E·XFL



Welcome to E-XFL.COM

Understanding <u>Embedded - FPGAs (Field</u> <u>Programmable Gate Array)</u>

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

Details	
Product Status	Active
Number of LABs/CLBs	2168
Number of Logic Elements/Cells	19512
Total RAM Bits	516096
Number of I/O	304
Number of Gates	1200000
Voltage - Supply	1.14V ~ 1.26V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	400-BGA
Supplier Device Package	400-FBGA (21x21)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xc3s1200e-4fg400c

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

Input Delay Functions

Each IOB has a programmable delay block that optionally delays the input signal. In Figure 6, the signal path has a coarse delay element that can be bypassed. The input signal then feeds a 6-tap delay line. The coarse and tap delays vary; refer to timing reports for specific delay values. All six taps are available via a multiplexer for use as an asynchronous input directly into the FPGA fabric. In this way, the delay is programmable in 12 steps. Three of the six taps are also available via a multiplexer to the D inputs of the synchronous storage elements. The delay inserted in the path to the storage element can be varied in six steps. The first, coarse delay element is common to both asynchronous and synchronous paths, and must be either used or not used for both paths.

The delay values are set up in the silicon once at configuration time—they are non-modifiable in device operation.

The primary use for the input delay element is to adjust the input delay path to ensure that there is no hold time requirement when using the input flip-flop(s) with a global clock. The default value is chosen automatically by the Xilinx software tools as the value depends on device size and the specific device edge where the flip-flop resides. The value set by the Xilinx ISE software is indicated in the Map report generated by the implementation tools, and the resulting effects on input timing are reported using the Timing Analyzer tool.

If the design uses a DCM in the clock path, then the delay element can be safely set to zero because the Delay-Locked Loop (DLL) compensation automatically ensures that there is still no input hold time requirement.

Both asynchronous and synchronous values can be modified, which is useful where extra delay is required on clock or data inputs, for example, in interfaces to various types of RAM.

These delay values are defined through the IBUF_DELAY_VALUE and the IFD_DELAY_VALUE parameters. The default IBUF_DELAY_VALUE is 0, bypassing the delay elements for the asynchronous input. The user can set this parameter to 0-12. The default IFD_DELAY_VALUE is AUTO. IBUF_DELAY_VALUE and IFD_DELAY_VALUE are independent for each input. If the same input pin uses both registered and non-registered input paths, both parameters can be used, but they must both be in the same half of the total delay (both either bypassing or using the coarse delay element).



Figure 6: Programmable Fixed Input Delay Elements

Table 7: Differential IOSTANDARD Bank Compatibility

Differential	V _{CCO} Supply Input				
IOSTANDARD	1.8V	2.5V	3.3V	Requirements: V _{REF}	Restriction ⁽¹⁾
LVDS_25	Input	Input, On-chip Differential Termination, Output	Input	V _{REF} is not used for these I/O standards	Applies to Outputs Only
RSDS_25	Input	Input, On-chip Differential Termination, Output	Input		Applies to Outputs Only
MINI_LVDS_25	Input	Input, On-chip Differential Termination, Output	Input		Applies to Outputs Only
LVPECL_25	Input	Input	Input		
BLVDS_25	Input	Input, Output	Input		
DIFF_HSTL_I_18	Input, Output	Input	Input		No Differential Bank Restriction
DIFF_HSTL_III_18	Input, Output	Input	Input		(other I/O bank restrictions might
DIFF_SSTL18_I Input, Output	Input, Output	Input	Input		appiy)
DIFF_SSTL2_I	Input	Input, Output	Input		

Notes:

Each bank can support any two of the following: LVDS_25 outputs, MINI_LVDS_25 outputs, RSDS_25 outputs. 1.

HSTL and SSTL inputs use the Reference Voltage (V_{REF}) to bias the input-switching threshold. Once a configuration data file is loaded into the FPGA that calls for the I/Os of a given bank to use HSTL/SSTL, a few specifically reserved I/O pins on the same bank automatically convert to V_{BFF} inputs. For banks that do not contain HSTL or SSTL, V_{RFF} pins remain available for user I/Os or input pins.

Differential standards employ a pair of signals, one the opposite polarity of the other. The noise canceling properties (for example, Common-Mode Rejection) of these standards permit exceptionally high data transfer rates. This subsection introduces the differential signaling capabilities of Spartan-3E devices.

Each device-package combination designates specific I/O pairs specially optimized to support differential standards. A unique L-number, part of the pin name, identifies the line-pairs associated with each bank (see Module 4, Pinout Descriptions). For each pair, the letters P and N designate the true and inverted lines, respectively. For example, the pin names IO_L43P_3 and IO_L43N_3 indicate the true and inverted lines comprising the line pair L43 on Bank 3.

V_{CCO} provides current to the outputs and additionally powers the On-Chip Differential Termination. V_{CCO} must be 2.5V when using the On-Chip Differential Termination. The V_{BFF} lines are not required for differential operation.

To further understand how to combine multiple IOSTANDARDs within a bank, refer to IOBs Organized into Banks, page 18.

On-Chip Differential Termination

Spartan-3E devices provide an on-chip \sim 120 Ω differential termination across the input differential receiver terminals. The on-chip input differential termination in Spartan-3E devices potentially eliminates the external 100Ω termination resistor commonly found in differential receiver circuits. Differential termination is used for LVDS, mini-LVDS, and RSDS as applications permit.

On-chip Differential Termination is available in banks with $V_{CCO} = 2.5V$ and is not supported on dedicated input pins. Set the DIFF_TERM attribute to TRUE to enable Differential Termination on a differential I/O pin pair.

The DIFF_TERM attribute uses the following syntax in the UCF file:

INST <1/0_BUFFER_INSTANTIATION_NAME> DIFF_TERM = "<TRUE/FALSE>";

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
х	SLICEL/M Bottom	Output	Combinatorial output
Y	SLICEL/M Top	Output	Combinatorial output
ХВ	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.
- 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.

- 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 <u>UG331</u>.

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 -131,072₁₀ to +131,071₁₀ 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.

FIXED Phase Shift Mode

The FIXED phase shift mode shifts the DCM outputs by a fixed amount (T_{PS}), controlled by the user-specified PHASE_SHIFT attribute. The PHASE_SHIFT value (shown as P in Figure 44) must be an integer ranging from –255 to +255. PHASE_SHIFT specifies a phase shift delay as a fraction of the T_{CLKIN}. The phase shift behavior is different between ISE 8.1, Service Pack 3 and prior software versions, as described below.

Design Note

Prior to ISE 8.1i, Service Pack 3, the FIXED phase shift feature operated differently than the Spartan-3 DCM design primitive and simulation model. Designs using software prior to ISE 8.1i, Service Pack 3 require recompilation using the latest ISE software release. The following Answer Record contains additional information:

http://www.xilinx.com/support/answers/23153.htm.

FIXED Phase Shift using ISE 8.1i, Service Pack 3 and later: See Equation 2. The value corresponds to a phase shift range of -360° to $+360^{\circ}$, which matches behavior of the Spartan-3 DCM design primitive and simulation model.

$$t_{PS} = \left(\frac{PHASESHIFT}{256}\right) \bullet T_{CLKIN} \qquad Eq 2$$

FIXED Phase Shift prior to ISE 8.1i, Service Pack 3: See Equation 3. The value corresponds to a phase shift range of -180° to $+180^{\circ}$ degrees, which is different from the Spartan-3 DCM design primitive and simulation model. Designs created prior to ISE 8.1i, Service Pack 3 must be recompiled using the most recent ISE development software.

$$t_{PS} = \left(\frac{PHASESHIFT}{512}\right) \bullet T_{CLKIN} \qquad Eq 3$$

When the PHASE_SHIFT value is zero, CLKFB and CLKIN are in phase, the same as when the PS unit is disabled. When the PHASE_SHIFT value is positive, the DCM outputs are shifted later in time with respect to CLKIN input. When the attribute value is negative, the DCM outputs are shifted earlier in time with respect to CLKIN.

Figure 44b illustrates the relationship between CLKFB and CLKIN in the Fixed Phase mode. In the Fixed Phase mode, the PSEN, PSCLK, and PSINCDEC inputs are not used and must be tied to GND.

Equation 2 or Equation 3 applies only to FIXED phase shift mode. The VARIABLE phase shift mode operates differently.



Figure 44: NONE and FIXED Phase Shifter Waveforms (ISE 8.1i, Service Pack 3 and later)

Figure 57, page 82 demonstrates how to configure multiple FPGAs with different configurations, all stored in a single SPI Flash. The diagram uses standard SPI Flash memories but the same general technique applies for Atmel DataFlash.



DS312-2_50a_082009

Figure 54: Atmel SPI-based DataFlash Configuration Interface

Voltage Compatibility

Available SPI Flash PROMs use a single 3.3V supply voltage. All of the FPGA's SPI Flash interface signals are within I/O Bank 2. Consequently, the FPGA's VCCO_2 supply voltage must also be 3.3V to match the SPI Flash PROM.

Power-On Precautions if 3.3V Supply is Last in Sequence

Spartan-3E FPGAs have a built-in power-on reset (POR) circuit, as shown in Figure 66, page 102. The FPGA waits

for its three power supplies — V_{CCINT} , V_{CCAUX} , and V_{CCO} to I/O Bank 2 (VCCO_2) — to reach their respective power-on thresholds before beginning the configuration process.

The SPI Flash PROM is powered by the same voltage supply feeding the FPGA's VCCO_2 voltage input, typically 3.3V. SPI Flash PROMs specify that they cannot be accessed until their V_{CC} supply reaches its minimum data sheet voltage, followed by an additional delay. For some devices, this additional delay is as little as 10 µs as shown in Table 56. For other vendors, this delay is as much as 20 ms.

Table 🗄	56: Example Minimum	Power-On to Select	Times for Various SI	PI Flash PROMs
---------	---------------------	--------------------	-----------------------------	----------------

Vondor	SPI Flash PROM	Data Sheet Minimu	Sheet Minimum Time from V _{CC} min to Select = Low			
vendor	Part Number	Symbol	Value	Units		
STMicroelectronics	M25Pxx	T _{VSL}	10	μs		
Spansion	S25FLxxxA	t _{PU}	10	ms		
NexFlash	NX25xx	T _{VSL}	10	μs		
Macronix	MX25Lxxxx	t _{VSL}	10	μs		
Silicon Storage Technology	SST25LFxx	T _{PU-READ}	10	μs		
Programmable Microelectronics Corporation	Pm25LVxxx	T _{VCS}	50	μs		
Atmel Corporation	AT45DBxxxD	t _{VCSL}	30	μs		
	AT45DBxxxB		20	ms		

In many systems, the 3.3V supply feeding the FPGA's VCCO_2 input is valid before the FPGA's other V_{CCINT} and V_{CCAUX} supplies, and consequently, there is no issue. However, if the 3.3V supply feeding the FPGA's VCCO_2 supply is last in the sequence, a potential race occurs between the FPGA and the SPI Flash PROM, as shown in Figure 55.





If the FPGA's V_{CCINT} and V_{CCAUX} supplies are already valid, then the FPGA waits for VCCO_2 to reach its minimum threshold voltage before starting configuration. This threshold voltage is labeled as V_{CCO2T} in Table 74 of Module 3 and ranges from approximately 0.4V to 1.0V, substantially lower than the SPI Flash PROM's minimum voltage. Once all three FPGA supplies reach their

respective Power On Reset (POR) thresholds, the FPGA starts the configuration process and begins initializing its internal configuration memory. Initialization requires approximately 1 ms (T_{POR} , minimum in Table 111 of Module 3, after which the FPGA de-asserts INIT_B, selects the SPI Flash PROM, and starts sending the appropriate read command. The SPI Flash PROM must be ready for

support byte-wide data. However, after configuration, the FPGA supports either x8 or x16 modes. In x16 mode, up to eight additional user I/O pins are required for the upper data bits, D[15:8].

Connecting a Spartan-3E FPGA to a x8/x16 Flash PROM is simple, but does require a precaution. Various Flash PROM vendors use slightly different interfaces to support both x8 and x16 modes. Some vendors (Intel, Micron, some STMicroelectronics devices) use a straightforward interface with pin naming that matches the FPGA connections. However, the PROM's A0 pin is wasted in x16 applications and a separate FPGA user-I/O pin is required for the D15 data line. Fortunately, the FPGA A0 pin is still available as a user I/O after configuration, even though it connects to the Flash PROM.

Table 63: FPGA Connections to Flash PROM with IO15/A-1 Pin

Other vendors (AMD, Atmel, Silicon Storage Technology, some STMicroelectronics devices) use a pin-efficient interface but change the function of one pin, called IO15/A-1, depending if the PROM is in x8 or x16 mode. In x8 mode, BYTE# = 0, this pin is the least-significant address line. The A0 address line selects the halfword location. The A-1 address line selects the byte location. When in x16 mode, BYTE# = 1, the IO15/A-1 pin becomes the most-significant data bit, D15 because byte addressing is not required in this mode. Check to see if the Flash PROM has a pin named "IO15/A-1" or "DQ15/A-1". If so, be careful to connect x8/x16 Flash PROMs correctly, as shown in Table 63. Also, remember that the D[14:8] data connections require FPGA user I/O pins but that the D15 data is already connected for the FPGA's A0 pin.

x16 Flash PROM Interface After

IO15/A-1 Pin FPGA Configuration	FPGA Pin
---------------------------------	----------

	1015/A-1 Pin	FPGA Configuration	FPGA Configuration
LDC2	BYTE#	Drive LDC2 Low or leave unconnected and tie PROM BYTE# input to GND	Drive LCD2 High
LDC1	OE#	Active-Low Flash PROM output-enable control	Active-Low Flash PROM output-enable control
LDC0	CS#	Active-Low Flash PROM chip-select control	Active-Low Flash PROM chip-select control
HDC	WE#	Flash PROM write-enable control	Flash PROM write-enable control
A[23:1]	A[n:0]	A[n:0]	A[n:0]
A0	IO15/A-1	IO15/A-1 is the least-significant address input	IO15/A-1 is the most-significant data line, IO15
D[7:0]	IO[7:0]	IO[7:0]	IO[7:0]
User I/O	Upper data lines IO[14:8] not required unless used as x16 Flash interface after configuration	Upper data lines IO[14:8] not required	IO[14:8]

Some x8/x16 Flash PROMs have a long setup time requirement on the BYTE# signal. For the FPGA to configure correctly, the PROM must be in x8 mode with BYTE# = 0 at power-on or when the FPGA's PROG_B pin is pulsed Low. If required, extend the BYTE# setup time for a 3.3V PROM using an external 680 Ω pull-down resistor on the FPGA's LDC2 pin or by delaying assertion of the CSI_B select input to the FPGA.

Daisy-Chaining

If the application requires multiple FPGAs with different configurations, then configure the FPGAs using a daisy chain, as shown in Figure 59. Use BPI mode (M[2:0] = <0:1:0> or <0:1:1>) for the FPGA connected to the parallel NOR Flash PROM and Slave Parallel mode (M[2:0] = <1:1:0>) for all downstream FPGAs in the daisy-chain. If there are more than two FPGAs in the chain, then last FPGA in the chain can be from any Xilinx FPGA family. However, all intermediate FPGAs located in the

chain between the first and last FPGAs must from either the Spartan-3E or Virtex®-5 FPGA families.

After the master FPGA—the FPGA on the left in the diagram—finishes loading its configuration data from the parallel Flash PROM, the master device continues generating addresses to the Flash PROM and asserts its CSO_B output Low, enabling the next FPGA in the daisy-chain. The next FPGA then receives parallel configuration data from the Flash PROM. The master FPGA's CCLK output synchronizes data capture.

If HSWAP = 1, an external $4.7k\Omega$ pull-up resistor must be added on the CSO_B pin. If HSWAP = 0, no external pull-up is necessary.

Design Note

BPI mode daisy chain software support is available starting in ISE 8.2i.

http://www.xilinx.com/support/answers/23061.htm

Stepping 0 Limitations when Reprogramming via JTAG if FPGA Set for BPI Configuration

The FPGA can always be reprogrammed via the JTAG port, regardless of the mode pin (M[2:0]) settings. However, Stepping 0 devices have a minor limitation. If a Stepping 0 FPGA is set to configure in BPI mode and the FPGA is attached to a parallel memory containing a valid FPGA configuration file, then subsequent reconfigurations using the JTAG port will fail. Potential workarounds include setting the mode pins for JTAG configuration (M[2:0] = <1:0:1>) or offsetting the initial memory location in Flash by 0x2000.

Stepping 1 devices fully support JTAG configuration even when the FPGA mode pins are set for BPI mode.



Figure 59: Daisy-Chaining from BPI Flash Mode

In-System Programming Support

 \bigcirc In a production application, the parallel Flash PROM is usually preprogrammed before it is mounted on the printed circuit board. In-system programming support is available from third-party boundary-scan tool vendors and from some third-party PROM programmers using a socket adapter with attached wires. To gain access to the parallel Flash signals, drive the FPGA's PROG_B input Low with an open-drain driver. This action places all FPGA I/O pins, including those attached to the parallel Flash, in high-impedance (Hi-Z). If the HSWAP input is Low, the I/Os have pull-up resistors to the V_{CCO} input on their respective I/O bank. The external programming hardware then has direct access to the parallel Flash pins. The programming access points are highlighted in the gray boxes in Figure 58 and Figure 59.

The FPGA itself can also be used as a parallel Flash PROM programmer during development and test phases. Initially, an FPGA-based programmer is downloaded into the FPGA via JTAG. Then the FPGA performs the Flash PROM programming algorithms and receives programming data from the host via the FPGA's JTAG interface. See the Embedded System Tools Reference Manual.

Dynamically Loading Multiple Configuration Images Using MultiBoot Option

For additional information, refer to the "Reconfiguration and MultiBoot" chapter in <u>UG332</u>.



Figure 62: Daisy-Chaining using Slave Parallel Mode

Slave Serial Mode

For additional information, refer to the "Slave Serial Mode" chapter in UG332.

In Slave Serial mode (M[2:0] = <1:1:1>), an external host such as a microprocessor or microcontroller writes serial configuration data into the FPGA, using the synchronous serial interface shown in Figure 63. The serial configuration data is presented on the FPGA's DIN input pin with sufficient setup time before each rising edge of the externally generated CCLK clock input. The intelligent host starts the configuration process by pulsing PROG_B and monitoring that the INIT_B pin goes High, indicating that the FPGA is ready to receive its first data. The host then continues supplying data and clock signals until either the DONE pin goes High, indicating a successful configuration, or until the INIT_B pin goes Low, indicating a configuration error. The configuration process requires more clock cycles than indicated from the configuration file size. Additional clocks are required during the FPGA's start-up sequence, especially if the FPGA is programmed to wait for selected Digital Clock Managers (DCMs) to lock to their respective clock inputs (see Start-Up, page 105).



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

W 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 <u>XAPP453</u>: *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:>) for all FPGAs in the daisy-chain. After the lead FPGA is filled with its configuration data, the lead

Maximum Bitstream Size for Daisy-Chains

The maximum bitstream length supported by Spartan-3E FPGAs in serial daisy-chains is 4,294,967,264 bits (4 Gbits), roughly equivalent to a daisy-chain with 720 XC3S1600E FPGAs. This is a limit only for serial daisy-chains where configuration data is passed via the FPGA's DOUT pin. There is no such limit for JTAG chains.

Configuration Sequence

For additional information including I/O behavior before and during configuration, refer to the "Sequence of Events" chapter in <u>UG332</u>.

The Spartan-3E configuration process is three-stage process that begins after the FPGA powers on (a POR event) or after the PROG_B input is asserted. Power-On Reset (POR) occurs after the V_{CCINT} , V_{CCAUX} , and the V_{CCO} Bank 2 supplies reach their respective input threshold levels. After either a POR or PROG_B event, the three-stage configuration process begins.

- 1. The FPGA clears (initializes) the internal configuration memory.
- 2. Configuration data is loaded into the internal memory.
- 3. The user-application is activated by a start-up process.

Figure 66 is a generalized block diagram of the Spartan-3E configuration logic, showing the interaction of different device inputs and Bitstream Generator (BitGen) options. A flow diagram for the configuration sequence of the Serial and Parallel modes appears in Figure 66. Figure 67 shows the Boundary-Scan or JTAG configuration sequence.

Initialization

Configuration automatically begins after power-on or after asserting the FPGA PROG_B pin, unless delayed using the FPGA's INIT_B pin. The FPGA holds the open-drain INIT_B signal Low while it clears its internal configuration memory. Externally holding the INIT_B pin Low forces the configuration sequencer to wait until INIT_B again goes High.

The FPGA signals when the memory-clearing phase is complete by releasing the open-drain INIT_B pin, allowing the pin to go High via the external pull-up resistor to VCCO_2.

Loading Configuration Data

After initialization, configuration data is written to the FPGA's internal memory. The FPGA holds the Global Set/Reset (GSR) signal active throughout configuration, holding all FPGA flip-flops in a reset state. The FPGA signals when the entire configuration process completes by releasing the DONE pin, allowing it to go High. The FPGA configuration sequence can also be initiated by asserting PROG_B. Once released, the FPGA begins clearing its internal configuration memory, and progresses through the remainder of the configuration process.

Table 81: DC Characteristics of User I/Os UsingSingle-Ended Standards

IOSTANDARD		Te Cond	st itions	Logic Charac	Level teristics
Attribute		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	AD Test Logic Level Conditions Characteristics		Test Logic Level Iditions Characteristics	
Attribute	l _{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 $\rm V_{OL}$ and $\rm V_{OH}$ limits apply for both the Fast and Slow slew attributes.
- Tested according to the relevant PCI specifications. For information on PCI IP solutions, see <u>www.xilinx.com/pci</u>. The PCIX IOSTANDARD is available and has equivalent characteristics but no PCI-X IP is supported.

Phase Shifter (PS)

Table 108: Recommended Operating Conditions for the PS in Variable Phase Mode

			Speed	Grade		
Symbol	Description	-	5		4	Units
		Min	Max	Min	Max	
Operating Frequency Ranges						
PSCLK_FREQ (F _{PSCLK})	Frequency for the PSCLK input	1	167	1	167	MHz
Input Pulse Requirements						
PSCLK_PULSE	PSCLK pulse width as a percentage of the PSCLK period	40%	60%	40%	60%	-

Table 109: Switching Characteristics for the PS in Variable Phase Mode

Symbol	Description	Equ	Units	
Phase Shifting Range				
MAX_STEPS ⁽²⁾	Maximum allowed number of DCM_DELAY_STEP steps for a given CLKIN clock period, where $T = CLKIN$ clock	CLKIN < 60 MHz	±[INTEGER(10 ● (T _{CLKIN} – 3 ns))]	steps
	double the effective clock period. ⁽³⁾	CLKIN ≥ 60 MHz	±[INTEGER(15 ● (T _{CLKIN} – 3 ns))]	steps
FINE_SHIFT_RANGE_MIN	Minimum guaranteed delay for variable phase shifting	±[MAX_STEPS • DCM_DELAY_STEP_MIN]		ns
FINE_SHIFT_RANGE_MAX	Maximum guaranteed delay for variable phase shifting	±[MAX_STEPS • DCM_DELAY_STEP_MAX]		ns

Notes:

1. The numbers in this table are based on the operating conditions set forth in Table 77 and Table 108.

- 2. The maximum variable phase shift range, MAX_STEPS, is only valid when the DCM is has no initial fixed phase shifting, i.e., the PHASE_SHIFT attribute is set to 0.
- 3. The DCM_DELAY_STEP values are provided at the bottom of Table 105.

Miscellaneous DCM Timing

Table 110: Miscellaneous DCM Timing

Symbol	Description	Min	Мах	Units
DCM_RST_PW_MIN ⁽¹⁾	Minimum duration of a RST pulse width	3	-	CLKIN cycles
DCM_RST_PW_MAX ⁽²⁾	Maximum duration of a RST pulse width	N/A	N/A	seconds
DCM_CONFIG_LAG_TIME ⁽³⁾	Maximum duration from V_{CCINT} applied to FPGA configuration successfully completed (DONE pin goes High) and clocks applied to DCM DLL	N/A	N/A	minutes

Notes:

1. This limit only applies to applications that use the DCM DLL outputs (CLK0, CLK90, CLK180, CLK270, CLK2X, CLK2X180, and CLKDV). The DCM DFS outputs (CLKFX, CLKFX180) are unaffected.

- 2. This specification is equivalent to the Virtex-4 DCM_RESET specification. This specification does not apply for Spartan-3E FPGAs.
- 3. This specification is equivalent to the Virtex-4 TCONFIG specification. This specification does not apply for Spartan-3E FPGAs.

Byte Peripheral Interface (BPI) Configuration Timing



Shaded values indicate specifications on attached parallel NOR Flash PROM.

DS312-3_08_032409

Figure 77: Waveforms for Byte-wide Peripheral Interface (BPI) Configuration (BPI-DN mode shown)

Table	120:	Timing f	or Byte-v	vide Periphera	I Interface	(BPI)	Configuration	Mode
-------	------	----------	-----------	----------------	-------------	-------	---------------	------

Symbol	Description		Minimum	Maximum	Units	
T _{CCLK1}	Initial CCLK clock period	See Table 112				
T _{CCLKn}	CCLK clock period after FPGA loads ConfigRate setting		See Table 112			
T _{MINIT}	Setup time on CSI_B, RDWR_B, and M[2:0] mode pins before the rising edge of INIT_B			-	ns	
T _{INITM}	Hold time on CSI_B, RDWR_B, and M[2:0] mode pins after the rising edge of INIT_B			-	ns	
T _{INITADDR}	Minimum period of initial A[23:0] address cycle; LDC[2:0] and HDC are asserted and valid	5	5	T _{CCLK1} cycles		
		BPI-DN: (M[2:0] = <0:1:1>)	2	2		
T _{CCO}	Address A[23:0] outputs valid after CCLK falling edge			See Table 116		
T _{DCC}	Setup time on D[7:0] data inputs before CCLK rising edge			See Table 116		
T _{CCD}	Hold time on D[7:0] data inputs after CCLK rising edge		See Table 116			

Table 121: Configuration Timing Requirements for Attached Parallel NOR Flash

Symbol	Description	Requirement	Units
T _{CE} (t _{ELQV})	Parallel NOR Flash PROM chip-select time	T _{CE} ≤ T _{INITADDR}	ns
T _{OE} (t _{GLQV})	Parallel NOR Flash PROM output-enable time	$T_{OE} \leq T_{INITADDR}$	ns
T _{ACC} (t _{AVQV})	Parallel NOR Flash PROM read access time	$T_{ACC} \le 0.5 T_{CCLKn(min)} - T_{CCO} - T_{DCC} - PCB$	ns
T _{BYTE} (t _{FLQV} , t _{FHQV})	For x8/x16 PROMs only: BYTE# to output valid time ⁽³⁾	$T_{BYTE} \leq T_{INITADDR}$	ns

Notes:

1. These requirements are for successful FPGA configuration in BPI mode, where the FPGA provides the CCLK frequency. The post

- configuration timing can be different to support the specific needs of the application loaded int o the FPGA and the resulting clock source.Subtract additional printed circuit board routing delay as required by the application.
- 3. The initial BYTE# timing can be extended using an external, appropriately sized pull-down resistor on the FPGA's LDC2 pin. The resistor value also depends on whether the FPGA's HSWAP pin is High or Low.

Table 122: MultiBoot Trigger (MBT) Timing

Symbol	Description	Minimum	Maximum	Units
T _{MBT}	MultiBoot Trigger (MBT) Low pulse width required to initiate MultiBoot reconfiguration	300	∞	ns

Notes:

1. MultiBoot re-configuration starts on the rising edge after MBT is Low for at least the prescribed minimum period.

User I/Os by Bank

Table 132 indicates how the 66 available user-I/O pins are distributed between the four I/O banks on the VQ100 package.

Table 132: User I/Os Per Bank for XC3S100E, XC3S250E, and XC3S500E in the VQ100 Package

Package	I/O Bank	Maximum I/O	All Possible I/O Pins by Type					
Edge			I/O	INPUT	DUAL	VREF ⁽¹⁾	CLK ⁽²⁾	
Тор	0	15	5	0	1	1	8	
Right	1	15	6	0	0	1	8	
Bottom	2	19	0	0	18	1	0 ⁽²⁾	
Left	3	17	5	1	2	1	8	
TOTAL		66	16	1	21	4	24	

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 production XC3S100E, XC3S250E, and XC3S500E FPGAs have identical footprints in the VQ100 package. Designs can migrate between the devices without further consideration.

Table 133: CP132 Package Pinout (Cont'd)

Bank	XC3S100E Pin Name	XC3S250E XC3S500E Pin Name	CP132 Ball	Туре
2	N.C. (♦)	IO_L08P_2/A23	N9	100E: N.C. Others: DUAL
2	N.C. (♦)	IO_L09N_2/A20	M10	100E: N.C. Others: DUAL
2	N.C. (♦)	IO_L09P_2/A21	N10	100E: N.C. Others: DUAL
2	IO_L10N_2/VS1/A18	IO_L10N_2/VS1/A18	M11	DUAL
2	IO_L10P_2/VS2/A19	IO_L10P_2/VS2/A19	N11	DUAL
2	IO_L11N_2/CCLK	IO_L11N_2/CCLK	N12	DUAL
2	IO_L11P_2/VS0/A17	IO_L11P_2/VS0/A17	P12	DUAL
2	IP/VREF_2	IP/VREF_2	N3	VREF
2	IP_L05N_2/M2/GCLK1	IP_L05N_2/M2/GCLK1	N6	DUAL/GCLK
2	IP_L05P_2/RDWR_B/GCLK0	IP_L05P_2/RDWR_B/GCLK0	M6	DUAL/GCLK
2	VCCO_2	VCCO_2	M8	VCCO
2	VCCO_2	VCCO_2	P3	VCCO
3	Ю	Ю	J3	I/O
3	IP/VREF_3	IO/VREF_3	К3	100E: VREF(INPUT) Others: VREF(I/O)
3	IO_L01N_3	IO_L01N_3	B1	I/O
3	IO_L01P_3	IO_L01P_3	B2	I/O
3	IO_L02N_3	IO_L02N_3	C2	I/O
3	IO_L02P_3	IO_L02P_3	C3	I/O
3	N.C. (♦)	IO_L03N_3	D1	100E: N.C. Others: I/O
3	IO	IO_L03P_3	D2	I/O
3	IO_L04N_3/LHCLK1	IO_L04N_3/LHCLK1	F2	LHCLK
3	IO_L04P_3/LHCLK0	IO_L04P_3/LHCLK0	F3	LHCLK
3	IO_L05N_3/LHCLK3/IRDY2	IO_L05N_3/LHCLK3/IRDY2	G1	LHCLK
3	IO_L05P_3/LHCLK2	IO_L05P_3/LHCLK2	F1	LHCLK
3	IO_L06N_3/LHCLK5	IO_L06N_3/LHCLK5	H1	LHCLK
3	IO_L06P_3/LHCLK4/TRDY2	IO_L06P_3/LHCLK4/TRDY2	G3	LHCLK
3	IO_L07N_3/LHCLK7	IO_L07N_3/LHCLK7	H3	LHCLK
3	IO_L07P_3/LHCLK6	IO_L07P_3/LHCLK6	H2	LHCLK
3	IO_L08N_3	IO_L08N_3	L2	I/O
3	IO_L08P_3	IO_L08P_3	L1	I/O
3	IO_L09N_3	IO_L09N_3	M1	I/O
3	IO_L09P_3	IO_L09P_3	L3	I/O
3	IP/VREF_3	IP/VREF_3	E2	VREF
3	VCCO_3	VCCO_3	E1	VCCO
3	VCCO_3	VCCO_3	J2	VCCO
GND	N.C. (GND)	GND	A4	GND
GND	GND	GND	A8	GND
GND	N.C. (GND)	GND	C1	GND
GND	GND	GND	C7	GND

www.xilinx.com

FT256: 256-ball Fine-pitch, Thin Ball Grid Array

The 256-ball fine-pitch, thin ball grid array package, FT256, supports three different Spartan-3E FPGAs, including the XC3S250E, the XC3S500E, and the XC3S1200E.

Table 143 lists all the package pins. They are sorted bybank number and then by pin name of the largest device.Pins that form a differential I/O pair appear together in thetable. The table also shows the pin number for each pin andthe pin type, as defined earlier.

The highlighted rows indicate pinout differences between the XC3S250E, the XC3S500E, and the XC3S1200E FPGAs. The XC3S250E has 18 unconnected balls, indicated as N.C. (No Connection) in Table 143 and with the black diamond character (\blacklozenge) in Table 143 and Figure 83.

If the table row is highlighted in tan, then this is an instance where an unconnected pin on the XC3S250E FPGA maps

to a VREF pin on the XC3S500E and XC3S1200E 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 XC3S250E 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 FT256 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 143: FT256 Package Pinout

Bank	XC3S250E Pin Name	XC3S500E Pin Name	XC3S1200E Pin Name	FT256 Ball	Туре
0	10	Ю	10	A7	I/O
0	10	Ю	Ю	A12	I/O
0	10	Ю	IO	B4	I/O
0	IP	IP	Ю	B6	250E: INPUT 500E: INPUT 1200E: I/O
0	IP	IP	Ю	B10	250E: INPUT 500E: INPUT 1200E: I/O
0	IO/VREF_0	IO/VREF_0	IO/VREF_0	D9	VREF
0	IO_L01N_0	IO_L01N_0	IO_L01N_0	A14	I/O
0	IO_L01P_0	IO_L01P_0	IO_L01P_0	B14	I/O
0	IO_L03N_0/VREF_0	IO_L03N_0/VREF_0	IO_L03N_0/VREF_0	A13	VREF
0	IO_L03P_0	IO_L03P_0	IO_L03P_0	B13	I/O
0	IO_L04N_0	IO_L04N_0	IO_L04N_0	E11	I/O
0	IO_L04P_0	IO_L04P_0	IO_L04P_0	D11	I/O
0	IO_L05N_0/VREF_0	IO_L05N_0/VREF_0	IO_L05N_0/VREF_0	B11	VREF
0	IO_L05P_0	IO_L05P_0	IO_L05P_0	C11	I/O
0	IO_L06N_0	IO_L06N_0	IO_L06N_0	E10	I/O
0	IO_L06P_0	IO_L06P_0	IO_L06P_0	D10	I/O
0	IO_L08N_0/GCLK5	IO_L08N_0/GCLK5	IO_L08N_0/GCLK5	F9	GCLK
0	IO_L08P_0/GCLK4	IO_L08P_0/GCLK4	IO_L08P_0/GCLK4	E9	GCLK
0	IO_L09N_0/GCLK7	IO_L09N_0/GCLK7	IO_L09N_0/GCLK7	A9	GCLK
0	IO_L09P_0/GCLK6	IO_L09P_0/GCLK6	IO_L09P_0/GCLK6	A10	GCLK
0	IO_L11N_0/GCLK11	IO_L11N_0/GCLK11	IO_L11N_0/GCLK11	D8	GCLK
0	IO_L11P_0/GCLK10	IO_L11P_0/GCLK10	IO_L11P_0/GCLK10	C8	GCLK
0	IO_L12N_0	IO_L12N_0	IO_L12N_0	F8	I/O

www.xilinx.com

Table 143: FT256 Package Pinout (Cont'd)

Bank	XC3S250E Pin Name	XC3S500E Pin Name	XC3S1200E Pin Name	FT256 Ball	Туре
VCCAUX	VCCAUX	VCCAUX	VCCAUX	A11	VCCAUX
VCCAUX	VCCAUX	VCCAUX	VCCAUX	F1	VCCAUX
VCCAUX	VCCAUX	VCCAUX	VCCAUX	F16	VCCAUX
VCCAUX	VCCAUX	VCCAUX	VCCAUX	L1	VCCAUX
VCCAUX	VCCAUX	VCCAUX	VCCAUX	L16	VCCAUX
VCCAUX	VCCAUX	VCCAUX	VCCAUX	T6	VCCAUX
VCCAUX	VCCAUX	VCCAUX	VCCAUX	T11	VCCAUX
VCCINT	VCCINT	VCCINT	VCCINT	D4	VCCINT
VCCINT	VCCINT	VCCINT	VCCINT	D13	VCCINT
VCCINT	VCCINT	VCCINT	VCCINT	E5	VCCINT
VCCINT	VCCINT	VCCINT	VCCINT	E12	VCCINT
VCCINT	VCCINT	VCCINT	VCCINT	M5	VCCINT
VCCINT	VCCINT	VCCINT	VCCINT	M12	VCCINT
VCCINT	VCCINT	VCCINT	VCCINT	N4	VCCINT
VCCINT	VCCINT	VCCINT	VCCINT	N13	VCCINT