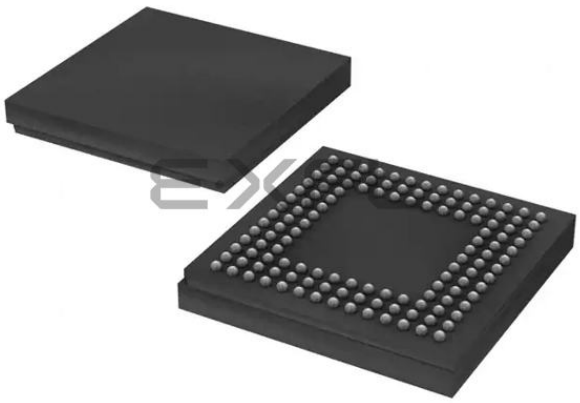
# AMD Xilinx - XC3S100E-4CPG132I Datasheet



**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 | 83 |
| Number of Gates | 100000 |
| Voltage - Supply | 1.14V ~ 1.26V |
| Mounting Type | Surface Mount |
| Operating Temperature | -40°C ~ 100°C (TJ) |
| Package / Case | 132-TFBGA, CSPBGA |
| Supplier Device Package | 132-CSPBGA (8x8) |
| Purchase URL | https://www.e-xfl.com/product-detail/xilinx/xc3s100e-4cpg132i |

## Architectural Overview

The Spartan-3E family architecture consists of five fundamental programmable functional elements:

- **Configurable Logic Blocks (CLBs)** contain flexible Look-Up Tables (LUTs) that implement logic plus storage elements used as flip-flops or latches. CLBs perform a wide variety of logical functions as well as store data.
- **Input/Output Blocks (IOBs)** control the flow of data between the I/O pins and the internal logic of the device. Each IOB supports bidirectional data flow plus 3-state operation. Supports a variety of signal standards, including four high-performance differential standards. Double Data-Rate (DDR) registers are included.
- **Block RAM** provides data storage in the form of 18-Kbit dual-port blocks.
- **Multiplier Blocks** accept two 18-bit binary numbers as inputs and calculate the product.
- **Digital Clock Manager (DCM) Blocks** provide self-calibrating, fully digital solutions for distributing, delaying, multiplying, dividing, and phase-shifting clock signals.

These elements are organized as shown in Figure 1. A ring of IOBs surrounds a regular array of CLBs. Each device has two columns of block RAM except for the XC3S100E, which has one column. Each RAM column consists of several 18-Kbit RAM blocks. Each block RAM is associated with a dedicated multiplier. The DCMs are positioned in the center with two at the top and two at the bottom of the device. The XC3S100E has only one DCM at the top and bottom, while the XC3S1200E and XC3S1600E add two DCMs in the middle of the left and right sides.

The Spartan-3E family features a rich network of traces that interconnect all five functional elements, transmitting signals among them. Each functional element has an associated switch matrix that permits multiple connections to the routing.

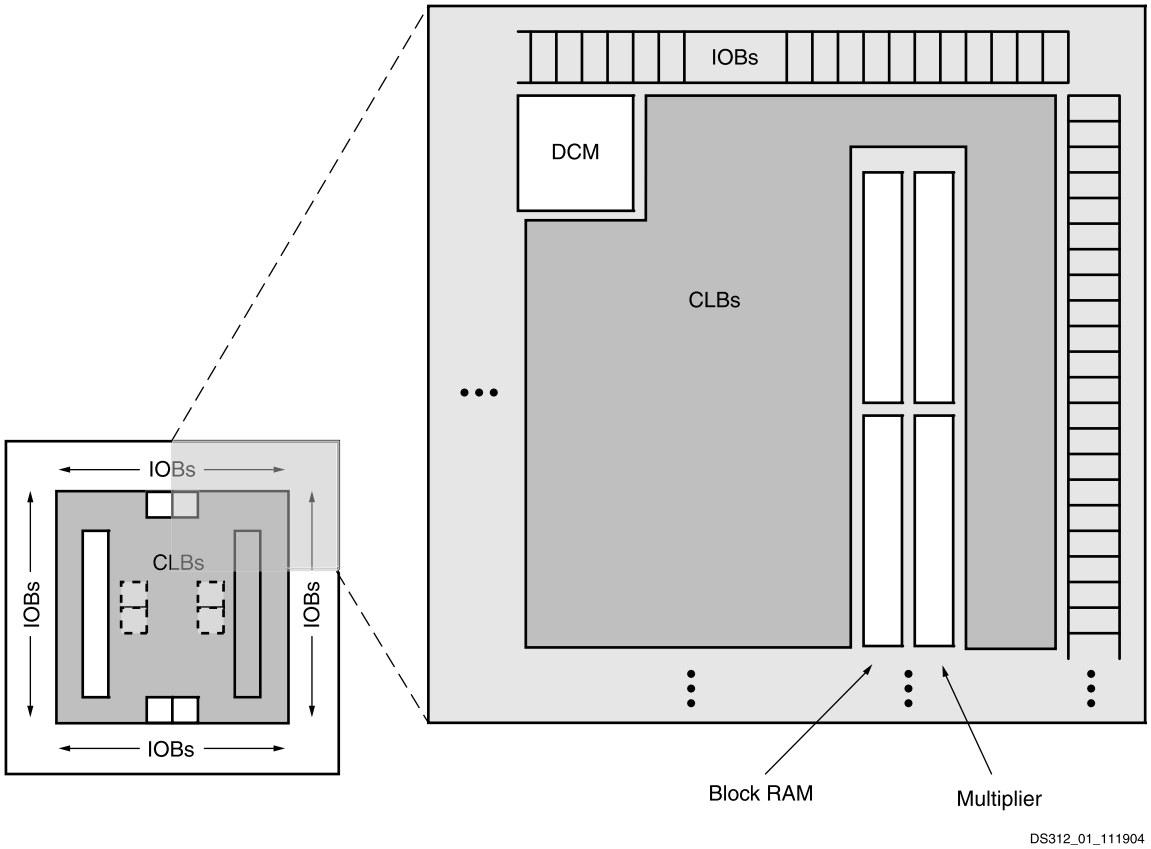



*Figure 1:* **Spartan-3E Family Architecture**

## Configuration

Spartan-3E FPGAs are programmed by loading configuration data into robust, reprogrammable, static CMOS configuration latches (CCLs) that collectively control all functional elements and routing resources. The FPGA's configuration data is stored externally in a PROM or some other non-volatile medium, either on or off the board. After applying power, the configuration data is written to the FPGA using any of seven different modes:

- Master Serial from a Xilinx Platform Flash PROM
- Serial Peripheral Interface (SPI) from an industry-standard SPI serial Flash
- Byte Peripheral Interface (BPI) Up or Down from an industry-standard x8 or x8/x16 parallel NOR Flash
- Slave Serial, typically downloaded from a processor
- Slave Parallel, typically downloaded from a processor
- Boundary Scan (JTAG), typically downloaded from a processor or system tester.

Furthermore, Spartan-3E FPGAs support MultiBoot configuration, allowing two or more FPGA configuration bitstreams to be stored in a single parallel NOR Flash. The FPGA application controls which configuration to load next and when to load it.

## I/O Capabilities

The Spartan-3E FPGA SelectIO interface supports many popular single-ended and differential standards. Table 2 shows the number of user I/Os as well as the number of differential I/O pairs available for each device/package combination.

Spartan-3E FPGAs support the following single-ended standards:

- 3.3V low-voltage TTL (LVTTL)
- Low-voltage CMOS (LVCMOS) at 3.3V, 2.5V, 1.8V, 1.5V, or 1.2V
- 3V PCI at 33 MHz, and in some devices, 66 MHz
- HSTL I and III at 1.8V, commonly used in memory applications
- SSTL I at 1.8V and 2.5V, commonly used for memory applications

Spartan-3E FPGAs support the following differential standards:

- LVDS
- Bus LVDS
- mini-LVDS
- RSDS
- Differential HSTL (1.8V, Types I and III)
- Differential SSTL (2.5V and 1.8V, Type I)
- 2.5V LVPECL inputs

*Table 2:* **Available User I/Os and Differential (Diff) I/O Pairs**

| Package | VQ100 VQG100 | | CP132 CPG132 | | TQ144 TQG144 | | PQ208 PQG208 | | FT256 FTG256 | | FG320 FGG320 | | FG400 FGG400 | | FG484 FGG484 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Footprint Size (mm)** | **16 x 16** | | **8 x 8** | | **22 x 22** | | **30.5 x 30.5** | | **17 x 17** | | **19 x 19** | | **21 x 21** | | **23 x 23** | |
| **Device** | **User** | **Diff** | **User** | **Diff** | **User** | **Diff** | **User** | **Diff** | **User** | **Diff** | **User** | **Diff** | **User** | **Diff** | **User** | **Diff** |
| **XC3S100E** | **66**(2) *9(7)* | **30** *(2)* | **83** *(11)* | **35** *(2)* | **108** *(28)* | **40** *(4)* | - | - | - | - | - | - | - | - | - | - |
| **XC3S250E** | **66** *(7)* | **30** *(2)* | **92** *(7)* | **41** *(2)* | **108** *(28)* | **40** *(4)* | **158** *(32)* | **65** *(5)* | **172** *(40)* | **68** *(8)* | - | - | - | - | - | - |
| **XC3S500E** | **66**(3) *(7)* | **30** *(2)* | **92** *(7)* | **41** *(2)* | - | - | **158** *(32)* | **65** *(5)* | **190** *(41)* | **77** *(8)* | **232** *(56)* | **92** *(12)* | - | - | - | - |
| **XC3S1200E** | - | - | - | - | - | - | - | - | **190** *(40)* | **77** *(8)* | **250** *(56)* | **99** *(12)* | **304** *(72)* | **124** *(20)* | - | - |
| **XC3S1600E** | - | - | - | - | - | - | - | - | - | - | **250** *(56)* | **99** *(12)* | **304** *(72)* | **124** *(20)* | **376** *(82)* | **156** *(21)* |

**Notes:**

1. All Spartan-3E devices provided in the same package are pin-compatible as further described in Module 4, Pinout Descriptions.
2. The number shown in **bold** indicates the maximum number of I/O and input-only pins. The number shown in (*italics*) indicates the number of input-only pins.
3. The XC3S500E is available in the VQG100 Pb-free package and not the standard VQ100. The VQG100 and VQ100 pin-outs are identical and general references to the VQ100 will apply to the XC3S500E.

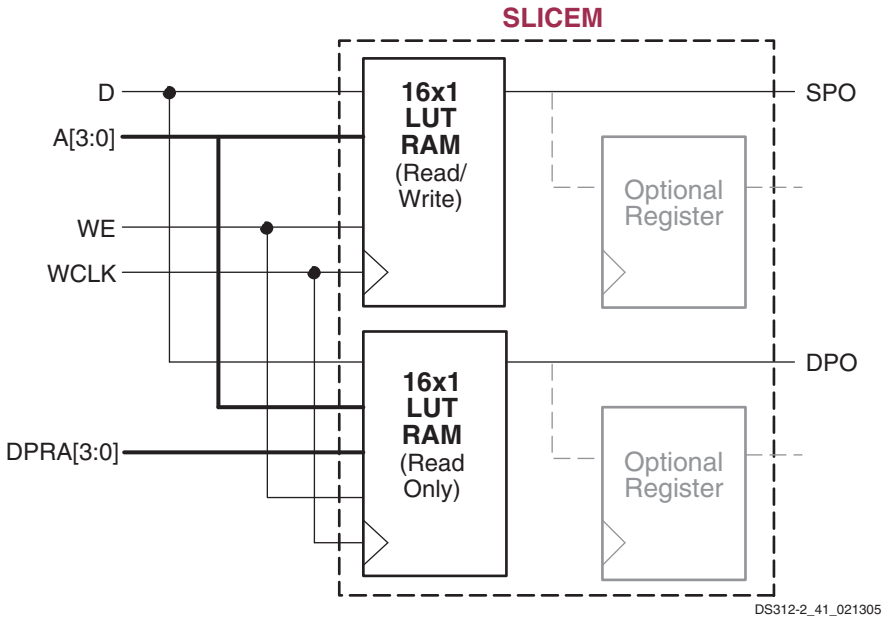



*Figure 26:* **RAM16X1D Dual-Port Usage**

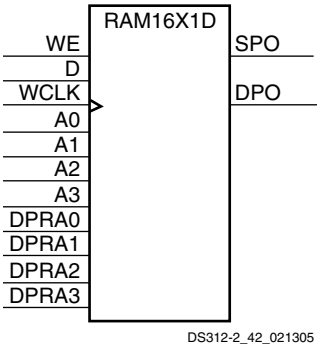



*Figure 27:* **Dual-Port RAM Component**

*Table 18:* **Dual-Port RAM Function**

| Inputs | | | Outputs | |
|---|---|---|---|---|
| **WE (mode)** | **WCLK** | **D** | **SPO** | **DPO** |
| 0 (read) | X | X | data_a | data_d |
| 1 (read) | 0 | X | data_a | data_d |
| 1 (read) | 1 | X | data_a | data_d |
| 1 (write) | ↑ | D | D | data_d |
| 1 (read) | ↓ | X | data_a | data_d |

**Notes:**

1. data_a = word addressed by bits A3-A0.
2. data_d = word addressed by bits DPRA3-DPRA0.

*Table 19:* **Distributed RAM Signals**

| Signal | Description |
|---|---|
| WCLK | The clock is used for synchronous writes. The data and the address input pins have setup times referenced to the WCLK pin. Active on the positive edge by default with built-in programmable polarity. |
| WE | The enable pin affects the write functionality of the port. An inactive Write Enable prevents any writing to memory cells. An active Write Enable causes the clock edge to write the data input signal to the memory location pointed to by the address inputs. Active High by default with built-in programmable polarity. |
| A0, A1, A2, A3 (A4, A5) | The address inputs select the memory cells for read or write. The width of the port determines the required address inputs. |
| D | The data input provides the new data value to be written into the RAM. |
| O, SPO, and DPO | The data output O on single-port RAM or the SPO and DPO outputs on dual-port RAM reflects the contents of the memory cells referenced by the address inputs. Following an active write clock edge, the data out (O or SPO) reflects the newly written data. |

The INIT attribute can be used to preload the memory with data during FPGA configuration. The default initial contents for RAM is all zeros. If the WE is held Low, the element can be considered a ROM. The ROM function is possible even in the SLICEL.

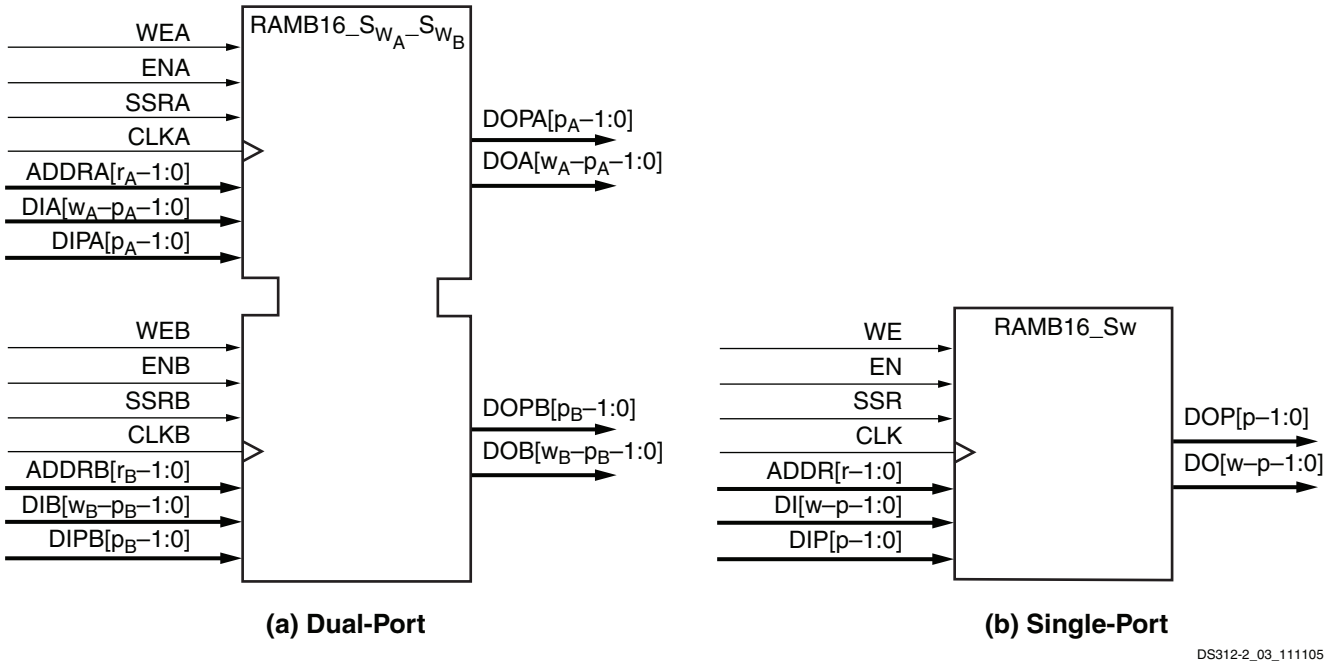
The global write enable signal, GWE, is asserted automatically at the end of device configuration to enable all writable elements. The GWE signal guarantees that the

## Block RAM Port Signal Definitions

Representations of the dual-port primitive RAMB16_S[$w_A$]_S[$w_B$] and the single-port primitive RAMB16_S[w] with their associated signals are shown in Figure 32a and Figure 32b, respectively. These signals are defined in Table 23. The control signals (WE, EN, CLK, and SSR) on the block RAM are active High. However, optional inverters on the control signals change the polarity of the active edge to active Low.

***Design Note***

Whenever a block RAM port is enabled (ENA or ENB = High), all address transitions must meet the data sheet setup and hold times with respect to the port clock (CLKA or CLKB), as shown in Table 103, page 138.This requirement must be met even if the RAM read output is of no interest.




**Notes:**

1. $w_A$ and $w_B$ are integers representing the total data path width (i.e., data bits plus parity bits) at Ports A and B, respectively.
2. $p_A$ and $p_B$ are integers that indicate the number of data path lines serving as parity bits.
3. $r_A$ and $r_B$ are integers representing the address bus width at ports A and B, respectively.
4. The control signals CLK, WE, EN, and SSR on both ports have the option of inverted polarity.

*Figure 32:* **Block RAM Primitives**

## Dedicated Multipliers

For additional information, refer to the “Using Embedded Multipliers” chapter in UG331.

The Spartan-3E devices provide 4 to 36 dedicated multiplier blocks per device. The multipliers are located together with the block RAM in one or two columns depending on device density. See Arrangement of RAM Blocks on Die for details on the location of these blocks and their connectivity.

### Operation

The multiplier blocks primarily perform two’s complement numerical multiplication but can also perform some less obvious applications, such as simple data storage and barrel shifting. Logic slices also implement efficient small multipliers and thereby supplement the dedicated multipliers. The Spartan-3E dedicated multiplier blocks have additional features beyond those provided in Spartan-3 FPGAs.

Each multiplier performs the principle operation P = A × B, where ‘A’ and ‘B’ are 18-bit words in two’s complement form, and ‘P’ is the full-precision 36-bit product, also in two’s complement form. The 18-bit inputs represent values ranging from $-131{,}072_{10}$ to $+131{,}071_{10}$ with a resulting product ranging from $-17{,}179{,}738{,}112_{10}$ to $+17{,}179{,}869{,}184_{10}$.

Implement multipliers with inputs less than 18 bits by sign-extending the inputs (i.e., replicating the most-significant bit). Wider multiplication operations are performed by combining the dedicated multipliers and slice-based logic in any viable combination or by time-sharing a single multiplier. Perform unsigned multiplication by restricting the inputs to the positive range. Tie the most-significant bit Low and represent the unsigned value in the remaining 17 lesser-significant bits.

### Optional Pipeline Registers

As shown in Figure 36, each multiplier block has optional registers on each of the multiplier inputs and the output. The registers are named AREG, BREG, and PREG and can be used in any combination. The clock input is common to all the registers within a block, but each register has an independent clock enable and synchronous reset controls making them ideal for storing data samples and coefficients. When used for pipelining, the registers boost the multiplier clock rate, beneficial for higher performance applications.

Figure 36 illustrates the principle features of the multiplier block.

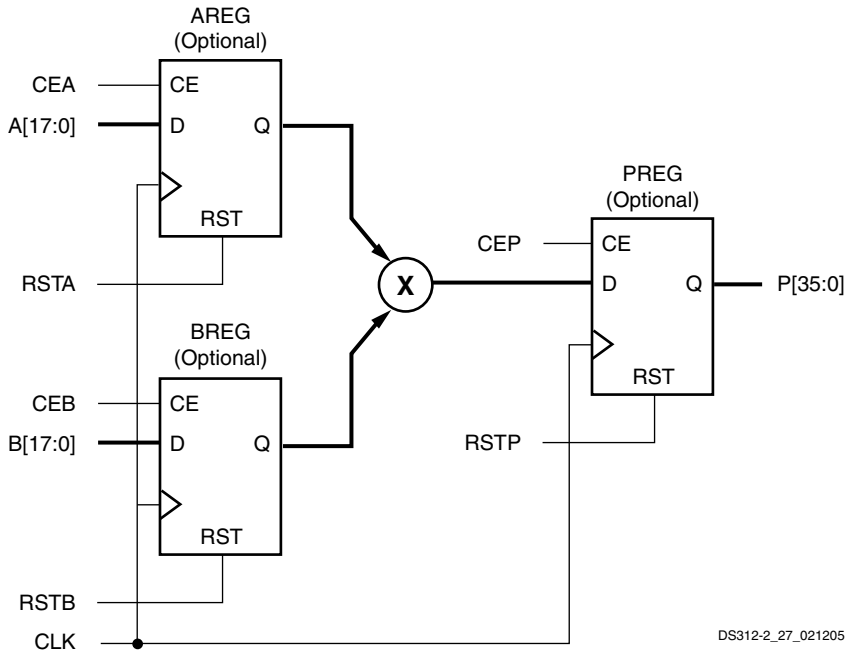



*Figure 36:* **Principle Ports and Functions of Dedicated Multiplier Blocks**

Use the MULT18X18SIO primitive shown in Figure 37 to instantiate a multiplier within a design. Although high-level logic synthesis software usually automatically infers a multiplier, adding the pipeline registers might require the MULT18X18SIO primitive. Connect the appropriate signals to the MULT18X18SIO multiplier ports and set the individual AREG, BREG, and PREG attributes to ‘1’ to insert the associated register, or to 0 to remove it and make the signal path combinatorial.

The CLKFX_DIVIDE is an integer ranging from 1 to 32, inclusive and forms the denominator in Equation 1. For example, if CLKFX_MULTIPLY = 5 and CLKFX_DIVIDE = 3, the frequency of the output clock signal is 5/3 that of the input clock signal. These attributes and their acceptable ranges are described in Table 34.

*Table 34:* **DFS Attributes**

| Attribute | Description | Values |
|---|---|---|
| CLKFX_MULTIPLY | Frequency multiplier constant | Integer from 2 to 32, inclusive |
| CLKFX_DIVIDE | Frequency divisor constant | Integer from 1 to 32, inclusive |

Any combination of integer values can be assigned to the CLKFX_MULTIPLY and CLKFX_DIVIDE attributes, provided that two conditions are met:

1. The two values fall within their corresponding ranges, as specified in Table 34.
2. The $f_{CLKFX}$ output frequency calculated in Equation 1 falls within the DCM's operating frequency specifications (see Table 107 in Module 3).

### DFS With or Without the DLL

Although the CLKIN input is shared with both units, the DFS unit functions with or separately from the DLL unit. Separate from the DLL, the DFS generates an output frequency from the CLKIN frequency according to the respective CLKFX_MULTIPLY and CLKFX_DIVIDE values. Frequency synthesis does not require a feedback loop. Furthermore, without the DLL, the DFS unit supports a broader operating frequency range.

With the DLL, the DFS unit operates as described above, only with the additional benefit of eliminating the clock distribution delay. In this case, a feedback loop from the CLK0 or CLK2X output to the CLKFB input must be present.

When operating with the DLL unit, the DFS's CLKFX and CLKFX180 outputs are phase-aligned with the CLKIN input every CLKFX_DIVIDE cycles of CLKIN and every CLKFX_MULTIPLY cycles of CLKFX. For example, when CLKFX_MULTIPLY = 5 and CLKFX_DIVIDE = 3, the input and output clock edges coincide every three CLKIN input periods, which is equivalent in time to five CLKFX output periods.

Smaller CLKFX_MULTIPLY and CLKFX_DIVIDE values result in faster lock times. Therefore, CLKFX_MULTIPLY and CLKFX_DIVIDE must be factored to reduce their values wherever possible. For example, given CLKFX_MULTIPLY = 9 and CLKFX_DIVIDE = 6, removing a factor of three yields CLKFX_MULTIPLY = 3 and CLKFX_DIVIDE = 2. While both value-pairs result in the multiplication of clock frequency by 3/2, the latter value-pair enables the DLL to lock more quickly.

## Phase Shifter (PS)

The DCM provides two approaches to controlling the phase of a DCM clock output signal relative to the CLKIN signal: First, eight of the nine DCM clock outputs — CLK0, CLK90, CLK180, CLK270, CLK2X, CLK2X180, CLKFX, and CLKFX180 — provide either quadrant or half-period phase shifting of the input clock.

Second, the PS unit provides additional fine phase shift control of all nine DCM outputs. The PS unit accomplishes this by introducing a "fine phase shift" delay ($T_{PS}$) between the CLKFB and CLKIN signals inside the DLL unit. In FIXED phase shift mode, the fine phase shift is specified at design time with a resolution down to $^{1}/_{256}^{th}$ of a CLKIN cycle or one delay step (DCM_DELAY_STEP), whichever is greater. This fine phase shift value is relative to the coarser quadrant or half-period phase shift of the DCM clock output. When used, the PS unit shifts the phase of all nine DCM clock output signals.

### Enabling Phase Shifting and Selecting an Operating Mode

The CLKOUT_PHASE_SHIFT attribute controls the PS unit for the specific DCM instantiation. As described in Table 35, this attribute has three possible values: NONE, FIXED, and VARIABLE. When CLKOUT_PHASE_SHIFT = NONE, the PS unit is disabled and the DCM output clocks are phase-aligned to the CLKIN input via the CLKFB feedback path. Figure 44a shows this case.

The PS unit is enabled when the CLKOUT_PHASE_SHIFT attribute is set to FIXED or VARIABLE modes. These two modes are described in the sections that follow.

*Table 35:* **PS Attributes**

| Attribute | Description | Values |
|---|---|---|
| CLKOUT_PHASE_SHIFT | Disables the PS component or chooses between Fixed Phase and Variable Phase modes. | ***NONE***, FIXED, VARIABLE |
| PHASE_SHIFT | Determines size and direction of initial fine phase shift. | Integers from –255 to +255 |

# 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<br><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 | | | ✓ | | | |

read operations at this time. Spartan-3E FPGAs issue the read command just once. If the SPI Flash is not ready, then the FPGA does not properly configure.

If the 3.3V supply is last in the sequence and does not ramp fast enough, or if the SPI Flash PROM cannot be ready when required by the FPGA, delay the FPGA configuration process by holding either the FPGA's PROG_B input or INIT_B input Low, as highlighted in Figure 54. Release the FPGA when the SPI Flash PROM is ready. For example, a simple R-C delay circuit attached to the INIT_B pin forces the FPGA to wait for a preselected amount of time. Alternately, a Power Good signal from the 3.3V supply or a system reset signal accomplishes the same purpose. Use an open-drain or open-collector output when driving PROG_B or INIT_B.

### SPI Flash PROM Density Requirements

Table 57 shows the smallest usable SPI Flash PROM to program a single Spartan-3E FPGA. Commercially available SPI Flash PROMs range in density from 1 Mbit to 128 Mbits. A multiple-FPGA daisy-chained application requires a SPI Flash PROM large enough to contain the sum of the FPGA file sizes. An application can also use a larger-density SPI Flash PROM to hold additional data beyond just FPGA configuration data. For example, the SPI Flash PROM can also store application code for a MicroBlaze™ RISC processor core integrated in the Spartan-3E FPGA. See Using the SPI Flash Interface after Configuration.

*Table 57:* **Number of Bits to Program a Spartan-3E FPGA and Smallest SPI Flash PROM**

| Device | Number of Configuration Bits | Smallest Usable SPI Flash PROM |
|---|---|---|
| XC3S100E | 581,344 | 1 Mbit |
| XC3S250E | 1,353,728 | 2 Mbit |
| XC3S500E | 2,270,208 | 4 Mbit |
| XC3S1200E | 3,841,184 | 4 Mbit |
| XC3S1600E | 5,969,696 | 8 Mbit |

### CCLK Frequency

In SPI Flash mode, the FPGA's internal oscillator generates the configuration clock frequency. The FPGA provides this clock on its CCLK output pin, driving the PROM's clock input pin. The FPGA starts configuration at its lowest frequency and increases its frequency for the remainder of the configuration process if so specified in the configuration bitstream. The maximum frequency is specified using the ***ConfigRate*** bitstream generator option. The maximum frequency supported by the FPGA configuration logic depends on the timing for the SPI Flash device. Without examining the timing for a specific SPI Flash PROM, use ***ConfigRate*** = 12 or lower. SPI Flash PROMs that support the FAST READ command support higher data rates. Some such PROMs support up to ***ConfigRate*** = 25 and beyond but require careful data sheet analysis. See Serial Peripheral Interface (SPI) Configuration Timing for more detailed timing analysis.

### Using the SPI Flash Interface after Configuration

After the FPGA successfully completes configuration, all of the pins connected to the SPI Flash PROM are available as user-I/O pins.

If not using the SPI Flash PROM after configuration, drive CSO_B High to disable the PROM. The MOSI, DIN, and CCLK pins are then available to the FPGA application.

Because all the interface pins are user I/O after configuration, the FPGA application can continue to use the SPI Flash interface pins to communicate with the SPI Flash PROM, as shown in Figure 56. SPI Flash PROMs offer random-accessible, byte-addressable, read/write, non-volatile storage to the FPGA application.

SPI Flash PROMs are available in densities ranging from 1 Mbit up to 128 Mbits. However, a single Spartan-3E FPGA requires less than 6 Mbits. If desired, use a larger SPI Flash PROM to contain additional non-volatile application data, such as MicroBlaze processor code, or other user data such as serial numbers and Ethernet MAC IDs. In the example shown in Figure 56, the FPGA configures from SPI Flash PROM. Then using FPGA logic after configuration, the FPGA copies MicroBlaze code from SPI Flash into external DDR SDRAM for code execution. Similarly, the FPGA application can store non-volatile application data within the SPI Flash PROM.

The FPGA configuration data is stored starting at location 0. Store any additional data beginning in the next available SPI Flash PROM sector or page. Do not mix configuration data and user data in the same sector or page.

Similarly, the SPI bus can be expanded to additional SPI peripherals. Because SPI is a common industry-standard interface, various SPI-based peripherals are available, such as analog-to-digital (A/D) converters, digital-to-analog (D/A) converters, CAN controllers, and temperature sensors. However, if sufficient I/O pins are available in the application, Xilinx recommends creating a separate SPI bus to control peripherals. Creating a second port reduces the loading on the CCLK and DIN pins, which are crucial for configuration.

The MOSI, DIN, and CCLK pins are common to all SPI peripherals. Connect the select input on each additional SPI peripheral to one of the FPGA user I/O pins. If HSWAP = 0 during configuration, the FPGA holds the select line High. If HSWAP = 1, connect the select line to +3.3V via an external 4.7 kΩ pull-up resistor to avoid spurious read or write operations. After configuration, drive the select line Low to select the desired SPI peripheral.

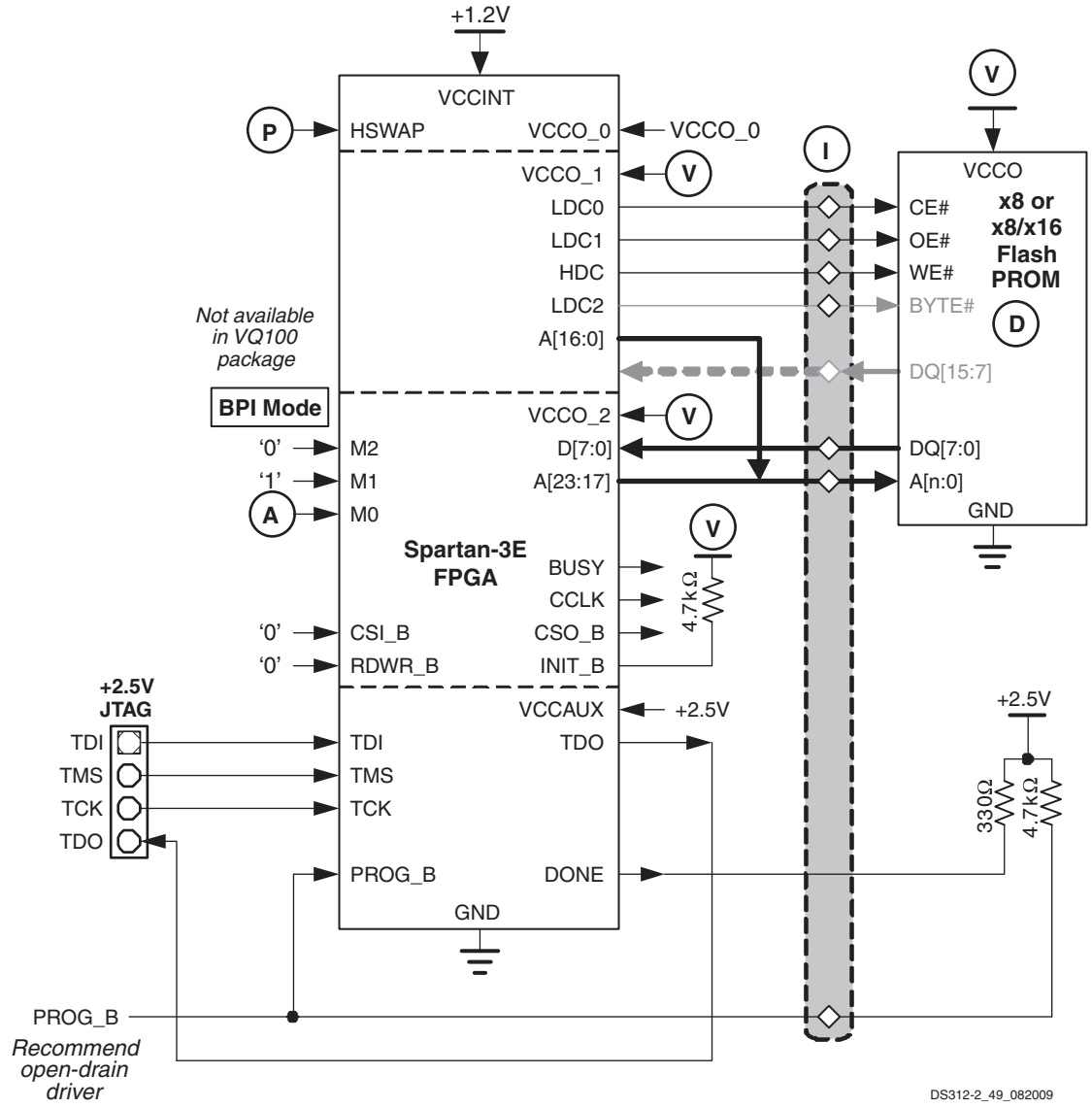



*Figure 58:* **Byte-wide Peripheral Interface (BPI) Mode Configured from Parallel NOR Flash PROMs**

(A) During configuration, the value of the M0 mode pin determines how the FPGA generates addresses, as shown Table 58. When M0 = 0, the FPGA generates addresses starting at 0 and increments the address on every falling CCLK edge. Conversely, when M0 = 1, the FPGA generates addresses starting at `0xFF_FFFF` (all ones) and decrements the address on every falling CCLK edge.

*Table 58:* **BPI Addressing Control**

| M2 | M1 | M0 | Start Address | Addressing |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | Incrementing |
| | | 1 | `0xFF_FFFF` | Decrementing |

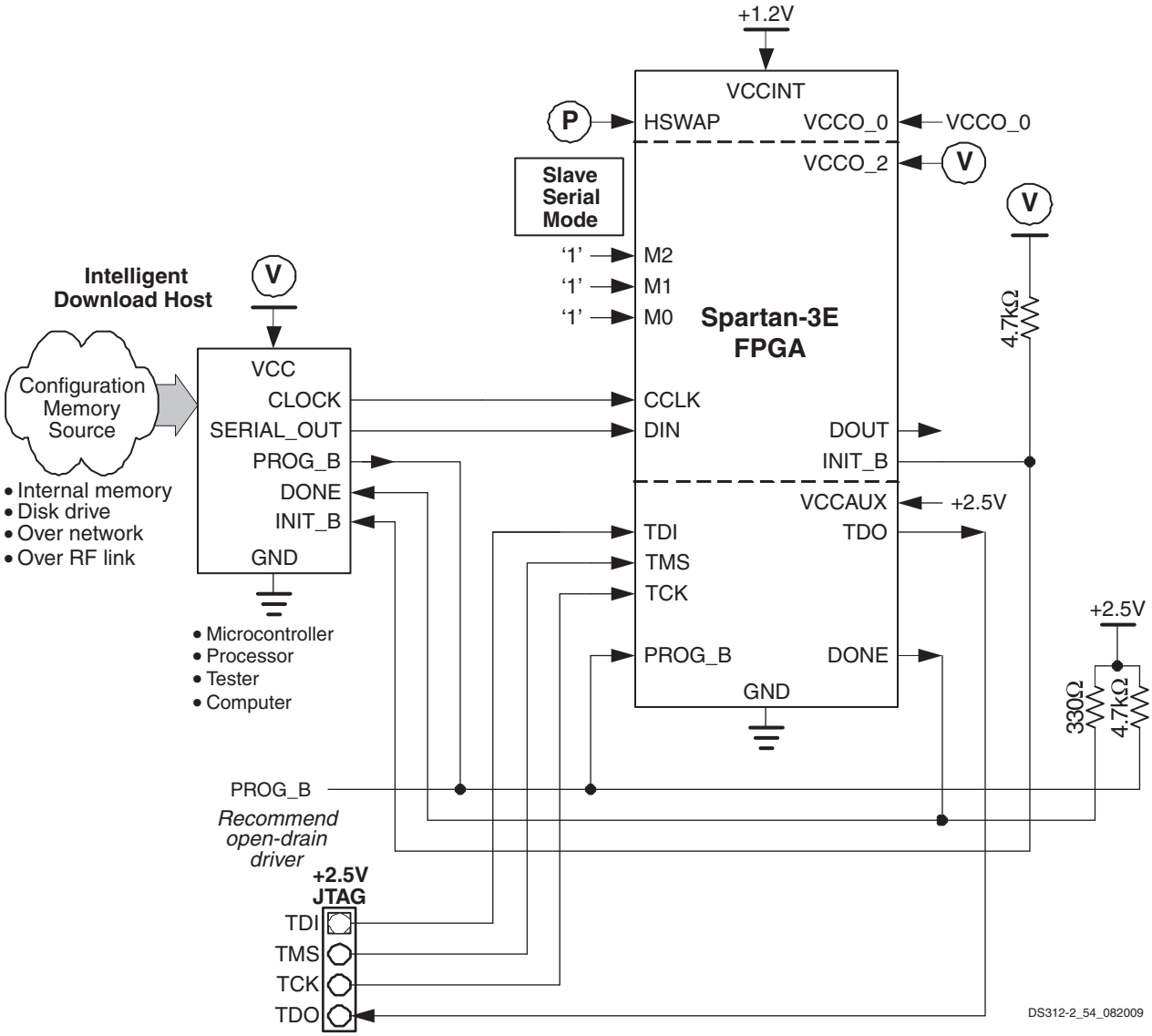



*Figure 63:* **Slave Serial Configuration**

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

## Readback

FPGA configuration data can be read back using either the Slave Parallel or JTAG mode. This function is disabled if the Bitstream Generator ***Security*** option is set to either ***Level1*** or ***Level2***.

Along with the configuration data, it is possible to read back the contents of all registers and distributed RAM.

To synchronously control when register values are captured for readback, use the CAPTURE_SPARTAN3 library primitive, which applies for both Spartan-3 and Spartan-3E FPGA families.

The Readback feature is available in most Spartan-3E FPGA product options, as indicated in Table 68. The Readback feature is not available in the XC3S1200E and XC3S1600E FPGAs when using the -4 speed grade in the Commercial temperature grade. Similarly, block RAM Readback support is not available in the -4 speed grade, Commercial temperature devices. If Readback is required in an XC3S1200E or XC3S1600E FPGA, or if block RAM Readback is required on any Spartan-3E FPGA, upgrade to either the Industrial temperature grade version or the -5 speed grade.

The Xilinx iMPACT programming software uses the Readback feature for its optional Verify and Readback operations. The Xilinx ChipScope™ software presently does not use Readback but may in future updates.

*Table 68:* **Readback Support in Spartan-3E FPGAs**

| Temperature Range | Commercial | | Industrial |
|---|---|---|---|
| **Speed Grade** | **-4** | **-5** | **-4** |
| **Block RAM Readback** | | | |
| All Spartan-3E FPGAs | No | Yes | Yes |
| **General Readback (registers, distributed RAM)** | | | |
| XC3S100E | Yes | Yes | Yes |
| XC3S250E | Yes | Yes | Yes |
| XC3S500E | Yes | Yes | Yes |
| XC3S1200E | No | Yes | Yes |
| XC3S1600E | No | Yes | Yes |

## Single-Ended I/O Standards

*Table 80:* **Recommended Operating Conditions for User I/Os Using Single-Ended Standards**

| IOSTANDARD Attribute | $V_{CCO}$ for Drivers[2] | | | $V_{REF}$ | | | $V_{IL}$ | $V_{IH}$[3] |
|---|---|---|---|---|---|---|---|---|
| | Min (V) | Nom (V) | Max (V) | Min (V) | Nom (V) | Max (V) | Max (V) | Min (V) |
| LVTTL | 3.0 | 3.3 | 3.465 | $V_{REF}$ is not used for these I/O standards | | | 0.8 | 2.0 |
| LVCMOS33[4] | 3.0 | 3.3 | 3.465 | | | | 0.8 | 2.0 |
| LVCMOS25[4,5] | 2.3 | 2.5 | 2.7 | | | | 0.7 | 1.7 |
| LVCMOS18 | 1.65 | 1.8 | 1.95 | | | | 0.4 | 0.8 |
| LVCMOS15 | 1.4 | 1.5 | 1.6 | | | | 0.4 | 0.8 |
| LVCMOS12 | 1.1 | 1.2 | 1.3 | | | | 0.4 | 0.7 |
| PCI33_3[6] | 3.0 | 3.3 | 3.465 | | | | 0.3 • $V_{CCO}$ | 0.5 • $V_{CCO}$ |
| PCI66_3[6] | 3.0 | 3.3 | 3.465 | | | | 0.3 • $V_{CCO}$ | 0.5 • $V_{CCO}$ |
| HSTL_I_18 | 1.7 | 1.8 | 1.9 | 0.8 | 0.9 | 1.1 | $V_{REF}$ - 0.1 | $V_{REF}$ + 0.1 |
| HSTL_III_18 | 1.7 | 1.8 | 1.9 | - | 1.1 | - | $V_{REF}$ - 0.1 | $V_{REF}$ + 0.1 |
| SSTL18_I | 1.7 | 1.8 | 1.9 | 0.833 | 0.900 | 0.969 | $V_{REF}$ - 0.125 | $V_{REF}$ + 0.125 |
| SSTL2_I | 2.3 | 2.5 | 2.7 | 1.15 | 1.25 | 1.35 | $V_{REF}$ - 0.125 | $V_{REF}$ + 0.125 |

**Notes:**

1. Descriptions of the symbols used in this table are as follows:
   $V_{CCO}$ – the supply voltage for output drivers
   $V_{REF}$ – the reference voltage for setting the input switching threshold
   $V_{IL}$ – the input voltage that indicates a Low logic level
   $V_{IH}$ – the input voltage that indicates a High logic level
2. The $V_{CCO}$ rails supply only output drivers, not input circuits.
3. For device operation, the maximum signal voltage ($V_{IH}$ max) may be as high as $V_{IN}$ max. See Table 73.
4. There is approximately 100 mV of hysteresis on inputs using LVCMOS33 and LVCMOS25 I/O standards.
5. All Dedicated pins (PROG_B, DONE, TCK, TDI, TDO, and TMS) use the LVCMOS25 standard and draw power from the $V_{CCAUX}$ rail (2.5V). The Dual-Purpose configuration pins use the LVCMOS standard before the User mode. When using these pins as part of a standard 2.5V configuration interface, apply 2.5V to the $V_{CCO}$ lines of Banks 0, 1, and 2 at power-on as well as throughout configuration.
6. For information on PCI IP solutions, see www.xilinx.com/pci. The PCIX IOSTANDARD is available and has equivalent characteristics but no PCI-X IP is supported.

*Table 81:* **DC Characteristics of User I/Os Using Single-Ended Standards**

| IOSTANDARD Attribute | | Test Conditions | | Logic Level Characteristics | |
|---|---|---|---|---|---|
| | | $I_{OL}$ (mA) | $I_{OH}$ (mA) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) |
| LVTTL(3) | 2 | 2 | –2 | 0.4 | 2.4 |
| | 4 | 4 | –4 | | |
| | 6 | 6 | –6 | | |
| | 8 | 8 | –8 | | |
| | 12 | 12 | –12 | | |
| | 16 | 16 | –16 | | |
| LVCMOS33(3) | 2 | 2 | –2 | 0.4 | $V_{CCO}$ – 0.4 |
| | 4 | 4 | –4 | | |
| | 6 | 6 | –6 | | |
| | 8 | 8 | –8 | | |
| | 12 | 12 | –12 | | |
| | 16 | 16 | –16 | | |
| LVCMOS25(3) | 2 | 2 | –2 | 0.4 | $V_{CCO}$ – 0.4 |
| | 4 | 4 | –4 | | |
| | 6 | 6 | –6 | | |
| | 8 | 8 | –8 | | |
| | 12 | 12 | –12 | | |
| LVCMOS18(3) | 2 | 2 | –2 | 0.4 | $V_{CCO}$ – 0.4 |
| | 4 | 4 | –4 | | |
| | 6 | 6 | –6 | | |
| | 8 | 8 | –8 | | |
| LVCMOS15(3) | 2 | 2 | –2 | 0.4 | $V_{CCO}$ – 0.4 |
| | 4 | 4 | –4 | | |
| | 6 | 6 | –6 | | |
| LVCMOS12(3) | 2 | 2 | –2 | 0.4 | $V_{CCO}$ – 0.4 |
| PCI33_3(4) | | 1.5 | –0.5 | 10% $V_{CCO}$ | 90% $V_{CCO}$ |
| PCI66_3(4) | | 1.5 | –0.5 | 10% $V_{CCO}$ | 90% $V_{CCO}$ |
| HSTL_I_18 | | 8 | –8 | 0.4 | $V_{CCO}$ – 0.4 |
| HSTL_III_18 | | 24 | –8 | 0.4 | $V_{CCO}$ – 0.4 |
| SSTL18_I | | 6.7 | –6.7 | $V_{TT}$ – 0.475 | $V_{TT}$ + 0.475 |

*Table 81:* **DC Characteristics of User I/Os Using Single-Ended Standards *(Cont'd)***

| IOSTANDARD Attribute | Test Conditions | | Logic Level Characteristics | |
|---|---|---|---|---|
| | $I_{OL}$ (mA) | $I_{OH}$ (mA) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) |
| SSTL2_I | 8.1 | –8.1 | $V_{TT}$ – 0.61 | $V_{TT}$ + 0.61 |

**Notes:**

1. The numbers in this table are based on the conditions set forth in Table 77 and Table 80.
2. Descriptions of the symbols used in this table are as follows:
   $I_{OL}$ – the output current condition under which VOL is tested
   $I_{OH}$ – the output current condition under which VOH is tested
   $V_{OL}$ – the output voltage that indicates a Low logic level
   $V_{OH}$ – the output voltage that indicates a High logic level
   $V_{CCO}$ – the supply voltage for output drivers
   $V_{TT}$ – the voltage applied to a resistor termination
3. For the LVCMOS and LVTTL standards: the same $V_{OL}$ and $V_{OH}$ limits apply for both the Fast and Slow slew attributes.
4. Tested according to the relevant PCI specifications. For information on PCI IP solutions, see www.xilinx.com/pci. The PCIX IOSTANDARD is available and has equivalent characteristics but no PCI-X IP is supported.

*Table 93:* **Timing for the IOB Three-State Path**

| Symbol | Description | Conditions | Device | Speed Grade | | Units |
|---|---|---|---|---|---|---|
| | | | | -5 | -4 | |
| | | | | Max | Max | |
| **Synchronous Output Enable/Disable Times** | | | | | | |
| $T_{IOCKHZ}$ | Time from the active transition at the OTCLK input of the Three-state Flip-Flop (TFF) to when the Output pin enters the high-impedance state | LVCMOS25, 12 mA output drive, Fast slew rate | All | 1.49 | 1.71 | ns |
| $T_{IOCKON}$(2) | Time from the active transition at TFF's OTCLK input to when the Output pin drives valid data | | All | 2.70 | 3.10 | ns |
| **Asynchronous Output Enable/Disable Times** | | | | | | |
| $T_{GTS}$ | Time from asserting the Global Three State (GTS) input on the STARTUP_SPARTAN3E primitive to when the Output pin enters the high-impedance state | LVCMOS25, 12 mA output drive, Fast slew rate | All | 8.52 | 9.79 | ns |
| **Set/Reset Times** | | | | | | |
| $T_{IOSRHZ}$ | Time from asserting TFF's SR input to when the Output pin enters a high-impedance state | LVCMOS25, 12 mA output drive, Fast slew rate | All | 2.11 | 2.43 | ns |
| $T_{IOSRON}$(2) | Time from asserting TFF's SR input at TFF to when the Output pin drives valid data | | All | 3.32 | 3.82 | ns |

**Notes:**

1. The numbers in this table are tested using the methodology presented in Table 95 and are based on the operating conditions set forth in Table 77 and Table 80.
2. This time requires adjustment whenever a signal standard other than LVCMOS25 with 12 mA drive and Fast slew rate is assigned to the data Output. When this is true, *add* the appropriate Output adjustment from Table 94.
3. For minimum delays use the values reported by the Timing Analyzer.

*Table 94:* **Output Timing Adjustments for IOB**

| Convert Output Time from LVCMOS25 with 12mA Drive and Fast Slew Rate to the Following Signal Standard (IOSTANDARD) | | | Add the Adjustment Below | | Units |
|---|---|---|---|---|---|
| | | | Speed Grade | | |
| | | | -5 | -4 | |
| **Single-Ended Standards** | | | | | |
| LVTTL | Slow | 2 mA | 5.20 | 5.41 | ns |
| | | 4 mA | 2.32 | 2.41 | ns |
| | | 6 mA | 1.83 | 1.90 | ns |
| | | 8 mA | 0.64 | 0.67 | ns |
| | | 12 mA | 0.68 | 0.70 | ns |
| | | 16 mA | 0.41 | 0.43 | ns |
| | Fast | 2 mA | 4.80 | 5.00 | ns |
| | | 4 mA | 1.88 | 1.96 | ns |
| | | 6 mA | 1.39 | 1.45 | ns |
| | | 8 mA | 0.32 | 0.34 | ns |
| | | 12 mA | 0.28 | 0.30 | ns |
| | | 16 mA | 0.28 | 0.30 | ns |
| LVCMOS33 | Slow | 2 mA | 5.08 | 5.29 | ns |
| | | 4 mA | 1.82 | 1.89 | ns |
| | | 6 mA | 1.00 | 1.04 | ns |
| | | 8 mA | 0.66 | 0.69 | ns |
| | | 12 mA | 0.40 | 0.42 | ns |
| | | 16 mA | 0.41 | 0.43 | ns |
| | Fast | 2 mA | 4.68 | 4.87 | ns |
| | | 4 mA | 1.46 | 1.52 | ns |
| | | 6 mA | 0.38 | 0.39 | ns |
| | | 8 mA | 0.33 | 0.34 | ns |
| | | 12 mA | 0.28 | 0.30 | ns |
| | | 16 mA | 0.28 | 0.30 | ns |
| LVCMOS25 | Slow | 2 mA | 4.04 | 4.21 | ns |
| | | 4 mA | 2.17 | 2.26 | ns |
| | | 6 mA | 1.46 | 1.52 | ns |
| | | 8 mA | 1.04 | 1.08 | ns |
| | | 12 mA | 0.65 | 0.68 | ns |
| | Fast | 2 mA | 3.53 | 3.67 | ns |
| | | 4 mA | 1.65 | 1.72 | ns |
| | | 6 mA | 0.44 | 0.46 | ns |
| | | 8 mA | 0.20 | 0.21 | ns |
| | | 12 mA | 0 | 0 | ns |

*Table 94:* **Output Timing Adjustments for IOB *(Cont'd)***

| Convert Output Time from LVCMOS25 with 12mA Drive and Fast Slew Rate to the Following Signal Standard (IOSTANDARD) | | | Add the Adjustment Below | | Units |
|---|---|---|---|---|---|
| | | | Speed Grade | | |
| | | | -5 | -4 | |
| LVCMOS18 | Slow | 2 mA | 5.03 | 5.24 | ns |
| | | 4 mA | 3.08 | 3.21 | ns |
| | | 6 mA | 2.39 | 2.49 | ns |
| | | 8 mA | 1.83 | 1.90 | ns |
| | Fast | 2 mA | 3.98 | 4.15 | ns |
| | | 4 mA | 2.04 | 2.13 | ns |
| | | 6 mA | 1.09 | 1.14 | ns |
| | | 8 mA | 0.72 | 0.75 | ns |
| LVCMOS15 | Slow | 2 mA | 4.49 | 4.68 | ns |
| | | 4 mA | 3.81 | 3.97 | ns |
| | | 6 mA | 2.99 | 3.11 | ns |
| | Fast | 2 mA | 3.25 | 3.38 | ns |
| | | 4 mA | 2.59 | 2.70 | ns |
| | | 6 mA | 1.47 | 1.53 | ns |
| LVCMOS12 | Slow | 2 mA | 6.36 | 6.63 | ns |
| | Fast | 2 mA | 4.26 | 4.44 | ns |
| HSTL_I_18 | | | 0.33 | 0.34 | ns |
| HSTL_III_18 | | | 0.53 | 0.55 | ns |
| PCI33_3 | | | 0.44 | 0.46 | ns |
| PCI66_3 | | | 0.44 | 0.46 | ns |
| SSTL18_I | | | 0.24 | 0.25 | ns |
| SSTL2_I | | | –0.20 | –0.20 | ns |
| **Differential Standards** | | | | | |
| LVDS_25 | | | –0.55 | –0.55 | ns |
| BLVDS_25 | | | 0.04 | 0.04 | ns |
| MINI_LVDS_25 | | | –0.56 | –0.56 | ns |
| LVPECL_25 | | | Input Only | | ns |
| RSDS_25 | | | –0.48 | –0.48 | ns |
| DIFF_HSTL_I_18 | | | 0.42 | 0.42 | ns |
| DIFF_HSTL_III_18 | | | 0.53 | 0.55 | ns |
| DIFF_SSTL18_I | | | 0.40 | 0.40 | ns |
| DIFF_SSTL2_I | | | 0.44 | 0.44 | ns |

**Notes:**

1. The numbers in this table are tested using the methodology presented in Table 95 and are based on the operating conditions set forth in Table 77, Table 80, and Table 82.
2. These adjustments are used to convert output- and three-state-path times originally specified for the LVCMOS25 standard with 12 mA drive and Fast slew rate to times that correspond to other signal standards. Do not adjust times that measure when outputs go into a high-impedance state.

*Table 105:* **Switching Characteristics for the DLL**

| Symbol | Description | | Device | Speed Grade | | | | Units |
|---|---|---|---|---|---|---|---|---|
| | | | | -5 | | -4 | | |
| | | | | Min | Max | Min | Max | |
| **Output Frequency Ranges** | | | | | | | | |
| CLKOUT_FREQ_CLK0 | Frequency for the CLK0 and CLK180 outputs | Stepping 0 | XC3S100E XC3S250E XC3S500E XC3S1600E | N/A | N/A | 5 | 90 | MHz |
| | | | XC3S1200E | | | | 200 | MHz |
| | | Stepping 1 | All | 5 | 275 | | 240 | MHz |
| CLKOUT_FREQ_CLK90 | Frequency for the CLK90 and CLK270 outputs | Stepping 0 | XC3S100E XC3S250E XC3S500E XC3S1600E | N/A | N/A | 5 | 90 | MHz |
| | | | XC3S1200E | | | | 167 | MHz |
| | | Stepping 1 | All | 5 | 200 | | 200 | MHz |
| CLKOUT_FREQ_2X | Frequency for the CLK2X and CLK2X180 outputs | Stepping 0 | XC3S100E XC3S250E XC3S500E XC3S1600E | N/A | N/A | 10 | 180 | MHz |
| | | | XC3S1200E | | | | 311 | MHz |
| | | Stepping 1 | All | 10 | 333 | | 311 | MHz |
| CLKOUT_FREQ_DV | Frequency for the CLKDV output | Stepping 0 | XC3S100E XC3S250E XC3S500E XC3S1600E | N/A | N/A | 0.3125 | 60 | MHz |
| | | | XC3S1200E | | | | 133 | MHz |
| | | Stepping 1 | All | 0.3125 | 183 | | 160 | MHz |
| **Output Clock Jitter**(2,3,4) | | | | | | | | |
| CLKOUT_PER_JITT_0 | Period jitter at the CLK0 output | | All | - | ±100 | - | ±100 | ps |
| CLKOUT_PER_JITT_90 | Period jitter at the CLK90 output | | | - | ±150 | - | ±150 | ps |
| CLKOUT_PER_JITT_180 | Period jitter at the CLK180 output | | | - | ±150 | - | ±150 | ps |
| CLKOUT_PER_JITT_270 | Period jitter at the CLK270 output | | | - | ±150 | - | ±150 | ps |
| CLKOUT_PER_JITT_2X | Period jitter at the CLK2X and CLK2X180 outputs | | | - | ±[1% of CLKIN period + 150] | - | ±[1% of CLKIN period + 150] | ps |
| CLKOUT_PER_JITT_DV1 | Period jitter at the CLKDV output when performing integer division | | | - | ±150 | - | ±150 | ps |
| CLKOUT_PER_JITT_DV2 | Period jitter at the CLKDV output when performing non-integer division | | | - | ±[1% of CLKIN period + 200] | - | ±[1% of CLKIN period + 200] | ps |
| **Duty Cycle**(4) | | | | | | | | |
| CLKOUT_DUTY_CYCLE_DLL | Duty cycle variation for the CLK0, CLK90, CLK180, CLK270, CLK2X, CLK2X180, and CLKDV outputs, including the BUFGMUX and clock tree duty-cycle distortion | | All | - | ±[1% of CLKIN period + 400] | - | ±[1% of CLKIN period + 400] | ps |

## CP132: 132-ball Chip-scale Package

The XC3S100E, XC3S250E and the XC3S500E FPGAs are available in the 132-ball chip-scale package, CP132. The devices share a common footprint for this package as shown in Table 133 and Figure 81.

Table 133 lists all the CP132 package pins. They are sorted by bank number and then by pin name. Pins that form a differential I/O pair appear together in the table. The table also shows the pin number for each pin and the pin type, as defined earlier.

Physically, the D14 and K2 balls on the XC3S100E and XC3S250E FPGAs are not connected but should be connected to VCCINT to maintain density migration compatibility.

Similarly, the A4, C1, and P10 balls on the XC3S100E FPGA are not connected but should be connected to GND to maintain density migration compatibility.

The XC3S100E FPGA has four fewer BPI address pins, A[19:0], whereas the XC3S250E and XC3S500E support A[23:0].

An electronic version of this package pinout table and footprint diagram is available for download from the Xilinx web site at:

http://www.xilinx.com/support/documentation/data_sheets/s3e_pin.zip

### Pinout Table

*Table 133:* **CP132 Package Pinout**

| Bank | XC3S100E Pin Name | XC3S250E XC3S500E Pin Name | CP132 Ball | Type |
|---|---|---|---|---|
| 0 | IO_L01N_0 | IO_L01N_0 | C12 | I/O |
| 0 | IO_L01P_0 | IO_L01P_0 | A13 | I/O |
| 0 | N.C. (◆) | IO_L02N_0 | A12 | ***100E:*** N.C. ***Others:*** I/O |
| 0 | N.C. (◆) | IO_L02P_0 | B12 | ***100E:*** N.C. ***Others:*** I/O |
| 0 | N.C. (◆) | IO_L03N_0/VREF_0 | B11 | ***100E:*** N.C. ***Others:*** VREF (I/O) |
| 0 | IP | IO_L03P_0 | C11 | ***100E:*** INPUT ***Others:*** I/O |
| 0 | IO_L04N_0/GCLK5 | IO_L04N_0/GCLK5 | C9 | GCLK |
| 0 | IO_L04P_0/GCLK4 | IO_L04P_0/GCLK4 | A10 | GCLK |
| 0 | IO_L05N_0/GCLK7 | IO_L05N_0/GCLK7 | A9 | GCLK |
| 0 | IO_L05P_0/GCLK6 | IO_L05P_0/GCLK6 | B9 | GCLK |
| 0 | IO_L07N_0/GCLK11 | IO_L07N_0/GCLK11 | B7 | GCLK |
| 0 | IO_L07P_0/GCLK10 | IO_L07P_0/GCLK10 | A7 | GCLK |
| 0 | IO_L08N_0/VREF_0 | IO_L08N_0/VREF_0 | C6 | VREF |
| 0 | IO_L08P_0 | IO_L08P_0 | B6 | I/O |
| 0 | IO_L09N_0 | IO_L09N_0 | C5 | I/O |
| 0 | IO_L09P_0 | IO_L09P_0 | B5 | I/O |
| 0 | N.C. (◆) | IO_L10N_0 | C4 | ***100E:*** N.C. ***Others:*** I/O |
| 0 | IP | IO_L10P_0 | B4 | ***100E:*** INPUT ***Others:*** I/O |
| 0 | IO_L11N_0/HSWAP | IO_L11N_0/HSWAP | B3 | DUAL |
| 0 | IO_L11P_0 | IO_L11P_0 | A3 | I/O |
| 0 | IP_L06N_0/GCLK9 | IP_L06N_0/GCLK9 | C8 | GCLK |
| 0 | IP_L06P_0/GCLK8 | IP_L06P_0/GCLK8 | B8 | GCLK |
| 0 | VCCO_0 | VCCO_0 | A6 | VCCO |

## Footprint Migration Differences

Table 136 summarizes any footprint and functionality differences between the XC3S100E, the XC3S250E, and the XC3S500E FPGAs that may affect easy migration between devices in the CP132 package. There are 14 such balls. All other pins not listed in Table 136 unconditionally migrate between Spartan-3E devices available in the CP132 package.

The XC3S100E is duplicated on both the left and right sides of the table to show migrations to and from the XC3S250E and the XC3S500E. The arrows indicate the direction for easy migration. A double-ended arrow (←→) indicates that the two pins have identical functionality. A left-facing arrow (←) indicates that the pin on the device on the right unconditionally migrates to the pin on the device on the left. It may be possible to migrate the opposite direction depending on the I/O configuration. For example, an I/O pin (Type = I/O) can migrate to an input-only pin (Type = INPUT) if the I/O pin is configured as an input.

The XC3S100E FPGA in the CP132 package has four fewer BPI-mode address lines than the XC3S250E and XC3S500E.

*Table 136:* **CP132 Footprint Migration Differences**

| CP132 Ball | Bank | XC3S100E Type | Migration | XC3S250E Type | Migration | XC3S500E Type | Migration | XC3S100E Type |
|---|---|---|---|---|---|---|---|---|
| A12 | 0 | N.C. | → | I/O | ←→ | I/O | ← | N.C. |
| B4 | 0 | INPUT | → | I/O | ←→ | I/O | ← | INPUT |
| B11 | 0 | N.C. | → | I/O | ←→ | I/O | ← | N.C. |
| B12 | 0 | N.C. | → | I/O | ←→ | I/O | ← | N.C. |
| C4 | 0 | N.C. | → | I/O | ←→ | I/O | ← | N.C. |
| C11 | 0 | INPUT | → | I/O | ←→ | I/O | ← | INPUT |
| D1 | 3 | N.C. | → | I/O | ←→ | I/O | ← | N.C. |
| D2 | 3 | I/O | → | I/O (Diff) | ←→ | I/O (Diff) | ← | I/O |
| K3 | 3 | VREF(INPUT) | → | VREF(I/O) | ←→ | VREF(I/O) | ← | VREF(INPUT) |
| M9 | 2 | N.C. | → | DUAL | ←→ | DUAL | ← | N.C. |
| M10 | 2 | N.C. | → | DUAL | ←→ | DUAL | ← | N.C. |
| N9 | 2 | N.C. | → | DUAL | ←→ | DUAL | ← | N.C. |
| N10 | 2 | N.C. | → | DUAL | ←→ | DUAL | ← | N.C. |
| P11 | 2 | VREF(INPUT) | → | VREF(I/O) | ←→ | VREF(I/O) | ← | VREF(INPUT) |
| **DIFFERENCES** | | | **14** | | **0** | | **14** | |

Legend:

←→ This pin is identical on the device on the left and the right.

→ This pin can unconditionally migrate from the device on the left to the device on the right. Migration in the other direction may be possible depending on how the pin is configured for the device on the right.

← This pin can unconditionally migrate from the device on the right to the device on the left. Migration in the other direction may be possible depending on how the pin is configured for the device on the left.

## PQ208 Footprint (Left)

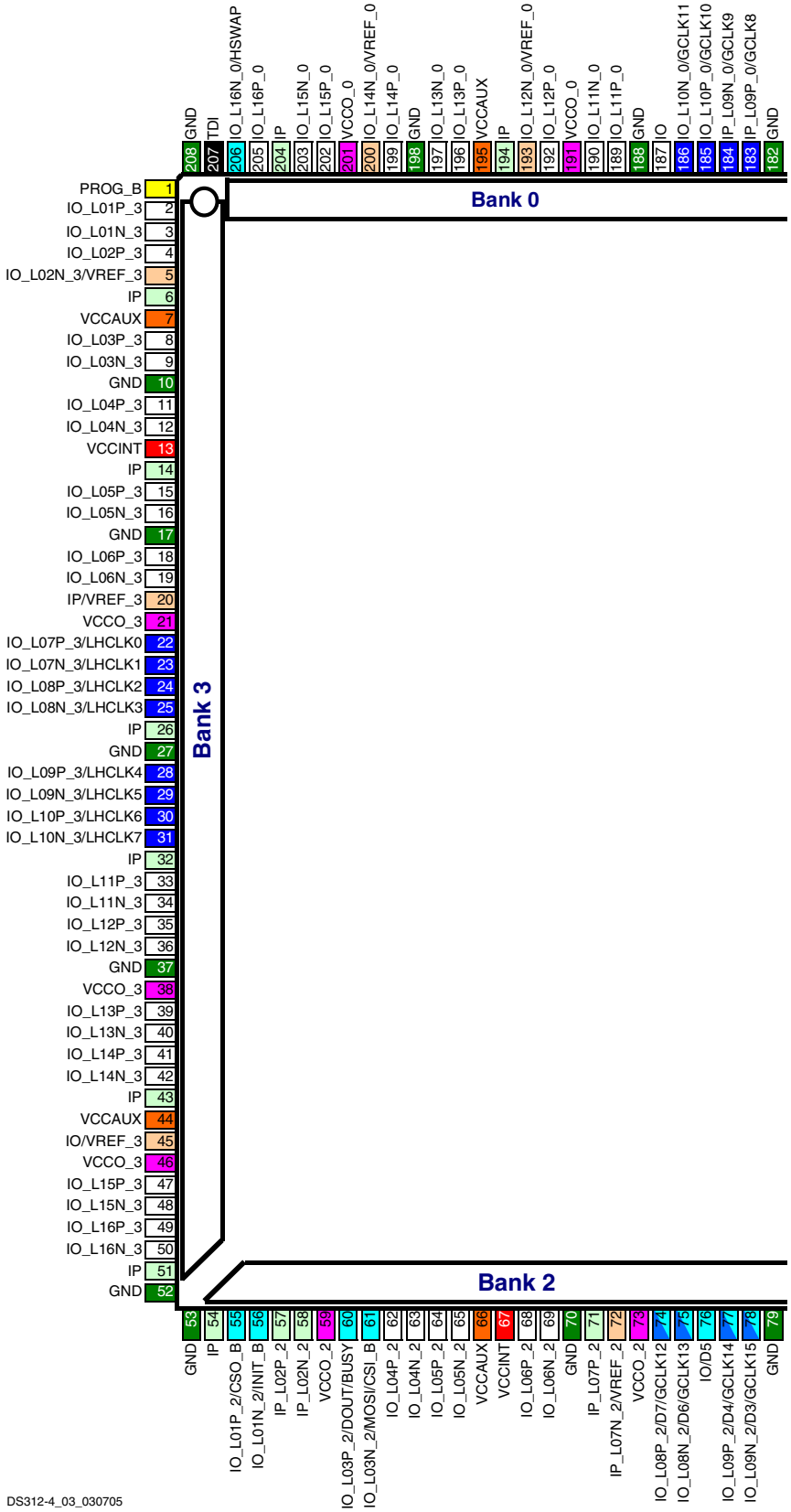



*Figure 83:* **PQ208 Footprint (Left)**

## FG320: 320-ball Fine-pitch Ball Grid Array

The 320-ball fine-pitch ball grid array package, FG320, supports three different Spartan-3E FPGAs, including the XC3S500E, the XC3S1200E, and the XC3S1600E, as shown in Table 148 and Figure 86.

The FG320 package is an 18 x 18 array of solder balls minus the four center balls.

Table 148 lists all the package pins. They are sorted by bank number and then by pin name of the largest device. Pins that form a differential I/O pair appear together in the table. The table also shows the pin number for each pin and the pin type, as defined earlier.

The highlighted rows indicate pinout differences between the XC3S500E, the XC3S1200E, and the XC3S1600E FPGAs. The XC3S500E has 18 unconnected balls, indicated as N.C. (No Connection) in Table 148 and with the black diamond character (◆) in Table 148 and Figure 86.

If the table row is highlighted in tan, then this is an instance where an unconnected pin on the XC3S500E FPGA maps to a VREF pin on the XC3S1200E and XC3S1600E FPGA. If the FPGA application uses an I/O standard that requires a VREF voltage reference, connect the highlighted pin to the VREF voltage supply, even though this does not actually connect to the XC3S500E FPGA. This VREF connection on the board allows future migration to the larger devices without modifying the printed-circuit board.

All other balls have nearly identical functionality on all three devices. Table 147 summarizes the Spartan-3E footprint migration differences for the FG320 package.

An electronic version of this package pinout table and footprint diagram is available for download from the Xilinx web site at:

http://www.xilinx.com/support/documentation/data_sheets/s3e_pin.zip

### Pinout Table

*Table 148:* **FG320 Package Pinout**

| Bank | XC3S500E Pin Name | XC3S1200E Pin Name | XC3S1600E Pin Name | FG320 Ball | Type |
|---|---|---|---|---|---|
| 0 | IP | IO | IO | A7 | **500E:** INPUT<br>**1200E:** I/O<br>**1600E:** I/O |
| 0 | IO | IO | IO | A8 | I/O |
| 0 | IO | IO | IO | A11 | I/O |
| 0 | N.C. (◆) | IO | IO | A12 | **500E:** N.C.<br>**1200E:** I/O<br>**1600E:** I/O |
| 0 | IO | IO | IO | C4 | I/O |
| 0 | IP | IO | IO | D13 | **500E:** INPUT<br>**1200E:** I/O<br>**1600E:** I/O |
| 0 | IO | IO | IO | E13 | I/O |
| 0 | IO | IO | IO | G9 | I/O |
| 0 | IO/VREF_0 | IO/VREF_0 | IO/VREF_0 | B11 | VREF |
| 0 | IO_L01N_0 | IO_L01N_0 | IO_L01N_0 | A16 | I/O |
| 0 | IO_L01P_0 | IO_L01P_0 | IO_L01P_0 | B16 | I/O |
| 0 | IO_L03N_0/VREF_0 | IO_L03N_0/VREF_0 | IO_L03N_0/VREF_0 | C14 | VREF |
| 0 | IO_L03P_0 | IO_L03P_0 | IO_L03P_0 | D14 | I/O |
| 0 | IO_L04N_0 | IO_L04N_0 | IO_L04N_0 | A14 | I/O |
| 0 | IO_L04P_0 | IO_L04P_0 | IO_L04P_0 | B14 | I/O |
| 0 | IO_L05N_0/VREF_0 | IO_L05N_0/VREF_0 | IO_L05N_0/VREF_0 | B13 | VREF |
| 0 | IO_L05P_0 | IO_L05P_0 | IO_L05P_0 | A13 | I/O |
| 0 | IO_L06N_0 | IO_L06N_0 | IO_L06N_0 | E12 | I/O |
| 0 | IO_L06P_0 | IO_L06P_0 | IO_L06P_0 | F12 | I/O |
| 0 | IO_L08N_0 | IO_L08N_0 | IO_L08N_0 | F11 | I/O |