	D2	I/O
3	IO_L01P_3	D3	I/O
3	IO_L02N_3/VREF_3	E3	VREF
3	IO_L02P_3	E4	I/O
3	IO_L03N_3	C1	I/O
3	IO_L03P_3	B1	I/O
3	IO_L04N_3	E1	I/O
3	IO_L04P_3	D1	I/O
3	IO_L05N_3	F3	I/O
3	IO_L05P_3	F4	I/O
3	IO_L06N_3	F1	I/O
3	IO_L06P_3	F2	I/O
3	IO_L07N_3	G4	I/O
3	IO_L07P_3	G3	I/O
3	IO_L08N_3	G5	I/O
3	IO_L08P_3	H5	I/O
3	IO_L09N_3/VREF_3	H3	VREF
3	IO_L09P_3	H2	I/O
3	IO_L10N_3	H7	I/O
3	IO_L10P_3	H6	I/O
3	IO_L11N_3	J4	I/O
3	IO_L11P_3	J3	I/O
3	IO_L12N_3	J1	I/O
3	IO_L12P_3	J2	I/O
3	IO_L13N_3	J6	I/O

Table 152: **FG400 Package Pinout *(Cont'd)***

Bank	XC3S1200E XC3S1600E Pin Name	FG400 Ball	Type
3	IO_L13P_3	K6	I/O
3	IO_L14N_3/LHCLK1	K2	LHCLK
3	IO_L14P_3/LHCLK0	K3	LHCLK
3	IO_L15N_3/LHCLK3/IRDY2	L7	LHCLK
3	IO_L15P_3/LHCLK2	K7	LHCLK
3	IO_L16N_3/LHCLK5	L1	LHCLK
3	IO_L16P_3/LHCLK4/TRDY2	M1	LHCLK
3	IO_L17N_3/LHCLK7	L3	LHCLK
3	IO_L17P_3/LHCLK6	M3	LHCLK
3	IO_L18N_3	M7	I/O
3	IO_L18P_3	M8	I/O
3	IO_L19N_3	M4	I/O
3	IO_L19P_3	M5	I/O
3	IO_L20N_3/VREF_3	N6	VREF
3	IO_L20P_3	M6	I/O
3	IO_L21N_3	N2	I/O
3	IO_L21P_3	N1	I/O
3	IO_L22N_3	P7	I/O
3	IO_L22P_3	N7	I/O
3	IO_L23N_3	N4	I/O
3	IO_L23P_3	N3	I/O
3	IO_L24N_3	R1	I/O
3	IO_L24P_3	P1	I/O
3	IO_L25N_3	R5	I/O
3	IO_L25P_3	P5	I/O
3	IO_L26N_3	T2	I/O
3	IO_L26P_3	R2	I/O
3	IO_L27N_3	R4	I/O
3	IO_L27P_3	R3	I/O
3	IO_L28N_3/VREF_3	T1	VREF
3	IO_L28P_3	U1	I/O
3	IO_L29N_3	T3	I/O
3	IO_L29P_3	U3	I/O
3	IO_L30N_3	V1	I/O
3	IO_L30P_3	V2	I/O
3	IP	F5	INPUT
3	IP	G1	INPUT
3	IP	G6	INPUT
3	IP	H1	INPUT
3	IP	J5	INPUT
3	IP	L5	INPUT
3	IP	L8	INPUT
3	IP	M2	INPUT

AUTOMOTIVE APPLICATIONS DISCLAIMER

XILINX PRODUCTS ARE NOT DESIGNED OR INTENDED TO BE FAIL-SAFE, OR FOR USE IN ANY APPLICATION REQUIRING FAIL-SAFE PERFORMANCE, SUCH AS APPLICATIONS RELATED TO: (I) THE DEPLOYMENT OF AIRBAGS, (II) CONTROL OF A VEHICLE, UNLESS THERE IS A FAIL-SAFE OR REDUNDANCY FEATURE (WHICH DOES NOT INCLUDE USE OF SOFTWARE IN THE XILINX DEVICE TO IMPLEMENT THE REDUNDANCY) AND A WARNING SIGNAL UPON FAILURE TO THE OPERATOR, OR (III) USES THAT COULD LEAD TO DEATH OR PERSONAL INJURY. CUSTOMER ASSUMES THE SOLE RISK AND LIABILITY OF ANY USE OF XILINX PRODUCTS IN SUCH APPLICATIONS.