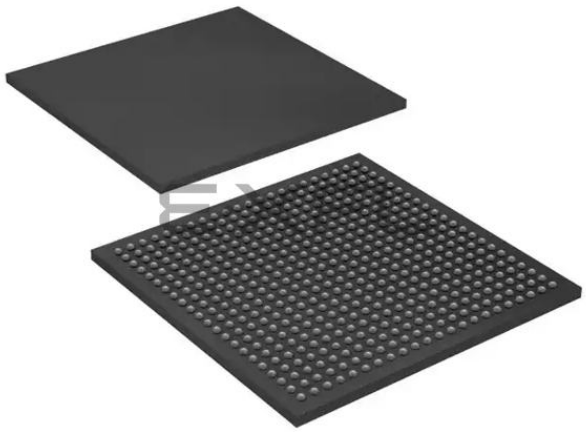


Welcome to E-XFL.COM

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

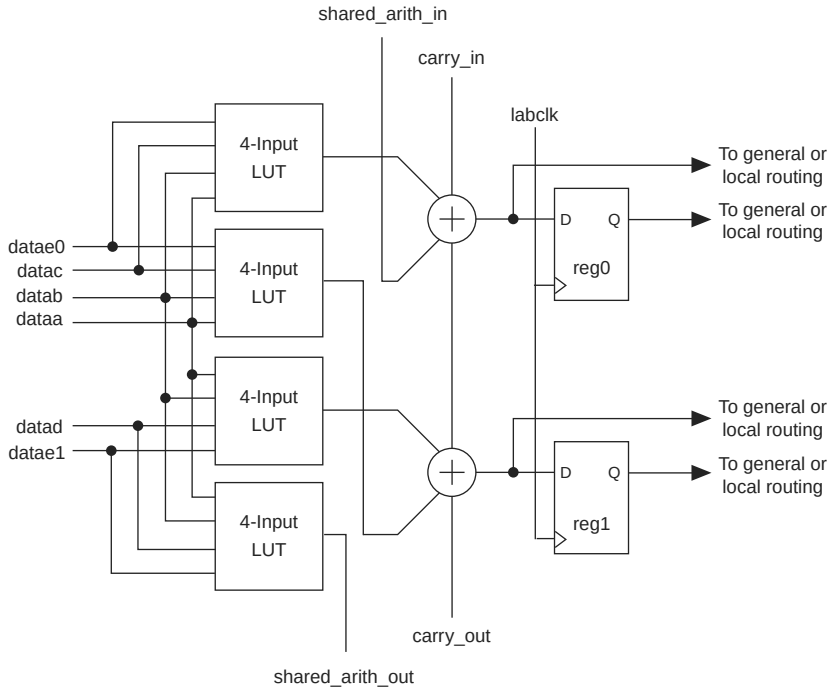
The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details	
Product Status	Obsolete
Number of LABs/CLBs	1900
Number of Logic Elements/Cells	47500
Total RAM Bits	5760000
Number of I/O	296
Number of Gates	-
Voltage - Supply	0.86V ~ 1.15V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	484-BBGA, FCBGA
Supplier Device Package	484-FBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep3se50f484c3n

Chapter 3. MultiTrack Interconnect in Stratix III Devices

Chapter 4. TriMatrix Embedded Memory Blocks in Stratix III Devices

Figure 2–13. ALM in Shared Arithmetic Mode

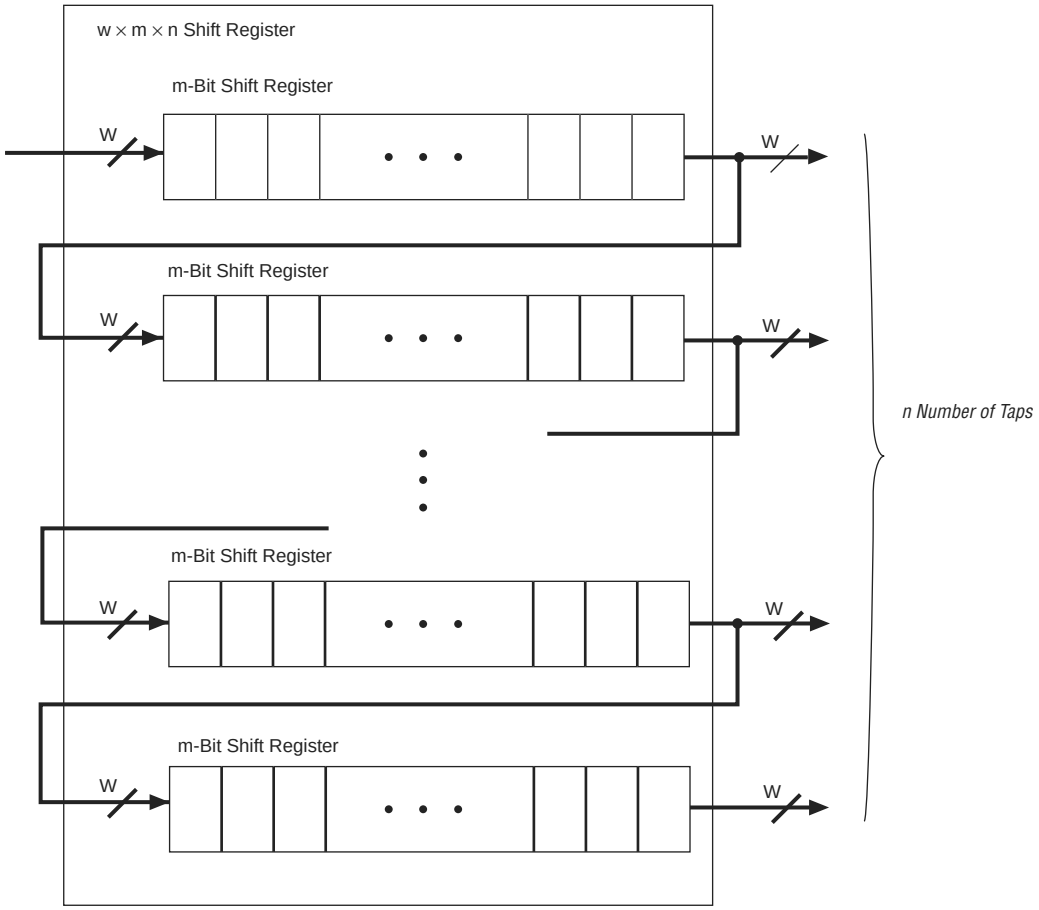


You can find adder trees in many different applications. For example, the summation of the partial products in a logic-based multiplier can be implemented in a tree structure. Another example is a correlator function that can use a large adder tree to sum filtered data samples in a given time frame to recover or to de-spread data that was transmitted utilizing spread spectrum technology.

An example of a three-bit add operation utilizing the shared arithmetic mode is shown in Figure 2–14. The partial sum (S[3..0]) and the partial carry (C[3..0]) is obtained using the LUTs, while the result (R[3..0]) is computed using the dedicated adders.

Figure 4–17 shows the TriMatrix memory block in shift-register mode.

Figure 4–17. Stratix III Shift-Register Memory Configuration



ROM Mode

All Stratix III TriMatrix memory blocks support ROM mode. A **.mif** file initializes the ROM content of these blocks. The address lines of the ROM are registered on M9K and M144K blocks, but can be unregistered on MLABs. The outputs can be registered or unregistered. Output registers can be asynchronously cleared. The ROM read operation is identical to the read operation in the single-port RAM configuration.

FIFO Mode

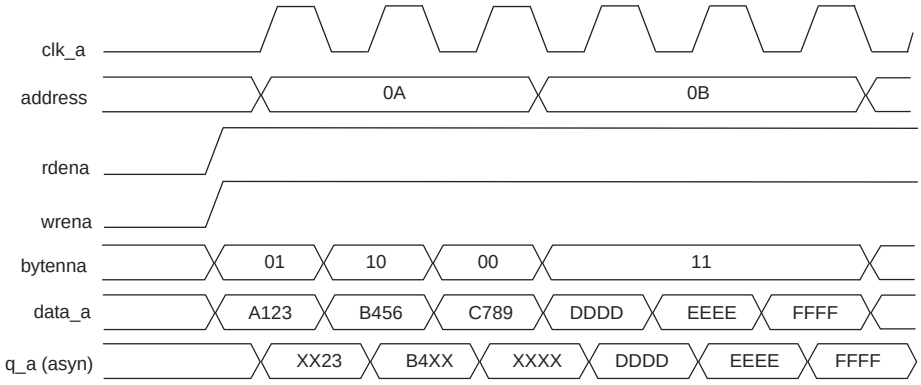
All TriMatrix memory blocks support FIFO mode. MLABs are ideal for designs with many small, shallow FIFO buffers. To implement FIFO buffers in your design, use the Quartus II software FIFO MegaWizard Plug-In Manager. Both single and dual-clock (asynchronous) FIFOs are supported.

For more information about implementing FIFO buffers, refer to the *Single- and Dual-Clock FIFO Megafunctions User Guide*.

MLABs do not support mixed-width FIFO mode.

Figure 4–19 shows the sample functional waveforms of same-port read-during-write behavior with new data.

Figure 4–19. Same Port Read-During-Write: New Data Mode *(Note 1)*

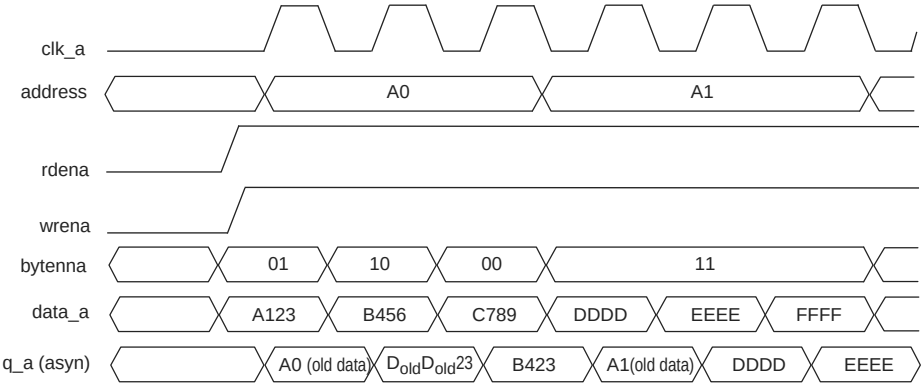


Note to Figure 4–19:

(1) "X" can be a don't care value or current data at that location, depending on the setting chosen in the Quartus II software.

Figure 4–20 shows the sample functional waveforms of same-port read-during-write behavior with old data mode.

Figure 4–20. Same Port Read-During-Write: Old Data Mode *(Note 1)*



Note to Figure 4–20:

(1) D_{old} is the old data bit at address A0, A0 (old data) is the old data at address A0, and A1 (old data) is the old data at address A1.

Mixed-Port Read-During-Write Mode

This mode applies to a RAM in simple or true dual-port mode which has one port reading and the other port writing to the same address location with the same clock.

In this mode you also have two output choices: old data or don't care. In old data mode, a read-during-write operation to different ports causes the RAM outputs to reflect the old data at that address location. In don't care mode, the same operation results in a "don't care" or "unknown" value on the RAM outputs.

Power Management

Stratix III memory block clock-enables allow you to control clocking of each memory block to reduce AC power consumption. Use the read-enable signal to ensure that read operations only occur when you need them to. If your design does not require read-during-write, you can reduce your power consumption by de-asserting the read-enable signal during write operations, or any period when no memory operations occur.

The Quartus II software automatically places any unused memory blocks in low power mode to reduce static power.

Programming File Compatibility

Beginning with version 8.1, the Quartus II software supports the logic option **STRATIXIII_MRAM_COMPATIBILITY**. When this option is set to on, the Quartus II software will generate programming files compatible with both affected and fixed silicons (for write speed decrease in M144K blocks). The default setting for this option is on.

For the list of devices that is affected by the write speed decrease for M144K blocks refer to the *Stratix III Device Family Errata Sheet*.

To set the **STRATIXIII_MRAM_COMPATIBILITY** variable, enter the following line in the Quartus Settings File:

```
set_global_assignment –name STRATIXIII_MRAM_COMPATIBILITY ON
```

When targeting fixed silicon devices, set the **STRATIXIII_MRAM_COMPATIBILITY** variable to OFF. When the **STRATIXIII_MRAM_COMPATIBILITY** option is set to OFF, you will be able to achieve the higher F_{MAX} that is published for M144K blocks in fixed silicons and the programming files will only be compatible with fixed silicons. These programming files will not configure other silicon revisions. The nSTATUS pin will drive out low and configuration will fail.

Conclusion

The Stratix III TriMatrix embedded memory structure provides three different on-chip RAM block sizes to address your design needs. All memory blocks are fully customizable and can be cascaded to implement wider or deeper memories with minimal speed penalty.

You can independently configure each embedded memory block to be a single- or dual-port RAM, FIFO, ROM, or shift register via the Quartus II MegaWizard Plug-In Manager software.

Shift registers are useful in DSP functions such as FIR filters. When implementing 18 × 18 or smaller width multipliers, you do not need external logic to create the shift register chain because the input shift registers are internal to the DSP block. This implementation significantly reduces the logical element (LE) resources required, avoids routing congestion, and results in predictable timing.

The first multiplier in every half DSP block (top- and bottom-half) in Stratix III devices has a multiplexer for the first multiplier B-input (lower-leg input) register to select between general routing and loopback, as shown in Figure 5–6. In loopback mode, the most significant 18-bit registered outputs are connected as feedback to the multiplier input of the first top multiplier in each half DSP block. Loopback modes are used by recursive filters where the previous output is needed to compute the current output.

The loopback mode is described in detail in "Two-Multiplier Adder Sum Mode" on page 5–21.

Table 5–3 lists the input register modes for the DSP block.

Table 5–3. Input Register Modes

Register Input Mode *(1)*	9 × 9	12 × 12	18 × 18	36 × 36	Double
Parallel input	✓	✓	✓	✓	✓
Shift register input *(2)*	—	—	✓	—	—
Loopback input *(3)*	—	—	✓	—	—

Notes to Table 5–3:

(1) The multiplier operand input wordlengths are statically configured at compile time.
(2) Available only on the A-operand.
(3) Only one loopback input is allowed per Half-Block. See Figure 5–15 for details.

Multiplier and First-Stage Adder

The multiplier stage natively supports 9 × 9, 12 × 12, 18 × 18, or 36 × 36 multipliers. Other wordlengths are padded up to the nearest appropriate native wordlength; for example, 16 × 16 would be padded up to use 18 × 18. Refer to "Independent Multiplier Modes" on page 5–15 for more details. Depending on the data width of the multiplier, a single DSP block can perform many multiplications in parallel.

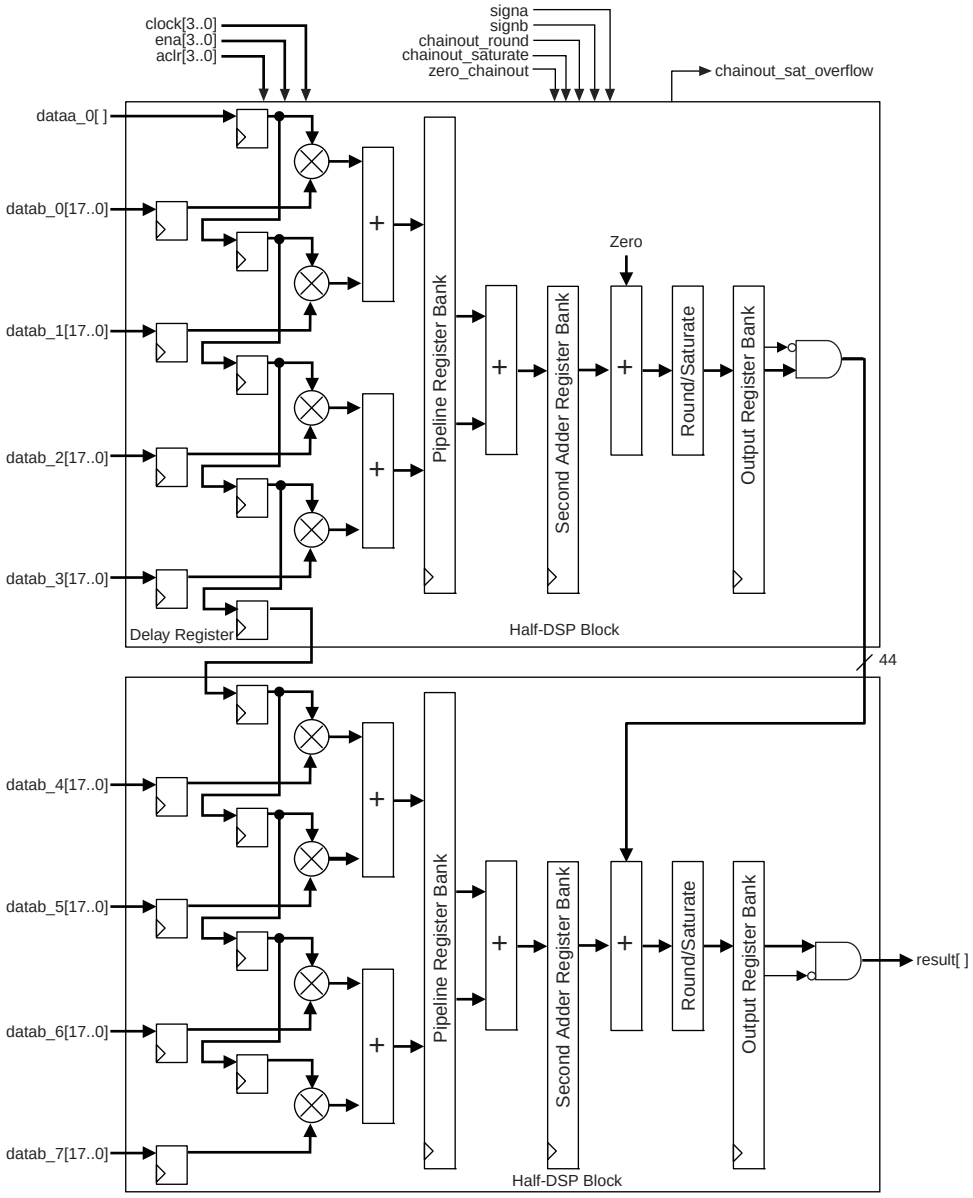
Each multiplier operand can be a unique signed or unsigned number. Two dynamic signals, `signa` and `signb`, control the representation of each operand, respectively. A `logic 1` value on the `signa/signb` signal indicates that `data A/data B` is a signed number; a `logic 0` value indicates an unsigned number. Table 5–4 lists the sign of the multiplication result for the various operand sign representations. The result of the multiplication is signed if any one of the operands is a signed value.

Table 5–4. Multiplier Sign Representation

Data A (signa Value)	Data B (signb Value)	Result
Unsigned (`logic 0`)	Unsigned (`logic 0`)	Unsigned
Unsigned (`logic 0`)	Signed (`logic 1`)	Signed
Signed (`logic 1`)	Unsigned (`logic 0`)	Signed
Signed (`logic 1`)	Signed (`logic 1`)	Signed

In Figure 5–23, the adder that adds the adjacent half DSP block to the current Four-Multiplier Adder is shown as the chainout adder for clarity. This scheme is used to chain and add multiple DSP blocks together. The output of the chainout adder can be registered. The registered chainout output can feed the lower adjacent DSP block for a chainout summation or it can feed general FPGA routing. The chainout result can be zeroed out by applying `logic 1` on the dynamic `zerochainout` signal. The `zerochainout` signal can also be registered.

Figure 5–23. FIR Filter using Tap-Delay Line Input and Chained Cascade Summation of Final Result



6. Clock Networks and PLLs in Stratix III Devices

SIII51006-2.0

This chapter describes the hierarchical clock networks and multiple phase-locked loops (PLLs) with advanced features in Stratix® III devices. The large number of clocking resources, in combination with the clock synthesis precision provided by the PLLs, provide a complete clock management solution. The Altera® Quartus® II software compiler automatically turns off clock networks not used in the design, thereby reducing the overall power consumption of the device.

Stratix III devices deliver abundant PLL resources with up to 12 PLLs per device and up to 10 outputs per PLL. You can independently program every output, creating a unique, customizable clock frequency. Inherent jitter filtration and fine granularity control over multiply, divide ratios, and dynamic phase shift reconfiguration provide the high performance precision required in today's high-speed applications. Stratix III device PLLs are feature-rich, supporting advanced capabilities such as clock switchover, dynamic phase shifting, PLL reconfiguration, and reconfigurable bandwidth. Stratix III PLLs also support external feedback mode, spread-spectrum tracking, and post-scale counter cascading features.

The Quartus II software enables the PLLs and their features without requiring any external devices. The following sections describe the Stratix III clock networks and PLLs in detail.

Clock Networks in Stratix III Devices

The global clock networks (GCLKs), regional clock networks (RCLKs), and periphery clock networks (PCLKs) available in Stratix III devices are organized into hierarchical clock structures that provide up to 220 unique clock domains (16 GCLKs + 88 RCLKs + 116 PCLKs) within the Stratix III device and allow up to 67 unique GCLK, RCLK, and PCLK clock sources (16 GCLKs + 22 RCLKs + 29 PCLKs) per device quadrant.

Table 6–1 lists the clock resources available in Stratix III devices.

Table 6–1. Clock Resources in Stratix III Devices (Part 1 of 2)

Clock Resource	# of Resources Available	Source of Clock Resource
Clock input pins	32 Single-ended (16 Differential)	`CLK[0..15]p` and `CLK[0..15]n` pins
Global clock networks	16	`CLK[0..15]p/n` pins, PLL clock outputs, and logic array
Regional clock networks	64/88 *(1)*	`CLK[0..15]p/n` pins, PLL clock outputs, and logic array
Peripheral clock networks	116 (29 per device quadrant) *(2)*	DPA clock outputs, horizontal I/O pins, and logic array
GCLKs/RCLKs per quadrant	32/38 *(3)*	16 GCLKs + 16 RCLKs/ 16 GCLKs + 22 RCLKs

Chapter Revision History

Table 6–23 lists the revision history for this chapter.

Table 6–23. Chapter Revision History (Part 1 of 2)

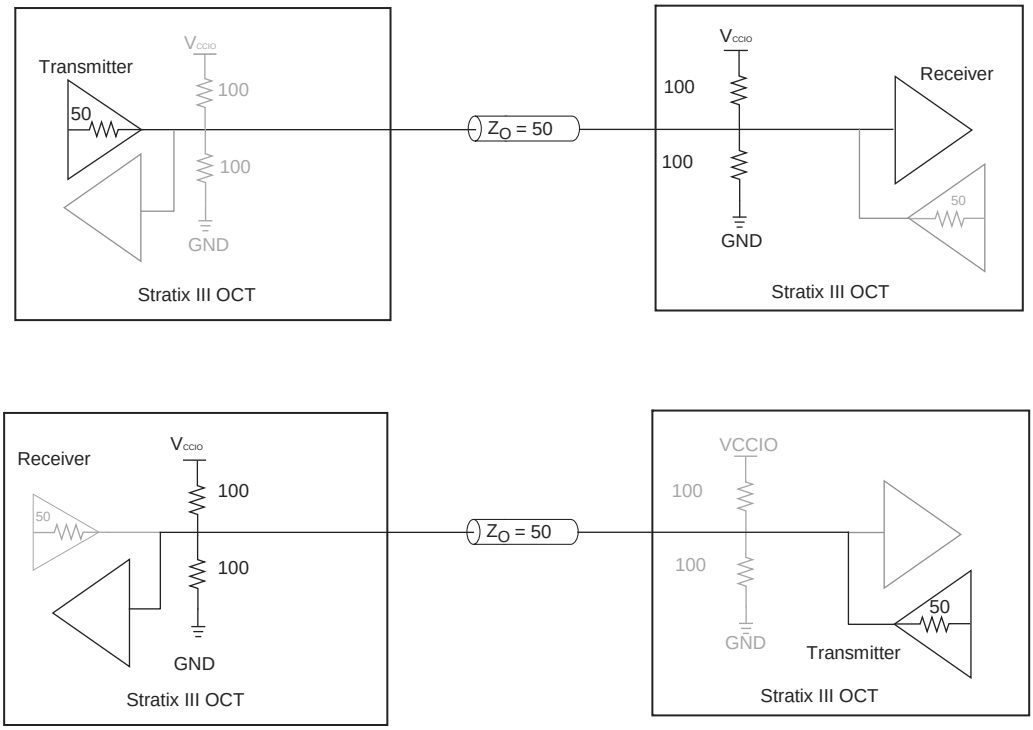
Date	Version	Changes Made
July 2010	2.0	Updated Figure 6–44.
March 2010	1.9	Updated for the Quartus II software version 9.1 SP2 release: ■ Updated Table 6–10 and Table 6–11. ■ Updated Figure 6–42. ■ Updated the "Guidelines" and "PLL Cascading and Clock Network Guidelines" sections. ■ Removed "sub-regional clock networks" information. ■ Minor text edits.
July 2009	1.8	■ Updated "Clock Switchover" section. ■ Updated Figure 6–37.
May 2009	1.7	■ Added "PLL and Clock Network Guidelines for External Memory Interface" and "Zero-Delay Buffer Mode" sections. ■ Updated Figure 6–17.
February 2009	1.6	■ Updated Table 6–7 and Table 6–10. ■ Updated Figure 6–23. ■ Updated "PLL Clock I/O Pins", "Logic Array Blocks (LABs)", and "Clock Feedback Modes" sections. ■ Removed "Reference Documents" section.
October 2008	1.5	■ Updated Table 6–10, Table 6–13, and Table 6–14. ■ Updated "locked", "Manual Override", "Bypassing PLL", "PLL Clock I/O Pins", and "Dynamic Phase-Shifting" sections. ■ Updated Figure 6–22, Figure 6–24, and Figure 6–26. ■ Updated (Note 2) to Figure 6–22. ■ Added (Note 3) to Table 6–14. ■ Added Figure 6–27. ■ Updated New Document Format.
May 2008	1.4	■ Updated Table 6–3, Table 6–4, Table 6–5, Table 6–6, Table 6–7, and Table 6–14. ■ Added new Figure 6–5 through Figure 6–9 to "Periphery Clock Networks" section. ■ Updated "Logic Array Blocks (LABs)", "External Feedback Mode", "Phase-Shift Implementation", and "Spread-Spectrum Tracking" sections. ■ Updated notes to Figure 6–17. ■ Updated notes to Figure 6–22. ■ Updated notes to Figure 6–27. ■ Updated Figure 6–43.
November 2007	1.3	Updated "pfdena" on page 6–42.

Dynamic OCT

Stratix III devices support on-off dynamic series and parallel termination for a bi-directional I/O in all I/O banks. Figure 7–11 shows the termination schemes supported in the Stratix III device. Dynamic parallel termination is enabled only when the bi-directional I/O acts as a receiver and is disabled when it acts as a driver. Similarly, dynamic series termination is enabled only when the bi-directional I/O acts as a driver and is disabled when it acts as a receiver. This feature is useful for terminating any high-performance bi-directional path because the signal integrity is optimized depending on the direction of the data.

You should connect a bi-directional pin that uses both 25-Ω or 50-Ω series termination and 50-Ω input termination to a calibration block that has a 50-Ω external resistor connected to its RUP and RDN pins. The 25-Ω series termination on the bi-directional pin is achieved through internal divide by two circuits.

Figure 7–11. Dynamic Parallel OCT in Stratix III Devices



For more information about tolerance specifications for OCT with calibration, refer to the *DC and Switching Characteristics of Stratix III Devices* chapter.

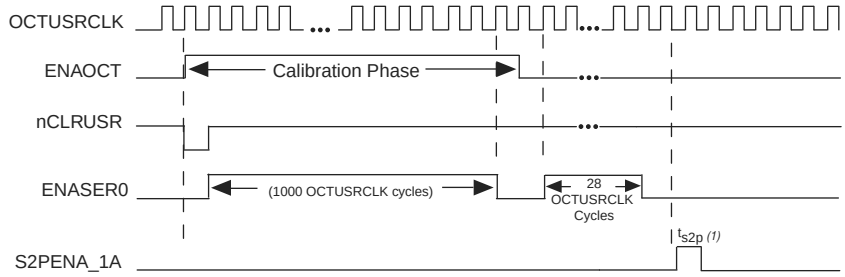
OCT Calibration

Figure 7–18 shows the user-mode signal-timing waveforms. To calibrate OCT block[N] (where N is a calibration block number), you must assert ENAOCT one cycle before asserting ENASER[N]. Also, nCLRUSR must be set to low for one OCTUSRCLK cycle before ENASER[N] signal is asserted. An asserted ENASER[N] signals for 1000 OCTUSRCLK cycles to perform $OCTR_S$ and $OCTR_T$ calibration. ENAOCT can be deasserted one clock cycle after the last ENASER is deasserted.

Serial Data Transfer

When calibration is complete, you must serially shift out the 28-bit OCT calibration code (14-bit OCT RS code and 14-bit OCT RT) from each OCT calibration block to the corresponding I/O buffers. Only one OCT calibration block can send out the codes at any given time by asserting only one ENASER[N] signal at a time. After ENAOCT is deasserted, you must wait at least 1 OCTUSRCLK cycle to enable any ENASER[N] signal to begin serial transfer. To shift 28-bit code from OCT calibration block[N], ENASER[N] must be asserted for exactly 28 OCTUSRCLK cycles. There must be at least one OCTUSRCLK cycle gap between two consecutive asserted ENASER signals. For these requirements, refer to Figure 7–18.

Figure 7–18. OCT User-Mode Signal Timing Waveform for One OCT Block



Note to Figure 7–18:

(1) ts2p ≥ 25 ns

After calibrated codes are shifted serially to the corresponding I/O buffers, they must be converted from serial format to parallel format before being used in the I/O buffers. Figure 7–18 shows S2PENA signals that can be asserted at any time to update the calibration codes in each I/O bank. All I/O banks that received the codes from the same OCT calibration block can have S2PENA asserted at the same time, or at a different time, even while another OCT calibration block is calibrating and serially shifting codes. The S2PENA signal is asserted one OCTUSRCLK cycle after ENASER is deasserted for at least 25 ns. You cannot use I/Os for transmitting or receiving data when their S2PENA is asserted for parallel codes transfer.

Example of Using Multiple OCT Calibration Blocks

Figure 7–19 shows a signal timing waveform for two OCT calibration blocks doing R_S and R_T calibration. Calibration blocks can start calibrating at different times by asserting ENASER signals at different times. ENAOCT must stay asserted while any calibration is ongoing. nCLRUSR must be set to low for one OCTUSRCLK cycle before each ENASER[N] signal is asserted. In Figure 7–19, when nCLRUSR is set to 0 for the second time to initialize OCT calibration block 0, this does not affect OCT calibration block 1, whose calibration is already in progress.

Figure 7–22. Differential SSTL I/O Standard Termination for Stratix III Devices

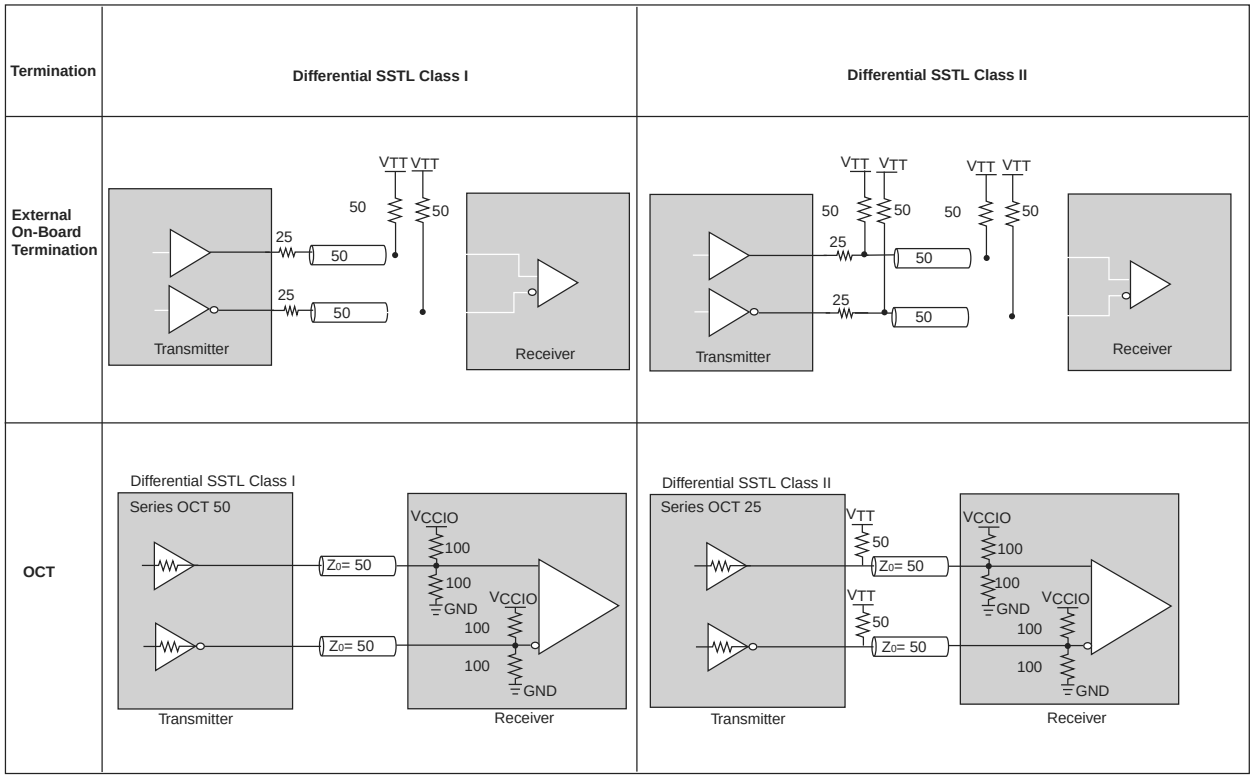
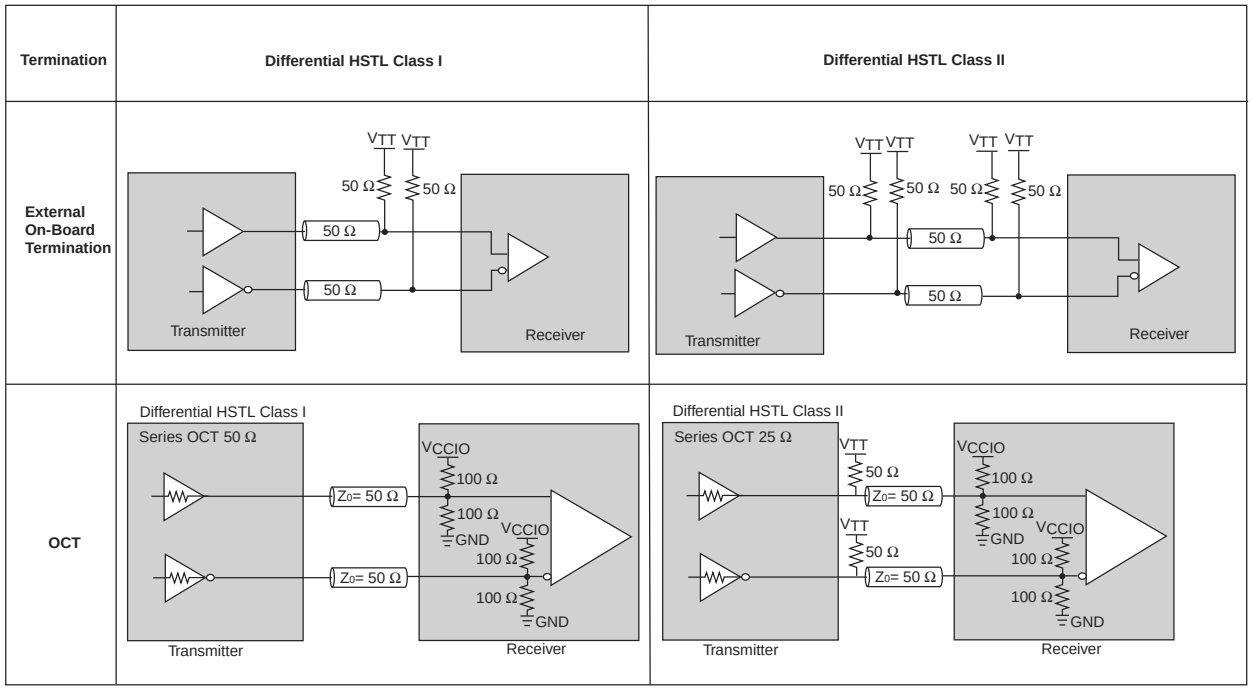
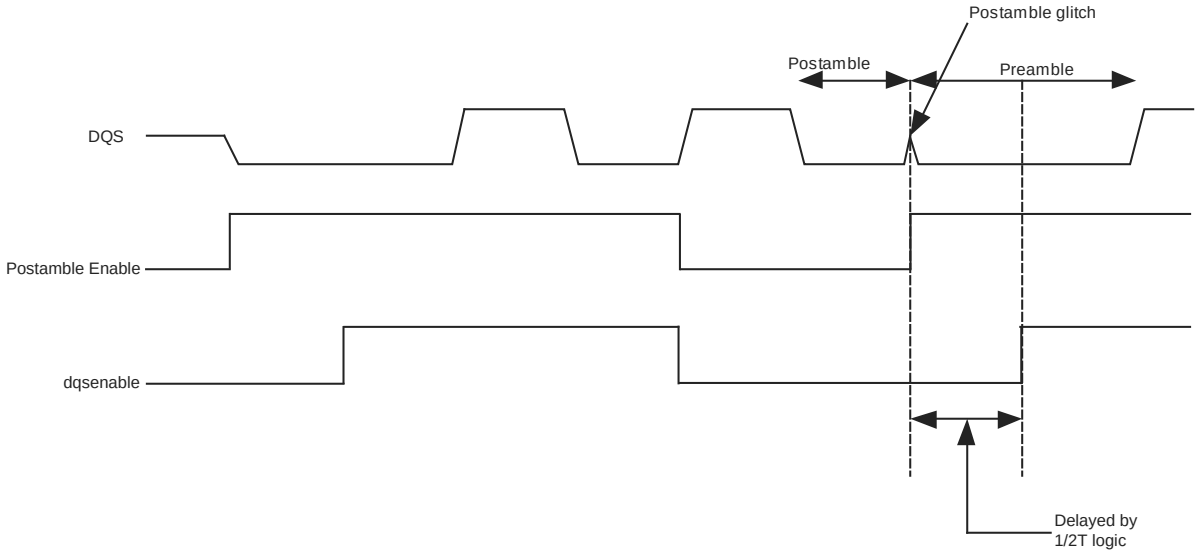


Figure 7–23. Differential HSTL I/O Standard Termination for Stratix III Devices



Using the HDR block as the first stage capture register in the postamble enable circuitry block is optional. The HDR block is clocked by the half-rate resynchronization clock, which is the output of the I/O clock divider circuit (shown in Figure 8–20 on page 8–35). There is an AND gate after the postamble register outputs that is used to avoid postamble glitches from a previous read burst on a non-consecutive read burst. This scheme allows a half-a-clock cycle latency for `dqsenable` assertion and zero latency for `dqsenable` deassertion, as shown in Figure 8–15.

Figure 8–15. Avoiding a Glitch on a Non-Consecutive Read Burst Waveform



Leveling Circuitry

DDR3 SDRAM unbuffered modules use a fly-by clock distribution topology for better signal integrity. This means that the CK/CK# signals arrive at each DDR3 SDRAM device in the module at different times. The difference in arrival time between the first DDR3 SDRAM device and the last device on the module can be as long as 1.6 ns. Figure 8–16 shows the clock topology in DDR3 SDRAM unbuffered modules.

If the configuration bitstream size exceeds the capacity of a serial configuration device, you must select a larger configuration device, enable the compression feature, or both. When configuring multiple devices, the size of the bitstream is the sum of the individual devices' configuration bitstreams.

A system may have multiple devices that contain the same configuration data. In active serial chains, you can implement this by storing one copy of the SOF in the serial configuration device. The same copy of the SOF configures the master Stratix III device and all remaining slave devices concurrently. All Stratix III devices must be the same density and package. The master device is set up in active serial mode and the slave devices are set up in passive serial mode.

To configure four identical Stratix III devices with the same SOF, you could set up the chain similar to the example shown in Figure 11–10. The first device is the master device, and its MSEL pins should be set to select AS configuration. The other three slave devices are set up for concurrent configuration, and their MSEL pins should be set to select PS configuration. The nCE input pins from the master and slave are connected to GND, and the DATA and DCLK pins connect in parallel to all four devices. During the configuration cycle, the master device reads its configuration data from the serial configuration device and transmits the second copy of the configuration data to all three slave devices, configuring all of them simultaneously.

Table 11–17. Chapter Revision History (Part 2 of 2)

Date	Version	Changes Made
May 2008	1.4	■ Updated "Power-On Reset Circuit" section. ■ Updated "Fast Active Serial Configuration (Serial Configuration Devices)" section. ■ Updated Table 11–4, Table 11–11, Table 11–12, Table 11–13, Table 11–14, and Table 11–16. ■ Updated Figure 11–3, Figure 11–4, Figure 11–5, Figure 11–6, Figure 11–7, Figure 11–8, Figure 11–9, Figure 11–10, Figure 11–11, Figure 11–13, Figure 11–14, Figure 11–15, Figure 11–17, Figure 11–18, Figure 11–19, Figure 11–20, and Figure 11–21.
November 2007	1.3	■ Updated Table 11–2. ■ Updated Figure 11–8, Figure 11–14, Figure 11–15, Figure 11–17, and Figure 11–18.
October 2007	1.2	■ Updated Table 11–13, Table 11–14. ■ Updated Figure 11–6, Figure 11–7, Figure 11–9, Figure 11–10, Figure 11–11, and Figure 11–13. ■ Removed text regarding enhanced configuration device support. Removed Figure 11–19. ■ Added live links for references. ■ Added section "Referenced Documents"
May 2007	1.1	Removed Bank Column from Table 11–13.
November 2006	1.0	Initial Release

Table 13–1 summarizes the functions of each of these pins.

Table 13–1. IEEE Std. 1149.1 Pin Descriptions

Pin	Description	Function
TDI	Test data input	Serial input pin for instructions as well as test and programming data. Signal applied to TDI is expected to change state at the falling edge of TCK. Data is shifted in on the rising edge of TCK.
TDO	Test data output	Serial data output pin for instructions as well as test and programming data. Data is shifted out on the falling edge of TCK. The pin is tri-stated if data is not being shifted out of the device.
TMS	Test mode select	Input pin that provides the control signal to determine the transitions of the test access port (TAP) controller state machine. Transitions within the state machine occur at the rising edge of TCK. Therefore, you must set up TMS before the rising edge of TCK. TMS is evaluated on the rising edge of TCK. During non-JTAG operation, Altera recommends you drive TMS high.
TCK	Test clock input	The clock input to the BST circuitry. Some operations occur at the rising edge, while others occur at the falling edge.
TRST *(1)*	Test reset input (optional)	Active-low input to asynchronously reset the boundary-scan circuit. For non-JTAG users, you should permanently tie the pin to GND.

Note to Table 13–1:

(1) The minimum TRST pulse width to reset the JTAG TAP controller is 60 ns.

The IEEE Std. 1149.1 BST circuitry requires the following registers:

- The instruction register determines the action to be performed and the data register to be accessed.
- The bypass register is a one-bit-long data register that provides a minimum-length serial path between TDI and TDO.
- The boundary-scan register is a shift register composed of all the boundary-scan cells of the device.

IEEE Std. 1149.1 BST Operation Control

Stratix III devices support the IEEE Std. 1149.1 (JTAG) instructions listed in Table 13–4.

Table 13–4. Stratix III JTAG Instructions

JTAG Instruction	Instruction Code	Description
SAMPLE / PRELOAD	00 0000 0101	Allows a snapshot of signals at the device pins to be captured and examined during normal device operation, and permits an initial data pattern to be output at the device pins. Also used by the SignalTap® II embedded logic analyzer.
EXTEST *(1)*	00 0000 1111	Allows the external circuitry and board-level interconnects to be tested by forcing a test pattern at the output pins and capturing test results at the input pins.
BYPASS	11 1111 1111	Places the 1-bit bypass register between the TDI and TDO pins, which allows the BST data to pass synchronously through selected devices to adjacent devices during normal device operation.
USERCODE	00 0000 0111	Places the 32-bit device identification register between TDI and TDO. The USERCODE value are loaded into this Device ID register for shifting out through TDO.
IDCODE	00 0000 0110	Selects the IDCODE register and places it between TDI and TDO, allowing the IDCODE to be serially shifted out of TDO.
HIGHZ *(1)*	00 0000 1011	Places the 1-bit bypass register between the TDI and TDO pins, which allows the BST data to pass synchronously through selected devices to adjacent devices during normal device operation, while tri-stating all of the I/O pins.
CLAMP *(1)*	00 0000 1010	Places the 1-bit bypass register between the TDI and TDO pins, which allows the BST data to pass synchronously through selected devices to adjacent devices during normal device operation while holding I/O pins to a state defined by the data in the boundary-scan register.
ICR instructions	—	Used when configuring a Stratix III device through the JTAG port with a USB-Blaster™, ByteBlaster™ II, MasterBlaster™ or ByteBlasterMV™ download cable, or when using a Jam File or Jam Byte-Code (JBC) File through an embedded processor.
PULSE_NCONFIG	00 0000 0001	Emulates pulsing the nCONFIG pin low to trigger reconfiguration even though the physical pin is unaffected.
CONFIG_IO	00 0000 1101	Allows I/O reconfiguration through JTAG ports using the IOCSR for JTAG testing. Can be executed before, after, or during configurations.

Note to Table 13–4:

(1) Bus hold and weak pull-up resistor features override the high-impedance state of the HIGHZ, CLAMP, and EXTEST instructions.

The IEEE Std. 1149.1 TAP controller, a 16-state machine clocked on the rising edge of TCK, uses the TMS pin to control IEEE Std. 1149.1 operation in the device. Figure 13–5 shows the TAP controller state machine.

HIGHZ Instruction Mode

The HIGHZ instruction mode sets all of the user I/O pins to an inactive drive state. These pins are tri-stated until a new JTAG instruction is executed. When this instruction is loaded into the instruction register, the bypass register is connected between the TDI and TDO ports.

I/O Voltage Support in JTAG Chain

The JTAG chain supports several devices. However, you should use caution if the chain contains devices that have different V_{CCIO} levels. The output voltage level of the TDO pin must meet the specifications of the TDI pin it drives. The TDI and TDO pins of Stratix III device are powered by the V_{CCPD} (2.5 V / 3.0 V / 3.3 V) of I/O Bank 1A. You should connect V_{CCPD} according to the I/O standard used in the same bank. For 3.3-V I/O standards, you should connect V_{CCPD} to 3.3 V. For 3.0-V I/O standards, you should connect V_{CCPD} to 3.0 V; for 2.5-V and below I/O standards, you should connect V_{CCPD} to 2.5 V. Table 13–6 lists board design recommendations to ensure proper JTAG chain operation.

Table 13–6. Supported TDO/TDI Voltage Combinations

Device	TDI Input Buffer Power	Stratix III TDO V_{CCPD}		
		V_{CCPD} = 3.3 V *(1)*	**V_{CCPD} = 3.0 V** *(1)*	**V_{CCPD} = 2.5 V** *(2)*
Stratix III	V_{CCPD} = 3.3V	✓	✓	✓
	V_{CCPD} = 3.0V	✓	✓	✓
	V_{CCPD} = 2.5V	✓	✓	✓
Non-Stratix III	V_{CC} = 3.3 V	✓ *(3)*	✓ *(4)*	✓ *(5)*
	V_{CC} = 2.5 V	✓ *(3)*	✓ *(4)*	✓ *(5)*
	V_{CC} = 1.8 V	✓ *(3)*	✓ *(4)*	✓ *(5)*
	V_{CC} = 1.5 V	✓ *(3)*	✓ *(4)*	✓ *(5)*

Notes to Table 13–6:

(1) The TDO output buffer meets V_{OH} (MIN) = 2.4 V.
(2) The TDO output buffer meets V_{OH} (MIN) = 2.0 V.
(3) Input buffer must be 3.3-V tolerant.
(4) Input buffer must be 3.0-V tolerant.
(5) Input buffer must be 2.5-V tolerant.

You can interface the TDI and TDO lines of the devices that have different V_{CCIO} levels by inserting a level shifter between the devices. If possible, you should build the JTAG chain in such a way that a device with a higher V_{CCIO} level drives to a device with an equal or lower V_{CCIO} level. This way, a level shifter is used only to shift the TDO level to a level acceptable to the JTAG tester. Figure 13–13 shows the JTAG chain of mixed voltages and how a level shifter is inserted in the chain.

Section V. Power and Thermal Management

This section provides information on Power and Thermal Management for the Stratix® III devices.

- Chapter 16, Programmable Power and Temperature-Sensing Diodes in Stratix III Devices

Revision History

Refer to each chapter for its own specific revision history. For information on when each chapter was updated, refer to the Chapter Revision Dates section, which appears in the full handbook.