

Welcome to [E-XFL.COM](https://www.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	612
Number of Logic Elements/Cells	5508
Total RAM Bits	221184
Number of I/O	92
Number of Gates	250000
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	<a href="https://www.e-xfl.com/product-detail/xilinx/xc3s250e-4cp132i">https://www.e-xfl.com/product-detail/xilinx/xc3s250e-4cp132i</a>

## Package Marking

Figure 2 provides a top marking example for Spartan-3E FPGAs in the quad-flat packages. Figure 3 shows the top marking for Spartan-3E FPGAs in BGA packages except the 132-ball chip-scale package (CP132 and CPG132). The markings for the BGA packages are nearly identical to those for the quad-flat packages, except that the marking is rotated with respect to the ball A1 indicator. Figure 4 shows the top marking for Spartan-3E FPGAs in the CP132 and CPG132 packages.

On the QFP and BGA packages, the optional numerical Stepping Code follows the Lot Code.

The “5C” and “4I” part combinations can have a dual mark of “5C/4I”. Devices with a single mark are only guaranteed for the marked speed grade and temperature range. All “5C” and “4I” part combinations use the Stepping 1 production silicon.

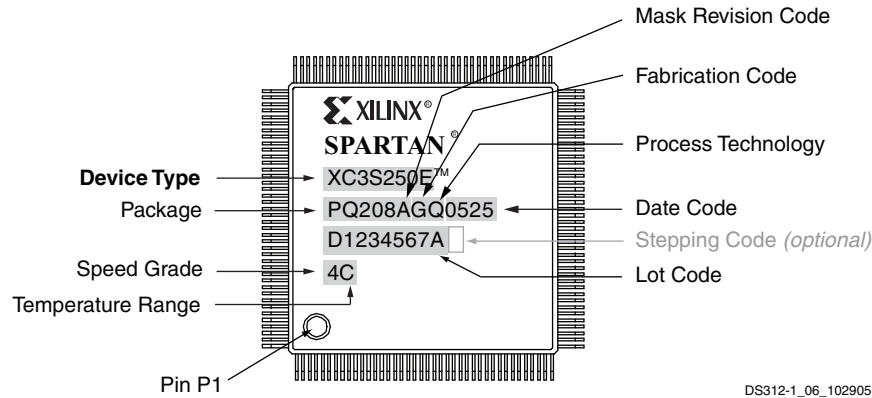


Figure 2: Spartan-3E QFP Package Marking Example

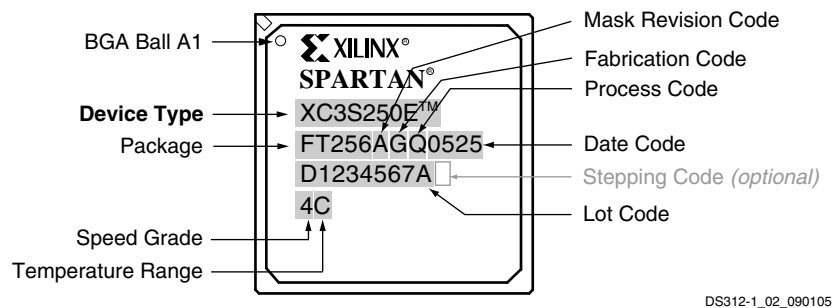


Figure 3: Spartan-3E BGA Package Marking Example

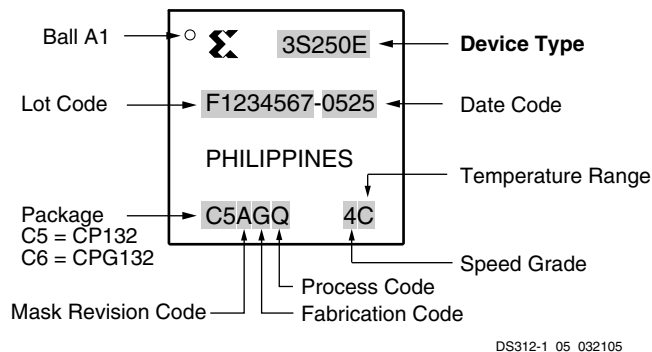


Figure 4: Spartan-3E CP132 and CPG132 Package Marking Example

**Table 10: Slice Inputs and Outputs (Cont'd)**

Name	Location	Direction	Description
SHIFTOUT	SLICEM Bottom	Output	Shift data output from F-LUT RAM
CIN	SLICEL/M Bottom	Input	Carry chain input
COUT	SLICEL/M Top	Output	Carry chain output
X	SLICEL/M Bottom	Output	Combinatorial output
Y	SLICEL/M Top	Output	Combinatorial output
XB	SLICEL/M Bottom	Output	Combinatorial output from carry or F-LUT SRL16 (SLICEM)
YB	SLICEL/M Top	Output	Combinatorial output from carry or G-LUT SRL16 (SLICEM)
XQ	SLICEL/M Bottom	Output	FFX output
YQ	SLICEL/M Top	Output	FFY output

## Main Logic Paths

Central to the operation of each slice are two nearly identical data paths at the top and bottom of the slice. The description that follows uses names associated with the bottom path. (The top path names appear in parentheses.) The basic path originates at an interconnect switch matrix outside the CLB. See [Interconnect](#) for more information on the switch matrix and the routing connections.

Four lines, F1 through F4 (or G1 through G4 on the upper path), enter the slice and connect directly to the LUT. Once inside the slice, the lower 4-bit path passes through a LUT 'F' (or 'G') that performs logic operations. The LUT Data output, 'D', offers five possible paths:

1. Exit the slice via line "X" (or "Y") and return to interconnect.
2. Inside the slice, "X" (or "Y") serves as an input to the DXMUX (or DYMUX) which feeds the data input, "D", of the FFX (or FFY) storage element. The "Q" output of the storage element drives the line XQ (or YQ) which exits the slice.
3. Control the CYMUXF (or CYMUXG) multiplexer on the carry chain.
4. With the carry chain, serve as an input to the XORF (or XORG) exclusive-OR gate that performs arithmetic operations, producing a result on "X" (or "Y").
5. Drive the multiplexer F5MUX to implement logic functions wider than four bits. The "D" outputs of both the F-LUT and G-LUT serve as data inputs to this multiplexer.

In addition to the main logic paths described above, there are two bypass paths that enter the slice as BX and BY. Once inside the FPGA, BX in the bottom half of the slice (or BY in the top half) can take any of several possible branches:

1. Bypass both the LUT and the storage element, and then exit the slice as BXOUT (or BYOUT) and return to interconnect.

2. Bypass the LUT, and then pass through a storage element via the D input before exiting as XQ (or YQ).
3. Control the wide function multiplexer F5MUX (or FiMUX).
4. Via multiplexers, serve as an input to the carry chain.
5. Drive the DI input of the LUT.
6. BY can control the REV inputs of both the FFY and FFX storage elements. See [Storage Element Functions](#).
7. Finally, the DIG\_MUX multiplexer can switch BY onto the DIG line, which exits the slice.

The control inputs CLK, CE, SR, BX and BY have programmable polarity. The LUT inputs do not need programmable polarity because their function can be inverted inside the LUT.

The sections that follow provide more detail on individual functions of the slice.

## Look-Up Tables

The Look-Up Table or LUT is a RAM-based function generator and is the main resource for implementing logic functions. Furthermore, the LUTs in each SLICEM pair can be configured as Distributed RAM or a 16-bit shift register, as described later.

Each of the two LUTs (F and G) in a slice have four logic inputs (A1-A4) and a single output (D). Any four-variable Boolean logic operation can be implemented in one LUT. Functions with more inputs can be implemented by cascading LUTs or by using the wide function multiplexers that are described later.

The output of the LUT can connect to the wide multiplexer logic, the carry and arithmetic logic, or directly to a CLB output or to the CLB storage element. See [Figure 18](#).

## 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  $P = A \times B$ , 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.

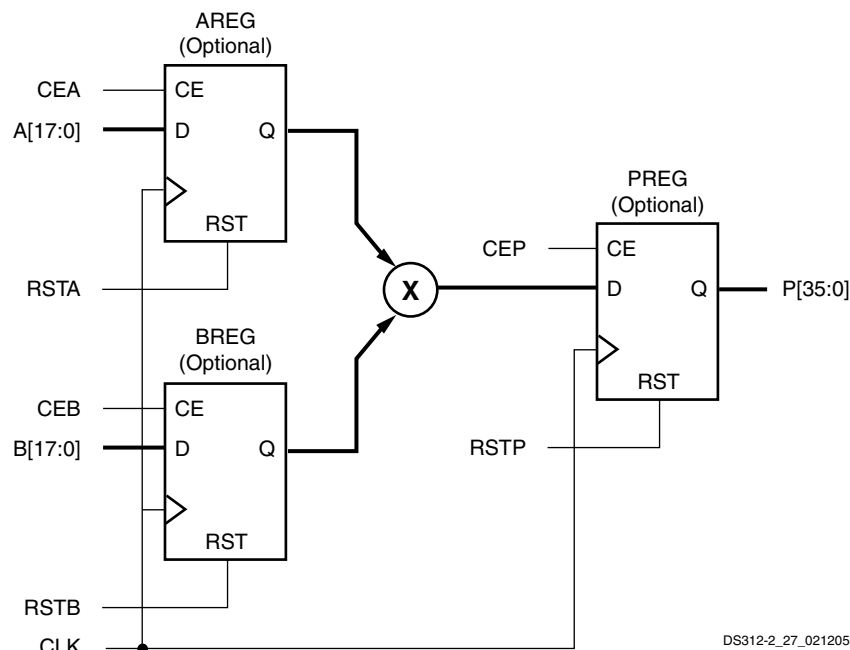


Figure 36: Principle Ports and Functions of Dedicated Multiplier Blocks

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.

## Direct Connections

Direct connect lines route signals to neighboring tiles: vertically, horizontally, and diagonally. These lines most often drive a signal from a “source” tile to a double, hex, or long line and conversely from the longer interconnect back to a direct line accessing a “destination” tile.

## Global Controls (STARTUP\_SPARTAN3E)

In addition to the general-purpose interconnect, Spartan-3E FPGAs have two global logic control signals, as described in [Table 43](#). These signals are available to the FPGA application via the STARTUP\_SPARTAN3E primitive.

*Table 43: Spartan-3E Global Logic Control Signals*

Global Control Input	Description
GSR	<b>Global Set/Reset:</b> When High, asynchronously places all registers and flip-flops in their initial state (see <a href="#">Initialization, page 31</a> ). Asserted automatically during the FPGA configuration process (see <a href="#">Start-Up, page 105</a> ).
GTS	<b>Global Three-State:</b> When High, asynchronously forces all I/O pins to a high-impedance state (Hi-Z, three-state).

The Global Set/Reset (GSR) signal replaces the global reset signal included in many ASIC-style designs. Use the GSR control instead of a separate global reset signal in the design to free up CLB inputs, resulting in a smaller, more efficient design. Similarly, the GSR signal is asserted automatically during the FPGA configuration process, guaranteeing that the FPGA starts-up in a known state.

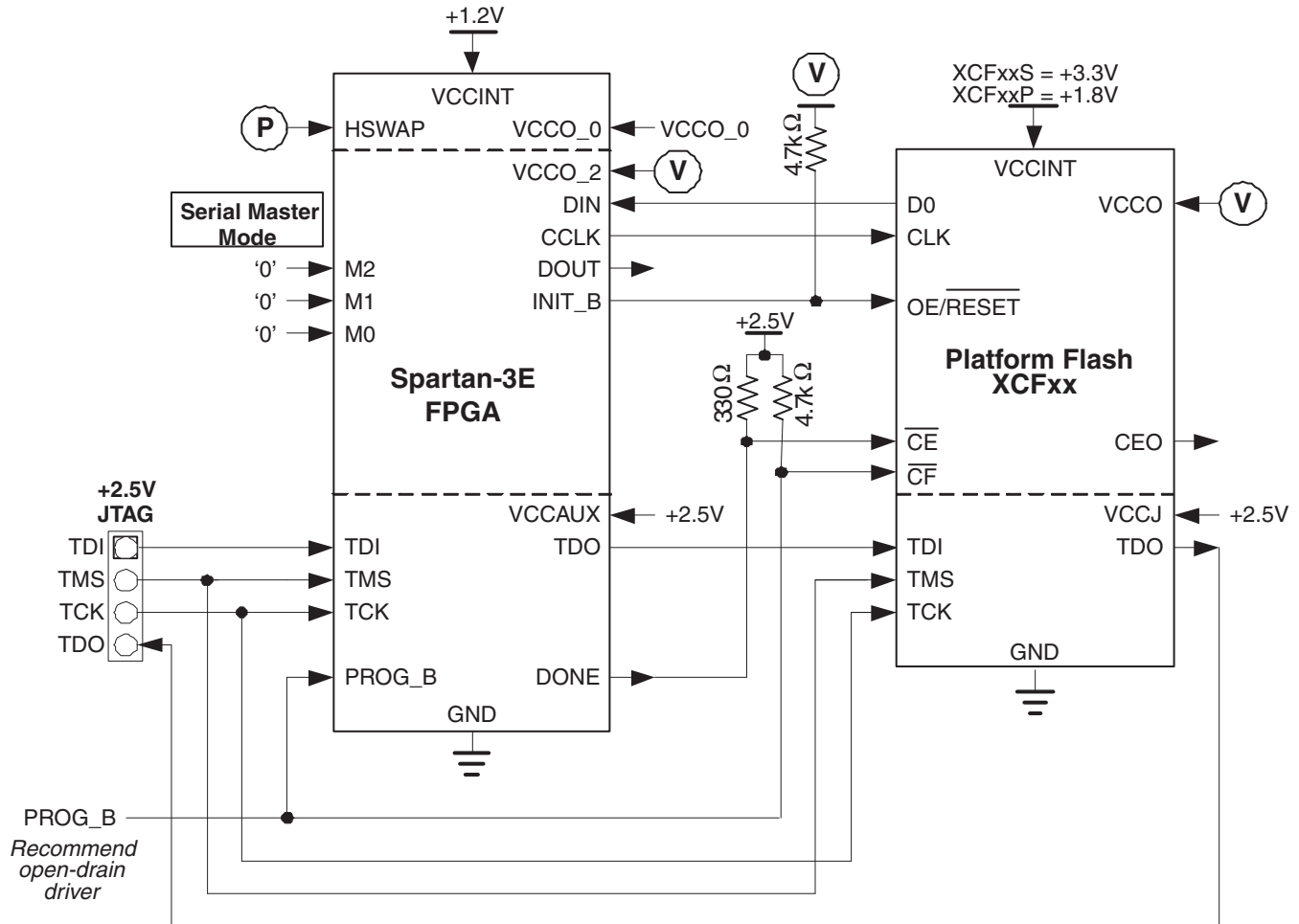
The STARTUP\_SPARTAN3E primitive also includes two other signals used specifically during configuration. The MBT signals are for [Dynamically Loading Multiple Configuration Images Using MultiBoot Option, page 91](#). The CLK input is an alternate clock for configuration [Start-Up, page 105](#).

## Master Serial Mode

For additional information, refer to the “Master Serial Mode” chapter in [UG332](#).

In Master Serial mode ( $M[2:0] = <0:0:0>$ ), the Spartan-3E FPGA configures itself from an attached Xilinx Platform

Flash PROM, as illustrated in [Figure 51](#). The FPGA supplies the CCLK output clock from its internal oscillator to the attached Platform Flash PROM. In response, the Platform Flash PROM supplies bit-serial data to the FPGA's DIN input, and the FPGA accepts this data on each rising CCLK edge.



DS312-2\_44\_082009

Figure 51: Master Serial Mode using Platform Flash PROM

All mode select pins,  $M[2:0]$ , must be Low when sampled, when the FPGA's INIT\_B output goes High. After configuration, when the FPGA's DONE output goes High, the mode select pins are available as full-featured user-I/O pins.

**W** Table 54 shows the connections between the SPI Flash PROM and the FPGA's SPI configuration interface. Each SPI Flash PROM vendor uses slightly different signal naming. The SPI Flash PROM's write protect and hold

controls are not used by the FPGA during configuration. However, the  $\overline{\text{HOLD}}$  pin must be High during the configuration process. The PROM's write protect input must be High in order to write or program the Flash memory.

Table 54: Example SPI Flash PROM Connections and Pin Naming

SPI Flash Pin	FPGA Connection	STMicro	NexFlash	Silicon Storage Technology	Atmel DataFlash
DATA_IN	MOSI	D	DI	SI	SI
DATA_OUT	DIN	Q	DO	SO	SO
$\overline{\text{SELECT}}$	CSO_B	$\overline{\text{S}}$	$\overline{\text{CS}}$	CE#	$\overline{\text{CS}}$
CLOCK	CCLK	C	CLK	SCK	SCK
$\overline{\text{WR\_PROTECT}}$ <b>W</b>	Not required for FPGA configuration. Must be High to program SPI Flash. Optional connection to FPGA user I/O after configuration.	$\overline{\text{W}}$	$\overline{\text{WP}}$	WP#	$\overline{\text{WP}}$
$\overline{\text{HOLD}}$ (see Figure 53)	Not required for FPGA configuration but must be High during configuration. Optional connection to FPGA user I/O after configuration. Not applicable to Atmel DataFlash.	$\overline{\text{HOLD}}$	$\overline{\text{HOLD}}$	HOLD#	N/A
$\overline{\text{RESET}}$ (see Figure 54)	Only applicable to Atmel DataFlash. Not required for FPGA configuration but must be High during configuration. Optional connection to FPGA user I/O after configuration. Do not connect to FPGA's PROG_B as this will prevent direct programming of the DataFlash.	N/A	N/A	N/A	$\overline{\text{RESET}}$
RDY/ $\overline{\text{BUSY}}$ (see Figure 54)	Only applicable to Atmel DataFlash and only available on certain packages. Not required for FPGA configuration. Output from DataFlash PROM. Optional connection to FPGA user I/O after configuration.	N/A	N/A	N/A	RDY/ $\overline{\text{BUSY}}$

The mode select pins, M[2:0], and the variant select pins, VS[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, these pins are all 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.

In a single-FPGA application, the FPGA's DOUT pin is not used but is actively driving during the configuration process.

Table 55: Serial Peripheral Interface (SPI) Connections

Pin Name	FPGA Direction	Description	During Configuration	After Configuration
HSWAP <b>P</b>	Input	<b>User I/O Pull-Up Control.</b> When Low during configuration, enables pull-up resistors in all I/O pins to respective I/O bank V <sub>CCO</sub> input. 0: Pull-ups during configuration 1: No pull-ups	Drive at valid logic level throughout configuration.	User I/O
M[2:0]	Input	<b>Mode Select.</b> Selects the FPGA configuration mode. See <a href="#">Design Considerations for the HSWAP, M[2:0], and VS[2:0] Pins</a> .	M2 = 0, M1 = 0, M0 = 1. Sampled when INIT_B goes High.	User I/O



## Compatible Flash Families

The Spartan-3E BPI configuration interface operates with a wide variety of x8 or x8/x16 parallel NOR Flash devices. [Table 61](#) provides a few Flash memory families that operate with the Spartan-3E BPI interface. Consult the data sheet for the desired parallel NOR Flash to determine its suitability. The basic timing requirements and waveforms are provided in [Byte Peripheral Interface \(BPI\) Configuration Timing](#) (Module 3).

Table 61: Compatible Parallel NOR Flash Families

Flash Vendor	Flash Memory Family
<a href="#">Numonyx</a>	M29W, J3D StrataFlash
<a href="#">Atmel</a>	<a href="#">AT29 / AT49</a>
<a href="#">Spansion</a>	S29
<a href="#">Macronix</a>	MX29

## CCLK Frequency

In BPI mode, the FPGA's internal oscillator generates the configuration clock frequency that controls all the interface timing. 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.

Table 62: Maximum ConfigRate Settings for Parallel Flash PROMs (Commercial Temperature Range)

Flash Read Access Time	Maximum ConfigRate Setting
250 ns	3
115 ns	6
45 ns	12

[Table 62](#) shows the maximum [ConfigRate](#) settings for various typical PROM read access times over the Commercial temperature operating range. See [Byte Peripheral Interface \(BPI\) Configuration Timing](#) (Module 3) and [UG332](#) for more detailed information. Despite using slower [ConfigRate](#) settings, BPI mode is equally fast as the other configuration modes. In BPI mode, data is accessed at the [ConfigRate](#) frequency and internally serialized with an 8X clock frequency.

## Using the BPI Interface after Configuration

After the FPGA successfully completes configuration, all pins connected to the parallel Flash PROM are available as user I/Os.

If not using the parallel Flash PROM after configuration, drive LDC0 High to disable the PROM's chip-select input. The remainder of the BPI pins then become available to the FPGA application, including all 24 address lines, the eight data lines, and the LDC2, LDC1, and HDC control pins.

Because all the interface pins are user I/Os after configuration, the FPGA application can continue to use the interface pins to communicate with the parallel Flash PROM. Parallel Flash PROMs are available in densities ranging from 1 Mbit up to 128 Mbits and beyond. However, a single Spartan-3E FPGA requires less than 6 Mbits for configuration. If desired, use a larger parallel 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 such an example, the FPGA configures from parallel Flash PROM. Then using FPGA logic after configuration, a MicroBlaze processor embedded within the FPGA can either execute code directly from parallel Flash PROM or copy the code to external DDR SDRAM and execute from DDR SDRAM. Similarly, the FPGA application can store non-volatile application data within the parallel Flash PROM.

The FPGA configuration data is stored starting at either at location 0 or the top of memory (addresses all ones) or at both locations for MultiBoot mode. Store any additional data beginning in other available parallel Flash PROM sectors. Do not mix configuration data and user data in the same sector.

Similarly, the parallel Flash PROM interface can be expanded to additional parallel peripherals.

The address, data, and LDC1 (OE#) and HDC (WE#) control signals are common to all parallel peripherals. Connect the chip-select input on each additional peripheral to one of the FPGA user I/O pins. If HSWAP = 0 during configuration, the FPGA holds the chip-select line High via an internal pull-up resistor. 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 peripheral. Refer to the individual peripheral data sheet for specific interface and communication protocol requirements.

The FPGA optionally supports a 16-bit peripheral interface by driving the LDC2 (BYTE#) control pin High after configuration. See [Precautions Using x8/x16 Flash PROMs](#) for additional information.

The FPGA provides up to 24 address lines during configuration, addressing up to 128 Mbits (16 Mbytes). If using a larger parallel PROM, connect the upper address lines to FPGA user I/O. During configuration, the upper address lines will be pulled High if HSWAP = 0. Otherwise, use external pull-up or pull-down resistors on these address lines to define their values during configuration.

## Precautions Using x8/x16 Flash PROMs

Ⓓ Most low- to mid-density PROMs are byte-wide (x8) only. Many higher-density Flash PROMs support both byte-wide (x8) and halfword-wide (x16) data paths and include a mode input called BYTE# that switches between x8 or x16. During configuration, Spartan-3E FPGAs only



Also, in a multi-FPGA daisy-chain configuration of more than two devices, all intermediate FPGAs between the first and last devices must be Spartan-3E or Virtex-5 FPGAs. The last FPGA in the chain can be from any Xilinx FPGA family.

### BPI Mode Interaction with Right and Bottom Edge Global Clock Inputs

Some of the BPI mode configuration pins are shared with global clock inputs along the right and bottom edges of the device (Bank 1 and Bank 2, respectively). These pins are not easily reclaimable for clock inputs after configuration, especially if the FPGA application access the parallel NOR Flash after configuration. [Table 64](#) summarizes the shared pins.

*Table 64: Shared BPI Configuration Mode and Global Buffer Input Pins*

Device Edge	Global Buffer Input Pin	BPI Mode Configuration Pin
Bottom	GCLK0	RDWR_B
	GCLK2	D2
	GCLK3	D1
	GCLK12	D7
	GCLK13	D6
	GCLK14	D4
	GCLK15	D3
Right	RHCLK0	A10
	RHCLK1	A9
	RHCLK2	A8
	RHCLK3	A7
	RHCLK4	A6
	RHCLK5	A5
	RHCLK6	A4
	RHCLK7	A3

FPGA passes configuration data via its DOUT output pin to the next FPGA on the falling CCLK edge.

**Table 66: Slave Serial Mode Connections**

Pin Name	FPGA Direction	Description	During Configuration	After Configuration
HSWAP	Input	<b>User I/O Pull-Up Control.</b> When Low during configuration, enables pull-up resistors in all I/O pins to respective I/O bank V <sub>CCO</sub> input. 0: Pull-up during configuration 1: No pull-ups	Drive at valid logic level throughout configuration.	User I/O
M[2:0]	Input	<b>Mode Select.</b> Selects the FPGA configuration mode. See <a href="#">Design Considerations for the HSWAP, M[2:0], and VS[2:0] Pins.</a>	M2 = 1, M1 = 1, M0 = 1 Sampled when INIT_B goes High.	User I/O
DIN	Input	<b>Data Input.</b>	Serial data provided by host. FPGA captures data on rising CCLK edge.	User I/O
CCLK	Input	<b>Configuration Clock.</b> If CCLK PCB trace is long or has multiple connections, terminate this output to maintain signal integrity. See <a href="#">CCLK Design Considerations.</a>	External clock.	User I/O
INIT_B	Open-drain bidirectional I/O	<b>Initialization Indicator.</b> Active Low. Goes Low at start of configuration during Initialization memory clearing process. Released at end of memory clearing, when mode select pins are sampled. In daisy-chain applications, this signal requires an external 4.7 kΩ pull-up resistor to V <sub>CCO_2</sub> .	Active during configuration. If CRC error detected during configuration, FPGA drives INIT_B Low.	User I/O. If unused in the application, drive INIT_B High.
DONE	Open-drain bidirectional I/O	<b>FPGA Configuration Done.</b> Low during configuration. Goes High when FPGA successfully completes configuration. Requires external 330 Ω pull-up resistor to 2.5V.	Low indicates that the FPGA is not yet configured.	Pulled High via external pull-up. When High, indicates that the FPGA successfully configured.
PROG_B	Input	<b>Program FPGA.</b> Active Low. When asserted Low for 500 ns or longer, forces the FPGA to restart its configuration process by clearing configuration memory and resetting the DONE and INIT_B pins once PROG_B returns High. Recommend external 4.7 kΩ pull-up resistor to 2.5V. Internal pull-up value may be weaker (see <a href="#">Table 78</a> ). If driving externally with a 3.3V output, use an open-drain or open-collector driver or use a current limiting series resistor.	Must be High to allow configuration to start.	Drive PROG_B Low and release to reprogram FPGA.

## Start-Up

At the end of configuration, the FPGA automatically pulses the Global Set/Reset (GSR) signal, placing all flip-flops in a known state. After configuration completes, the FPGA switches over to the user application loaded into the FPGA. The sequence and timing of how the FPGA switches over is programmable as is the clock source controlling the sequence.

The default start-up sequence appears in Figure 68, where the Global Three-State signal (GTS) is released one clock cycle after DONE goes High. This sequence allows the DONE signal to enable or disable any external logic used during configuration before the user application in the FPGA starts driving output signals. One clock cycle later, the Global Write Enable (GWE) signal is released. This allows signals to propagate within the FPGA before any clocked storage elements such as flip-flops and block ROM are enabled.

become user I/Os. Like all user-I/O pins, GTS controls when the dual-purpose pins can drive out.

The relative timing of configuration events is programmed via the Bitstream Generator (BitGen) options in the Xilinx development software. For example, the GTS and GWE events can be programmed to wait for all the DONE pins to High on all the devices in a multiple-FPGA daisy-chain, forcing the FPGAs to start synchronously. Similarly, the start-up sequence can be paused at any stage, waiting for selected DCMs to lock to their respective input clock signals. See also [Stabilizing DCM Clocks Before User Mode](#).

By default, the start-up sequence is synchronized to CCLK. Alternatively, the start-up sequence can be synchronized to a user-specified clock from within the FPGA application using the STARTUP\_SPARTAN3E library primitive and by setting the [StartupClk](#) bitstream generator option. The FPGA application can optionally assert the GSR and GTS signals via the STARTUP\_SPARTAN3E primitive. For JTAG configuration, the start-up sequence can be synchronized to the TCK clock input.

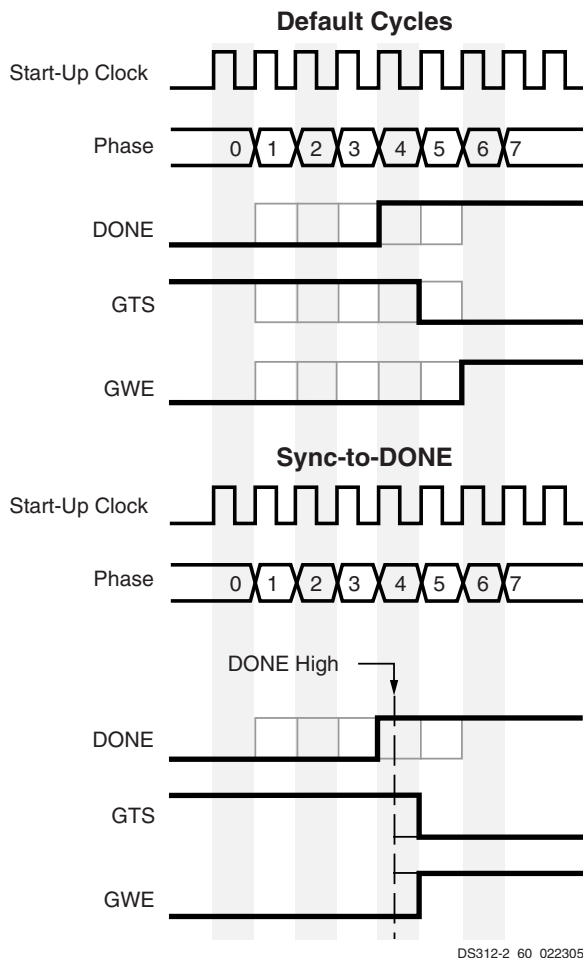


Figure 68: Default Start-Up Sequence

The function of the dual-purpose I/O pins, such as M[2:0], VS[2:0], HSWAP, and A[23:0], also changes when the DONE pin goes High. When DONE is High, these pins

Table 95: Test Methods for Timing Measurement at I/Os (Cont'd)

Signal Standard (IOSTANDARD)	Inputs			Outputs		Inputs and Outputs
	$V_{REF}$ (V)	$V_L$ (V)	$V_H$ (V)	$R_T$ ( $\Omega$ )	$V_T$ (V)	$V_M$ (V)
DIFF_HSTL_I_18	-	$V_{REF} - 0.5$	$V_{REF} + 0.5$	50	0.9	$V_{ICM}$
DIFF_HSTL_III_18	-	$V_{REF} - 0.5$	$V_{REF} + 0.5$	50	1.8	$V_{ICM}$
DIFF_SSTL18_I	-	$V_{REF} - 0.5$	$V_{REF} + 0.5$	50	0.9	$V_{ICM}$
DIFF_SSTL2_I	-	$V_{REF} - 0.5$	$V_{REF} + 0.5$	50	1.25	$V_{ICM}$

**Notes:**

- Descriptions of the relevant symbols are as follows:  
 $V_{REF}$  – The reference voltage for setting the input switching threshold  
 $V_{ICM}$  – The common mode input voltage  
 $V_M$  – Voltage of measurement point on signal transition  
 $V_L$  – Low-level test voltage at Input pin  
 $V_H$  – High-level test voltage at Input pin  
 $R_T$  – Effective termination resistance, which takes on a value of  $1M\Omega$  when no parallel termination is required  
 $V_T$  – Termination voltage
- The load capacitance ( $C_L$ ) at the Output pin is 0 pF for all signal standards.
- According to the PCI specification.

The capacitive load ( $C_L$ ) is connected between the output and GND. *The Output timing for all standards, as published in the speed files and the data sheet, is always based on a  $C_L$  value of zero.* High-impedance probes (less than 1 pF) are used for all measurements. Any delay that the test fixture might contribute to test measurements is subtracted from those measurements to produce the final timing numbers as published in the speed files and data sheet.

## Using IBIS Models to Simulate Load Conditions in Application

IBIS models permit the most accurate prediction of timing delays for a given application. The parameters found in the IBIS model ( $V_{REF}$ ,  $R_{REF}$ , and  $V_{MEAS}$ ) correspond directly with the parameters used in Table 95 ( $V_T$ ,  $R_T$ , and  $V_M$ ). Do not confuse  $V_{REF}$  (the termination voltage) from the IBIS model with  $V_{REF}$  (the input-switching threshold) from the table. A fourth parameter,  $C_{REF}$  is always zero. The four parameters describe all relevant output test conditions. IBIS models are found in the Xilinx development software as well as at the following link:

<http://www.xilinx.com/support/download/index.htm>

Delays for a given application are simulated according to its specific load conditions as follows:

- Simulate the desired signal standard with the output driver connected to the test setup shown in Figure 72. Use parameter values  $V_T$ ,  $R_T$ , and  $V_M$  from Table 95.  $C_{REF}$  is zero.
- Record the time to  $V_M$ .
- Simulate the same signal standard with the output driver connected to the PCB trace with load. Use the appropriate IBIS model (including  $V_{REF}$ ,  $R_{REF}$ ,  $C_{REF}$ , and  $V_{MEAS}$  values) or capacitive value to represent the load.
- Record the time to  $V_{MEAS}$ .
- Compare the results of steps 2 and 4. Add (or subtract) the increase (or decrease) in delay to (or from) the appropriate Output standard adjustment (Table 94) to yield the worst-case delay of the PCB trace.

**Table 131: VQ100 Package Pinout (Cont'd)**

<b>Bank</b>	<b>XC3S100E XC3S250E XC3S500E Pin Name</b>	<b>VQ100 Pin Number</b>	<b>Type</b>
3	IO_L02P_3	P4	I/O
3	IO_L03N_3/LHCLK1	P10	LHCLK
3	IO_L03P_3/LHCLK0	P9	LHCLK
3	IO_L04N_3/LHCLK3	P12	LHCLK
3	IO_L04P_3/LHCLK2	P11	LHCLK
3	IO_L05N_3/LHCLK5	P16	LHCLK
3	IO_L05P_3/LHCLK4	P15	LHCLK
3	IO_L06N_3/LHCLK7	P18	LHCLK
3	IO_L06P_3/LHCLK6	P17	LHCLK
3	IO_L07N_3	P23	I/O
3	IO_L07P_3	P22	I/O
3	IP	P13	INPUT
3	VCCO_3	P8	VCCO
3	VCCO_3	P20	VCCO
GND	GND	P7	GND
GND	GND	P14	GND
GND	GND	P19	GND
GND	GND	P29	GND
GND	GND	P37	GND
GND	GND	P52	GND
GND	GND	P59	GND
GND	GND	P64	GND
GND	GND	P72	GND
GND	GND	P81	GND
GND	GND	P87	GND
GND	GND	P93	GND
VCCAUX	DONE	P51	CONFIG
VCCAUX	PROG_B	P1	CONFIG
VCCAUX	TCK	P77	JTAG
VCCAUX	TDI	P100	JTAG
VCCAUX	TDO	P76	JTAG
VCCAUX	TMS	P75	JTAG
VCCAUX	VCCAUX	P21	VCCAUX
VCCAUX	VCCAUX	P46	VCCAUX
VCCAUX	VCCAUX	P74	VCCAUX
VCCAUX	VCCAUX	P96	VCCAUX
VCCINT	VCCINT	P6	VCCINT
VCCINT	VCCINT	P28	VCCINT
VCCINT	VCCINT	P56	VCCINT
VCCINT	VCCINT	P80	VCCINT

**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	<b>100E:</b> VREF(INPUT) <b>Others:</b> 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/DOOUT/BUSY	IO_L02P_2/DOOUT/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	<b>100E:</b> N.C. <b>Others:</b> DUAL



## TQ144: 144-lead Thin Quad Flat Package

The XC3S100E and the XC3S250E FPGAs are available in the 144-lead thin quad flat package, TQ144. Both devices share a common footprint for this package as shown in [Table 137](#) and [Figure 82](#).

[Table 137](#) 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 TQ144 package only supports 20 address output pins in the Byte-wide Peripheral Interface (BPI) configuration mode. In larger packages, there are 24 BPI address outputs.

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

### Pinout Table

*Table 137: TQ144 Package Pinout*

Bank	XC3S100E Pin Name	XC3S250E Pin Name	TQ144 Pin	Type
0	IO	IO	P132	I/O
0	IO/VREF_0	IO/VREF_0	P124	VREF
0	IO_L01N_0	IO_L01N_0	P113	I/O
0	IO_L01P_0	IO_L01P_0	P112	I/O
0	IO_L02N_0	IO_L02N_0	P117	I/O
0	IO_L02P_0	IO_L02P_0	P116	I/O
0	IO_L04N_0/GCLK5	IO_L04N_0/GCLK5	P123	GCLK
0	IO_L04P_0/GCLK4	IO_L04P_0/GCLK4	P122	GCLK
0	IO_L05N_0/GCLK7	IO_L05N_0/GCLK7	P126	GCLK
0	IO_L05P_0/GCLK6	IO_L05P_0/GCLK6	P125	GCLK
0	IO_L07N_0/GCLK11	IO_L07N_0/GCLK11	P131	GCLK
0	IO_L07P_0/GCLK10	IO_L07P_0/GCLK10	P130	GCLK
0	IO_L08N_0/VREF_0	IO_L08N_0/VREF_0	P135	VREF
0	IO_L08P_0	IO_L08P_0	P134	I/O
0	IO_L09N_0	IO_L09N_0	P140	I/O
0	IO_L09P_0	IO_L09P_0	P139	I/O
0	IO_L10N_0/HSWAP	IO_L10N_0/HSWAP	P143	DUAL
0	IO_L10P_0	IO_L10P_0	P142	I/O
0	IP	IP	P111	INPUT
0	IP	IP	P114	INPUT
0	IP	IP	P136	INPUT
0	IP	IP	P141	INPUT
0	IP_L03N_0	IP_L03N_0	P120	INPUT
0	IP_L03P_0	IP_L03P_0	P119	INPUT
0	IP_L06N_0/GCLK9	IP_L06N_0/GCLK9	P129	GCLK
0	IP_L06P_0/GCLK8	IP_L06P_0/GCLK8	P128	GCLK
0	VCCO_0	VCCO_0	P121	VCCO
0	VCCO_0	VCCO_0	P138	VCCO
1	IO/A0	IO/A0	P98	DUAL
1	IO/VREF_1	IO/VREF_1	P83	VREF
1	IO_L01N_1/A15	IO_L01N_1/A15	P75	DUAL
1	IO_L01P_1/A16	IO_L01P_1/A16	P74	DUAL
1	IO_L02N_1/A13	IO_L02N_1/A13	P77	DUAL

**Table 141: PQ208 Package Pinout (Cont'd)**

Bank	XC3S250E XC3S500E Pin Name	PQ208 Pin	Type
3	IO_L14P_3	P41	I/O
3	IO_L15N_3	P48	I/O
3	IO_L15P_3	P47	I/O
3	IO_L16N_3	P50	I/O
3	IO_L16P_3	P49	I/O
3	IP	P6	INPUT
3	IP	P14	INPUT
3	IP	P26	INPUT
3	IP	P32	INPUT
3	IP	P43	INPUT
3	IP	P51	INPUT
3	IP/VREF_3	P20	VREF
3	VCCO_3	P21	VCCO
3	VCCO_3	P38	VCCO
3	VCCO_3	P46	VCCO
GND	GND	P10	GND
GND	GND	P17	GND
GND	GND	P27	GND
GND	GND	P37	GND
GND	GND	P52	GND
GND	GND	P53	GND
GND	GND	P70	GND
GND	GND	P79	GND
GND	GND	P85	GND
GND	GND	P95	GND
GND	GND	P105	GND
GND	GND	P121	GND
GND	GND	P131	GND
GND	GND	P141	GND
GND	GND	P156	GND
GND	GND	P173	GND
GND	GND	P182	GND
GND	GND	P188	GND
GND	GND	P198	GND
GND	GND	P208	GND
VCCAUX	DONE	P104	CONFIG
VCCAUX	PROG_B	P1	CONFIG
VCCAUX	TCK	P158	JTAG
VCCAUX	TDI	P207	JTAG
VCCAUX	TDO	P157	JTAG
VCCAUX	TMS	P155	JTAG
VCCAUX	VCCAUX	P7	VCCAUX
VCCAUX	VCCAUX	P44	VCCAUX

**Table 141: PQ208 Package Pinout (Cont'd)**

Bank	XC3S250E XC3S500E Pin Name	PQ208 Pin	Type
VCCAUX	VCCAUX	P66	VCCAUX
VCCAUX	VCCAUX	P92	VCCAUX
VCCAUX	VCCAUX	P111	VCCAUX
VCCAUX	VCCAUX	P149	VCCAUX
VCCAUX	VCCAUX	P166	VCCAUX
VCCAUX	VCCAUX	P195	VCCAUX
VCCINT	VCCINT	P13	VCCINT
VCCINT	VCCINT	P67	VCCINT
VCCINT	VCCINT	P117	VCCINT
VCCINT	VCCINT	P170	VCCINT

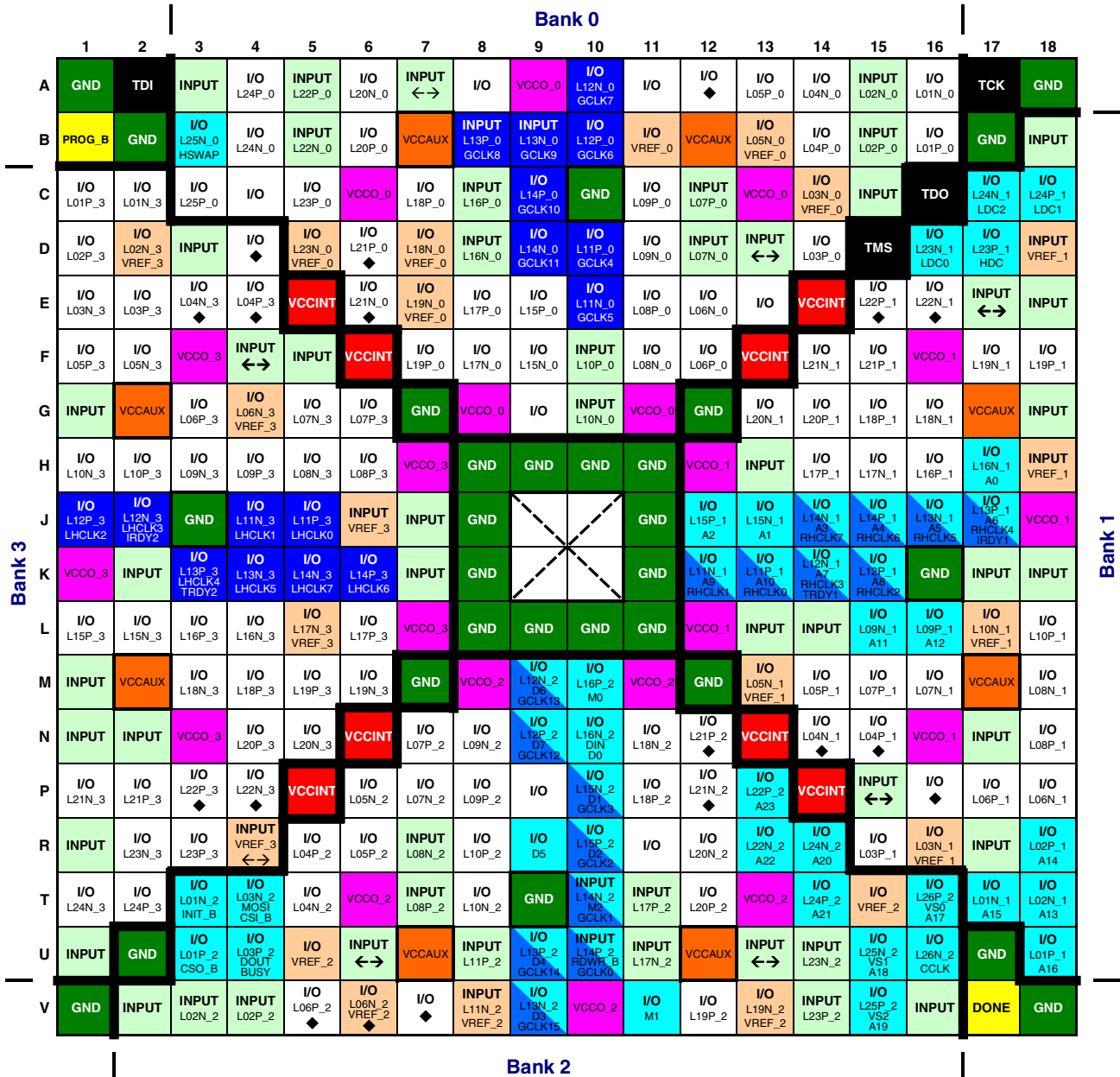
**Table 143: FT256 Package Pinout (Cont'd)**

Bank	XC3S250E Pin Name	XC3S500E Pin Name	XC3S1200E Pin Name	FT256 Ball	Type
2	N.C. (◆)	IO_L14N_2/VREF_2	IO_L14N_2/VREF_2	R10	<b>250E:</b> N.C. <b>500E:</b> VREF <b>1200E:</b> VREF
2	N.C. (◆)	IO_L14P_2	IO_L14P_2	P10	<b>250E:</b> N.C. <b>500E:</b> I/O <b>1200E:</b> I/O
2	IO_L15N_2	IO_L15N_2	IO_L15N_2	M10	I/O
2	IO_L15P_2	IO_L15P_2	IO_L15P_2	N10	I/O
2	IO_L16N_2/A22	IO_L16N_2/A22	IO_L16N_2/A22	P11	DUAL
2	IO_L16P_2/A23	IO_L16P_2/A23	IO_L16P_2/A23	R11	DUAL
2	IO_L18N_2/A20	IO_L18N_2/A20	IO_L18N_2/A20	N12	DUAL
2	IO_L18P_2/A21	IO_L18P_2/A21	IO_L18P_2/A21	P12	DUAL
2	IO_L19N_2/VS1/A18	IO_L19N_2/VS1/A18	IO_L19N_2/VS1/A18	R13	DUAL
2	IO_L19P_2/VS2/A19	IO_L19P_2/VS2/A19	IO_L19P_2/VS2/A19	T13	DUAL
2	IO_L20N_2/CCLK	IO_L20N_2/CCLK	IO_L20N_2/CCLK	R14	DUAL
2	IO_L20P_2/VS0/A17	IO_L20P_2/VS0/A17	IO_L20P_2/VS0/A17	P14	DUAL
2	IP	IP	IP	T2	INPUT
2	IP	IP	IP	T14	INPUT
2	IP_L02N_2	IP_L02N_2	IP_L02N_2	R3	INPUT
2	IP_L02P_2	IP_L02P_2	IP_L02P_2	T3	INPUT
2	IP_L08N_2/VREF_2	IP_L08N_2/VREF_2	IP_L08N_2/VREF_2	T7	VREF
2	IP_L08P_2	IP_L08P_2	IP_L08P_2	R7	INPUT
2	IP_L11N_2/M2/GCLK1	IP_L11N_2/M2/GCLK1	IP_L11N_2/M2/GCLK1	R9	DUAL/GCLK
2	IP_L11P_2/RDWR_B/ GCLK0	IP_L11P_2/RDWR_B/ GCLK0	IP_L11P_2/RDWR_B/ GCLK0	T9	DUAL/GCLK
2	IP_L17N_2	IP_L17N_2	IP_L17N_2	M11	INPUT
2	IP_L17P_2	IP_L17P_2	IP_L17P_2	N11	INPUT
2	VCCO_2	VCCO_2	VCCO_2	L7	VCCO
2	VCCO_2	VCCO_2	VCCO_2	L10	VCCO
2	VCCO_2	VCCO_2	VCCO_2	R5	VCCO
2	VCCO_2	VCCO_2	VCCO_2	R12	VCCO
3	IO_L01N_3	IO_L01N_3	IO_L01N_3	B2	I/O
3	IO_L01P_3	IO_L01P_3	IO_L01P_3	B1	I/O
3	IO_L02N_3/VREF_3	IO_L02N_3/VREF_3	IO_L02N_3/VREF_3	C2	VREF
3	IO_L02P_3	IO_L02P_3	IO_L02P_3	C1	I/O
3	IO_L03N_3	IO_L03N_3	IO_L03N_3	E4	I/O
3	IO_L03P_3	IO_L03P_3	IO_L03P_3	E3	I/O
3	N.C. (◆)	IO_L04N_3/VREF_3	IO_L04N_3/VREF_3	F4	<b>250E:</b> N.C. <b>500E:</b> VREF <b>1200E:</b> VREF
3	N.C. (◆)	IO_L04P_3	IO_L04P_3	F3	<b>250E:</b> N.C. <b>500E:</b> I/O <b>1200E:</b> I/O
3	IO_L05N_3	IO_L05N_3	IO_L05N_3	E1	I/O
3	IO_L05P_3	IO_L05P_3	IO_L05P_3	D1	I/O
3	IO_L06N_3	IO_L06N_3	IO_L06N_3	G4	I/O

**Table 148: FG320 Package Pinout (Cont'd)**

Bank	XC3S500E Pin Name	XC3S1200E Pin Name	XC3S1600E Pin Name	FG320 Ball	Type
2	IP	IO	IO	U6	<b>500E: INPUT</b> <b>1200E: I/O</b> <b>1600E: I/O</b>
2	IP	IO	IO	U13	<b>500E: INPUT</b> <b>1200E: I/O</b> <b>1600E: I/O</b>
2	N.C. (◆)	IO	IO	V7	<b>500E: N.C.</b> <b>1200E: I/O</b> <b>1600E: I/O</b>
2	IO/D5	IO/D5	IO/D5	R9	DUAL
2	IO/M1	IO/M1	IO/M1	V11	DUAL
2	IO/VREF_2	IO/VREF_2	IO/VREF_2	T15	VREF
2	IO/VREF_2	IO/VREF_2	IO/VREF_2	U5	VREF
2	IO_L01N_2/INIT_B	IO_L01N_2/INIT_B	IO_L01N_2/INIT_B	T3	DUAL
2	IO_L01P_2/CSO_B	IO_L01P_2/CSO_B	IO_L01P_2/CSO_B	U3	DUAL
2	IO_L03N_2/MOSI/CSI_B	IO_L03N_2/MOSI/CSI_B	IO_L03N_2/MOSI/CSI_B	T4	DUAL
2	IO_L03P_2/DOUT/BUSY	IO_L03P_2/DOUT/BUSY	IO_L03P_2/DOUT/BUSY	U4	DUAL
2	IO_L04N_2	IO_L04N_2	IO_L04N_2	T5	I/O
2	IO_L04P_2	IO_L04P_2	IO_L04P_2	R5	I/O
2	IO_L05N_2	IO_L05N_2	IO_L05N_2	P6	I/O
2	IO_L05P_2	IO_L05P_2	IO_L05P_2	R6	I/O
2	N.C. (◆)	IO_L06N_2/VREF_2	IO_L06N_2/VREF_2	V6	<b>500E: N.C.</b> <b>1200E: VREF</b> <b>1600E: VREF</b>
2	N.C. (◆)	IO_L06P_2	IO_L06P_2	V5	<b>500E: N.C.</b> <b>1200E: I/O</b> <b>1600E: I/O</b>
2	IO_L07N_2	IO_L07N_2	IO_L07N_2	P7	I/O
2	IO_L07P_2	IO_L07P_2	IO_L07P_2	N7	I/O
2	IO_L09N_2	IO_L09N_2	IO_L09N_2	N8	I/O
2	IO_L09P_2	IO_L09P_2	IO_L09P_2	P8	I/O
2	IO_L10N_2	IO_L10N_2	IO_L10N_2	T8	I/O
2	IO_L10P_2	IO_L10P_2	IO_L10P_2	R8	I/O
2	IO_L12N_2/D6/GCLK13	IO_L12N_2/D6/GCLK13	IO_L12N_2/D6/GCLK13	M9	DUAL/GCLK
2	IO_L12P_2/D7/GCLK12	IO_L12P_2/D7/GCLK12	IO_L12P_2/D7/GCLK12	N9	DUAL/GCLK
2	IO_L13N_2/D3/GCLK15	IO_L13N_2/D3/GCLK15	IO_L13N_2/D3/GCLK15	V9	DUAL/GCLK
2	IO_L13P_2/D4/GCLK14	IO_L13P_2/D4/GCLK14	IO_L13P_2/D4/GCLK14	U9	DUAL/GCLK
2	IO_L15N_2/D1/GCLK3	IO_L15N_2/D1/GCLK3	IO_L15N_2/D1/GCLK3	P10	DUAL/GCLK
2	IO_L15P_2/D2/GCLK2	IO_L15P_2/D2/GCLK2	IO_L15P_2/D2/GCLK2	R10	DUAL/GCLK
2	IO_L16N_2/DIN/D0	IO_L16N_2/DIN/D0	IO_L16N_2/DIN/D0	N10	DUAL
2	IO_L16P_2/M0	IO_L16P_2/M0	IO_L16P_2/M0	M10	DUAL
2	IO_L18N_2	IO_L18N_2	IO_L18N_2	N11	I/O
2	IO_L18P_2	IO_L18P_2	IO_L18P_2	P11	I/O
2	IO_L19N_2/VREF_2	IO_L19N_2/VREF_2	IO_L19N_2/VREF_2	V13	VREF
2	IO_L19P_2	IO_L19P_2	IO_L19P_2	V12	I/O
2	IO_L20N_2	IO_L20N_2	IO_L20N_2	R12	I/O

# FG320 Footprint



DS312-4\_06\_022106

Figure 86: FG320 Package Footprint (top view)

102-120	I/O: Unrestricted, general-purpose user I/O	46	DUAL: Configuration pin, then possible user-I/O	20-21	VREF: User I/O or input voltage reference for bank
47-48	INPUT: Unrestricted, general-purpose input pin	16	CLK: User I/O, input, or global buffer input	20	VCCO: Output voltage supply for bank
2	CONFIG: Dedicated configuration pins	4	JTAG: Dedicated JTAG port pins	8	VCCINT: Internal core supply voltage (+1.2V)
18	N.C.: Not connected. Only the XC3S500E has these pins (◆).	28	GND: Ground	8	VCCAUX: Auxiliary supply voltage (+2.5V)

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

Bank	XC3S1200E XC3S1600E Pin Name	FG400 Ball	Type
3	IP	N5	INPUT
3	IP	P3	INPUT
3	IP	T4	INPUT
3	IP	W1	INPUT
3	IP/VREF_3	K5	VREF
3	IP/VREF_3	P6	VREF
3	VCCO_3	E2	VCCO
3	VCCO_3	H4	VCCO
3	VCCO_3	L2	VCCO
3	VCCO_3	L6	VCCO
3	VCCO_3	P4	VCCO
3	VCCO_3	U2	VCCO
GND	GND	A1	GND
GND	GND	A11	GND
GND	GND	A20	GND
GND	GND	B7	GND
GND	GND	B14	GND
GND	GND	C3	GND
GND	GND	C18	GND
GND	GND	D10	GND
GND	GND	F6	GND
GND	GND	F15	GND
GND	GND	G2	GND
GND	GND	G12	GND
GND	GND	G19	GND
GND	GND	H8	GND
GND	GND	J9	GND
GND	GND	J11	GND
GND	GND	K1	GND
GND	GND	K8	GND
GND	GND	K10	GND
GND	GND	K12	GND
GND	GND	K17	GND
GND	GND	L4	GND
GND	GND	L9	GND
GND	GND	L11	GND
GND	GND	L13	GND
GND	GND	L20	GND
GND	GND	M10	GND
GND	GND	M12	GND
GND	GND	N13	GND
GND	GND	P2	GND
GND	GND	P9	GND

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

Bank	XC3S1200E XC3S1600E Pin Name	FG400 Ball	Type
GND	GND	P19	GND
GND	GND	R6	GND
GND	GND	R15	GND
GND	GND	U11	GND
GND	GND	V3	GND
GND	GND	V18	GND
GND	GND	W7	GND
GND	GND	W14	GND
GND	GND	Y1	GND
GND	GND	Y10	GND
GND	GND	Y20	GND
VCCAUX	DONE	V17	CONFIG
VCCAUX	PROG_B	C2	CONFIG
VCCAUX	TCK	D17	JTAG
VCCAUX	TDI	B3	JTAG
VCCAUX	TDO	B19	JTAG
VCCAUX	TMS	E17	JTAG
VCCAUX	VCCAUX	D11	VCCAUX
VCCAUX	VCCAUX	H12	VCCAUX
VCCAUX	VCCAUX	J7	VCCAUX
VCCAUX	VCCAUX	K4	VCCAUX
VCCAUX	VCCAUX	L17	VCCAUX
VCCAUX	VCCAUX	M14	VCCAUX
VCCAUX	VCCAUX	N9	VCCAUX
VCCAUX	VCCAUX	U10	VCCAUX
VCCINT	VCCINT	H9	VCCINT
VCCINT	VCCINT	H11	VCCINT
VCCINT	VCCINT	H13	VCCINT
VCCINT	VCCINT	J8	VCCINT
VCCINT	VCCINT	J10	VCCINT
VCCINT	VCCINT	J12	VCCINT
VCCINT	VCCINT	K9	VCCINT
VCCINT	VCCINT	K11	VCCINT
VCCINT	VCCINT	L10	VCCINT
VCCINT	VCCINT	L12	VCCINT
VCCINT	VCCINT	M9	VCCINT
VCCINT	VCCINT	M11	VCCINT
VCCINT	VCCINT	M13	VCCINT
VCCINT	VCCINT	N8	VCCINT
VCCINT	VCCINT	N10	VCCINT
VCCINT	VCCINT	N12	VCCINT



## User I/Os by Bank

Table 153 indicates how the 304 available user-I/O pins are distributed between the four I/O banks on the FG400 package.

Table 153: User I/Os Per Bank for the XC3S1200E and XC3S1600E in the FG400 Package

Package Edge	I/O Bank	Maximum I/O	All Possible I/O Pins by Type				
			I/O	INPUT	DUAL	VREF <sup>(1)</sup>	CLK <sup>(2)</sup>
Top	0	78	43	20	1	6	8
Right	1	74	35	12	21	6	0 <sup>(2)</sup>
Bottom	2	78	30	18	24	6	0 <sup>(2)</sup>
Left	3	74	48	12	0	6	8
<b>TOTAL</b>		<b>304</b>	<b>156</b>	<b>62</b>	<b>46</b>	<b>24</b>	<b>16</b>

### 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.

## Footprint Migration Differences

The XC3S1200E and XC3S1600E FPGAs have identical footprints in the FG400 package. Designs can migrate between the XC3S1200E and XC3S1600E FPGAs without further consideration.