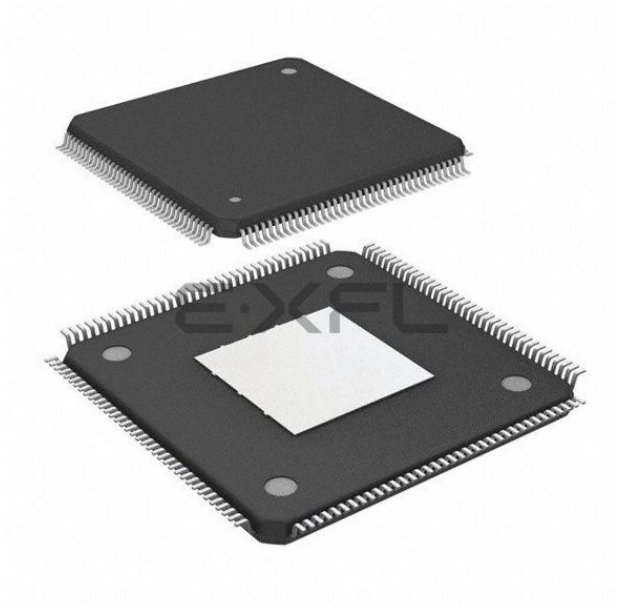


Welcome to E-XFL.COM

## Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

## Applications of Embedded - FPGAs

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

| Details | |
|---|---|
| Product Status | Active |
| Number of LABs/CLBs | 1395 |
| Number of Logic Elements/Cells | 22320 |
| Total RAM Bits | 608256 |
| Number of I/O | 79 |
| Number of Gates | - |
| Voltage - Supply | 0.97V ~ 1.03V |
| Mounting Type | Surface Mount |
| Operating Temperature | 0°C ~ 85°C (TJ) |
| Package / Case | 144-LQFP Exposed Pad |
| Supplier Device Package | 144-EQFP (20x20) |
| Purchase URL | https://www.e-xfl.com/product-detail/intel/ep4ce22e22c8ln |

This chapter provides additional information about the document and Altera.

## About this Handbook

This handbook provides comprehensive information about the Altera® Cyclone® IV family of devices.

## How to Contact Altera

To locate the most up-to-date information about Altera products, refer to the following table.

| Contact (1) | Contact Method | Address |
|---|---|---|
| Technical support | Website | www.altera.com/support |
| Technical training | Website | www.altera.com/training |
| | Email | custrain@altera.com |
| Product literature | Website | www.altera.com/literature |
| Nontechnical support (general) | Email | nacomp@altera.com |
| (software licensing) | Email | authorization@altera.com |

**Note to Table:**

(1) You can also contact your local Altera sales office or sales representative.

## Typographic Conventions

The following table shows the typographic conventions this document uses.

| Visual Cue | Meaning |
|---|---|
| **Bold Type with Initial Capital Letters** | Indicate command names, dialog box titles, dialog box options, and other GUI labels. For example, **Save As** dialog box. For GUI elements, capitalization matches the GUI. |
| **bold type** | Indicates directory names, project names, disk drive names, file names, file name extensions, software utility names, and GUI labels. For example, **\qdesigns** directory, **D:** drive, and **chiptrip.gdf** file. |
| *Italic Type with Initial Capital Letters* | Indicate document titles. For example, *Stratix IV Design Guidelines*. |
| *italic type* | Indicates variables. For example, *n* + 1.<br>Variable names are enclosed in angle brackets (< >). For example, *<file name>* and *<project name>***.pof** file. |
| Initial Capital Letters | Indicate keyboard keys and menu names. For example, the Delete key and the Options menu. |
| "Subheading Title" | Quotation marks indicate references to sections in a document and titles of Quartus II Help topics. For example, "Typographic Conventions." |

## I/O Features

Cyclone IV device I/O supports programmable bus hold, programmable pull-up resistors, programmable delay, programmable drive strength, programmable slew-rate control to optimize signal integrity, and hot socketing. Cyclone IV devices support calibrated on-chip series termination (Rs OCT) or driver impedance matching (Rs) for single-ended I/O standards. In Cyclone IV GX devices, the high-speed transceiver I/Os are located on the left side of the device. The top, bottom, and right sides can implement general-purpose user I/Os.

Table 1–8 lists the I/O standards that Cyclone IV devices support.

**Table 1–8. I/O Standards Support for the Cyclone IV Device Family**

| Type | I/O Standard |
|---|---|
| Single-Ended I/O | LVTTL, LVCMOS, SSTL, HSTL, PCI, and PCI-X |
| Differential I/O | SSTL, HSTL, LVPECL, BLVDS, LVDS, mini-LVDS, RSDS, and PPDS |

The LVDS SERDES is implemented in the core of the device using logic elements.

For more information, refer to the *I/O Features in Cyclone IV Devices* chapter.

## Clock Management

Cyclone IV devices include up to 30 global clock (GCLK) networks and up to eight PLLs with five outputs per PLL to provide robust clock management and synthesis. You can dynamically reconfigure Cyclone IV device PLLs in user mode to change the clock frequency or phase.

Cyclone IV GX devices support two types of PLLs: multipurpose PLLs and general-purpose PLLs:

- Use multipurpose PLLs for clocking the transceiver blocks. You can also use them for general-purpose clocking when they are not used for transceiver clocking.
- Use general purpose PLLs for general-purpose applications in the fabric and periphery, such as external memory interfaces. Some of the general purpose PLLs can support transceiver clocking.

For more information, refer to the *Clock Networks and PLLs in Cyclone IV Devices* chapter.

## External Memory Interfaces

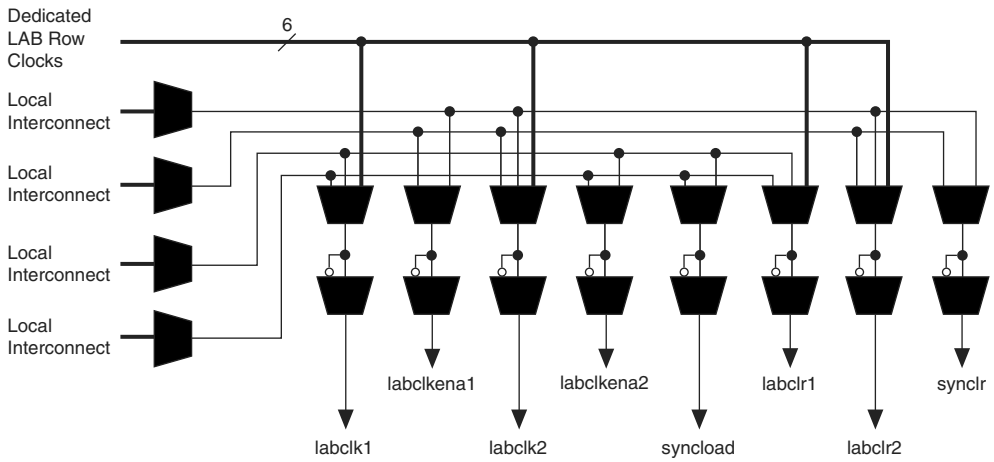
Cyclone IV devices support SDR, DDR, DDR2 SDRAM, and QDRII SRAM interfaces on the top, bottom, and right sides of the device. Cyclone IV E devices also support these interfaces on the left side of the device. Interfaces may span two or more sides of the device to allow more flexible board design. The Altera® DDR SDRAM memory interface solution consists of a PHY interface and a memory controller. Altera supplies the PHY IP and you can use it in conjunction with your own custom memory controller or an Altera-provided memory controller. Cyclone IV devices support the use of error correction coding (ECC) bits on DDR and DDR2 SDRAM interfaces.

Each LAB can use two clocks and two clock enable signals. The clock and clock enable signals of each LAB are linked. For example, any LE in a particular LAB using the `labclk1` signal also uses the `labclkena1`. If the LAB uses both the rising and falling edges of a clock, it also uses both LAB-wide clock signals. Deasserting the clock enable signal turns off the LAB-wide clock.

The LAB row clocks `[5..0]` and LAB local interconnect generate the LAB-wide control signals. The MultiTrack interconnect inherent low skew allows clock and control signal distribution in addition to data distribution.

Figure 2–6 shows the LAB control signal generation circuit.

**Figure 2–6. Cyclone IV Device LAB-Wide Control Signals**




LAB-wide signals control the logic for the clear signal of the register. The LE directly supports an asynchronous clear function. Each LAB supports up to two asynchronous clear signals (`labclr1` and `labclr2`).

A LAB-wide asynchronous load signal to control the logic for the preset signal of the register is not available. The register preset is achieved with a NOT gate push-back technique. Cyclone IV devices only support either a preset or asynchronous clear signal.

In addition to the clear port, Cyclone IV devices provide a chip-wide reset pin (`DEV_CLRn`) that resets all registers in the device. An option set before compilation in the Quartus II software controls this pin. This chip-wide reset overrides all other control signals.

# Document Revision History

Table 2–1 shows the revision history for this chapter.

**Table 2–1. Document Revision History**

| Date | Version | Changes |
|---|---|---|
| November 2009 | 1.0 | Initial release. |

# 3. Memory Blocks in Cyclone IV Devices

CYIV-51003-1.1

Cyclone® IV devices feature embedded memory structures to address the on-chip memory needs of Altera® Cyclone IV device designs. The embedded memory structure consists of columns of M9K memory blocks that you can configure to provide various memory functions, such as RAM, shift registers, ROM, and FIFO buffers.

This chapter contains the following sections:



## Overview

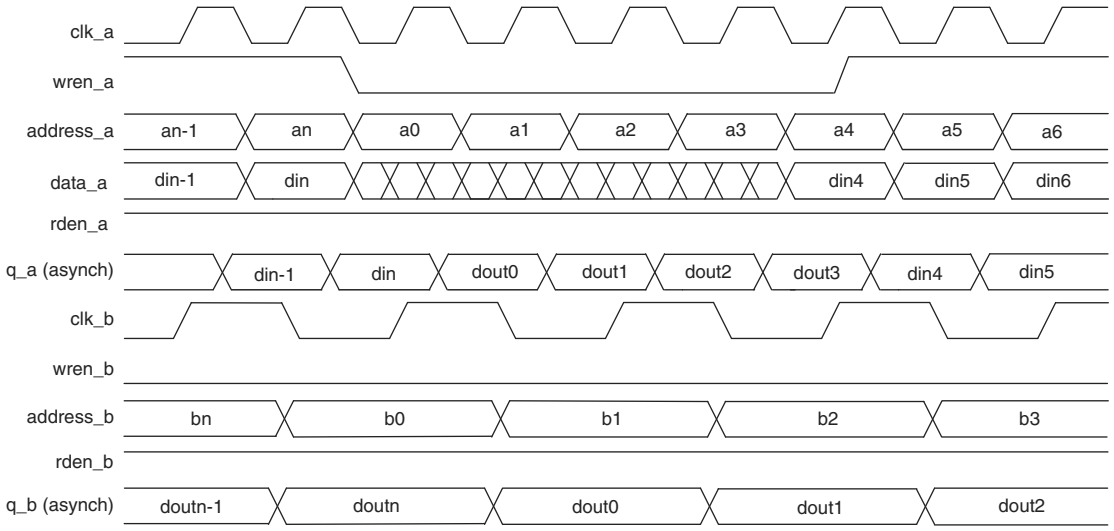
M9K blocks support the following features:

- 8,192 memory bits per block (9,216 bits per block including parity)
- Independent read-enable (`rden`) and write-enable (`wren`) signals for each port
- Packed mode in which the M9K memory block is split into two 4.5 K single-port RAMs
- Variable port configurations
- Single-port and simple dual-port modes support for all port widths
- True dual-port (one read and one write, two reads, or two writes) operation
- Byte enables for data input masking during writes
- Two clock-enable control signals for each port (port A and port B)
- Initialization file to pre-load memory content in RAM and ROM modes

In true dual-port mode, you can access any memory location at any time from either port A or port B. However, when accessing the same memory location from both ports, you must avoid possible write conflicts. When you attempt to write to the same address location from both ports at the same time, a write conflict happens. This results in unknown data being stored to that address location. There is no conflict resolution circuitry built into the Cyclone IV devices M9K memory blocks. You must handle address conflicts external to the RAM block.

Figure 3–11 shows true dual-port timing waveforms for the write operation at port A and read operation at port B. Registering the outputs of the RAM simply delays the `q` outputs by one clock cycle.

**Figure 3–11. Cyclone IV Devices True Dual-Port Timing Waveform**




## Shift Register Mode

Cyclone IV devices M9K memory blocks can implement shift registers for digital signal processing (DSP) applications, such as finite impulse response (FIR) filters, pseudo-random number generators, multi-channel filtering, and auto-correlation and cross-correlation functions. These and other DSP applications require local data storage, traditionally implemented with standard flipflops that quickly exhaust many logic cells for large shift registers. A more efficient alternative is to use embedded memory as a shift register block, which saves logic cell and routing resources.

The size of a ($w \times m \times n$) shift register is determined by the input data width ($w$), the length of the taps ($m$), and the number of taps ($n$), and must be less than or equal to the maximum number of memory bits, which is 9,216 bits. In addition, the size of ($w \times n$) must be less than or equal to the maximum width of the block, which is 36 bits. If you need a larger shift register, you can cascade the M9K memory blocks.

If you do not use dedicated clock pins to feed the GCLKs, you can use them as general-purpose input pins to feed the logic array. However, when using them as general-purpose input pins, they do not have support for an I/O register and must use LE-based registers in place of an I/O register.

For more information about how to connect the clock and PLL pins, refer to the *Cyclone IV Device Family Pin Connection Guidelines*.

## Clock Control Block

The clock control block drives the GCLKs. Clock control blocks are located on each side of the device, close to the dedicated clock input pins. GCLKs are optimized for minimum clock skew and delay.

Table 5–4 lists the sources that can feed the clock control block, which in turn feeds the GCLKs.

**Table 5–4. Clock Control Block Inputs**

| Input | Description |
|---|---|
| Dedicated clock inputs | Dedicated clock input pins can drive clocks or global signals, such as synchronous and asynchronous clears, presets, or clock enables onto given GCLKs. |
| Dual-purpose clock (DPCLK and CDPCLK) I/O input | DPCLK and CDPCLK I/O pins are bidirectional dual function pins that are used for high fan-out control signals, such as protocol signals, TRDY and IRDY signals for PCI, via the GCLK. Clock control blocks that have inputs driven by dual-purpose clock I/O pins are not able to drive PLL inputs. |
| PLL outputs | PLL counter outputs can drive the GCLK. |
| Internal logic | You can drive the GCLK through logic array routing to enable internal logic elements (LEs) to drive a high fan-out, low-skew signal path. Clock control blocks that have inputs driven by internal logic are not able to drive PLL inputs. |

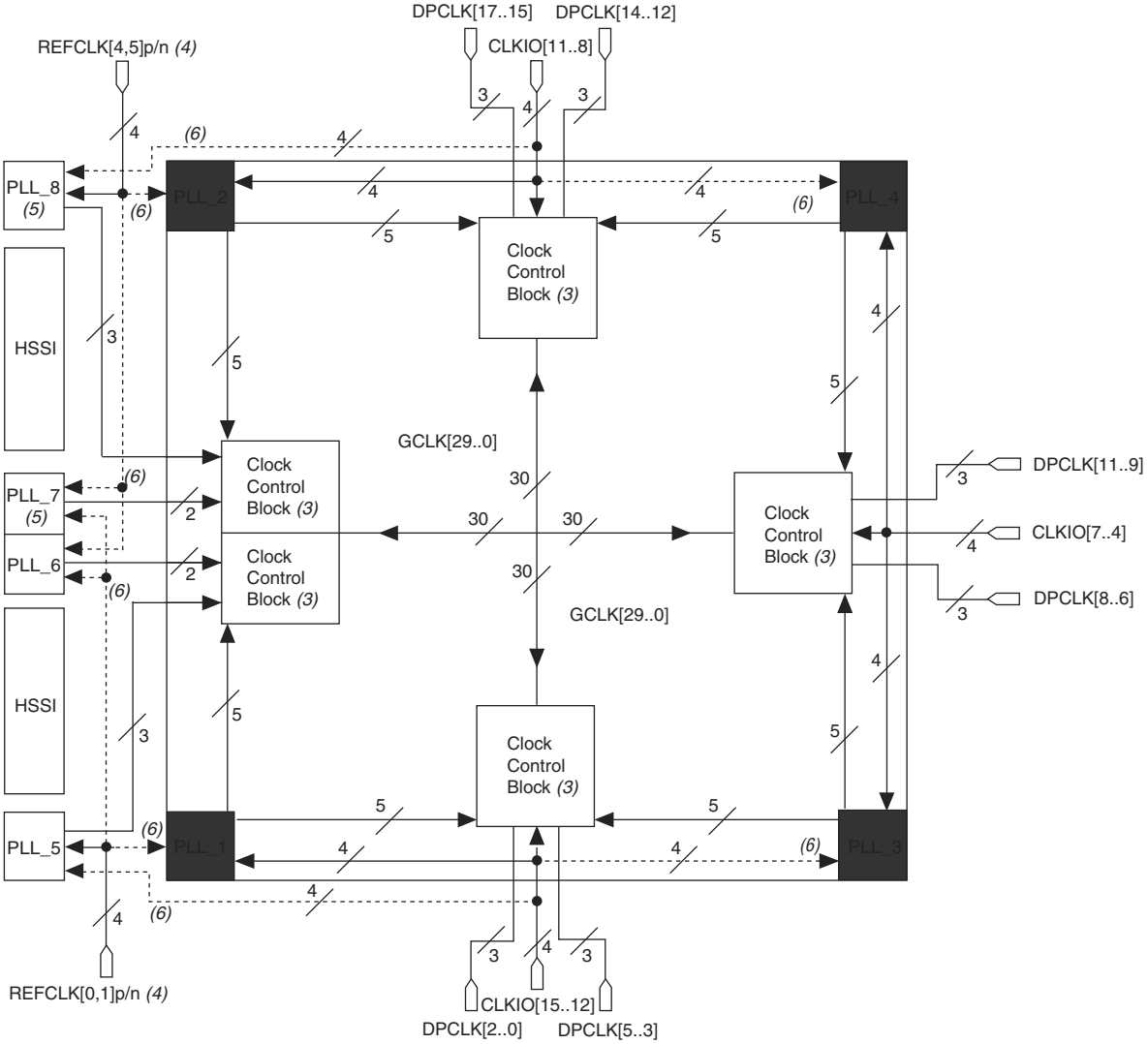
In Cyclone IV devices, dedicated clock input pins, PLL counter outputs, dual-purpose clock I/O inputs, and internal logic can all feed the clock control block for each GCLK. The output from the clock control block in turn feeds the corresponding GCLK. The GCLK can drive the PLL input if the clock control block inputs are outputs of another PLL or dedicated clock input pins. There are five or six clock control blocks on each side of the device periphery—depending on device density; providing up to 30 clock control blocks in each Cyclone IV GX device. The maximum number of clock control blocks per Cyclone IV E device is 20. For the clock control block locations, refer to Figure 5–2 on page 5–12, Figure 5–3 on page 5–13, and Figure 5–4 on page 5–14.

The clock control blocks on the left side of the Cyclone IV GX device do not support any clock inputs.

The control block has two functions:

- Dynamic GCLK clock source selection (not applicable for DPCLK, CDPCLK, and internal logic input)
- GCLK network power down (dynamic enable and disable)

**Figure 5–3. Clock Networks and Clock Control Block Locations in EP4CGX30, EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 Devices** *(1), (2)*




**Notes to Figure 5–3:**

(1) The clock networks and clock control block locations in this figure apply to only the EP4CGX30 device in F484 package and all EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 devices.

(2) `PLL_1`, `PLL_2`, `PLL_3`, and `PLL_4` are general purpose PLLs while `PLL_5`, `PLL_6`, `PLL_7`, and `PLL_8` are multipurpose PLLs.

(3) There are 6 clock control blocks on the top, right and bottom sides of the device and 12 clock control blocks on the left side of the device.

(4) `REFCLK[0,1]p/n` and `REFCLK[4,5]p/n` can only drive the general purpose PLLs and multipurpose PLLs on the left side of the device. These clock pins do not have access to the clock control blocks and GCLK networks. The `REFCLK[4,5]p/n` pins are not available in devices in F484 package.

(5) Not available for EP4CGX30, EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 devices in F484 package.

(6) Dedicated clock pins can feed into this PLL. However, these paths are not fully compensated.

**Table 5–5. Cyclone IV GX PLL Features (Part 2 of 2)**

| Features | Availability | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | General Purpose PLLs | | | | Multipurpose PLLs | | | | | |
| | PLL_1 (1), (10) | PLL_2 (1), (10) | PLL_3 (2) | PLL_4 (3) | PLL_1 (4) | PLL_2 (4) | PLL_5 (1), (10) | PLL_6 (1), (10) | PLL_7 (1) | PLL_8 (1) |
| Input clock switchover | ✓ | | | | | | | | | |
| User mode reconfiguration | ✓ | | | | | | | | | |
| Loss of lock detection | ✓ | | | | | | | | | |
| PLL drives TX Serial Clock, TX Load Enable, and TX Parallel Clock | ✓ | ✓ | — | — | ✓ | | | | | |
| VCO output drives RX clock data recovery (CDR) clock | — | | | | ✓ | | | | | |
| PLL drives FREF for ppm detect | ✓ | ✓ | — | — | ✓ | | | | | |

**Notes to Table 5–5:**

(1) This is only applicable to EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 devices in F672 and F896 package.

(2) This is applicable to all Cyclone IV devices.

(3) This is applicable to all Cyclone IV devices except EP4CGX15 devices in all packages, EP4CGX22, and EP4CGX30 devices in F169 package.

(4) This is only applicable to EP4CGX15, EP4CGX22, and all EP4CGX30 devices except EP4CGX30 in the F484 package..

(5) C counters range from 1 through 512 if the output clock uses a 50% duty cycle. For any output clocks using a non-50% duty cycle, the post-scale counters range from 1 through 256.

(6) These clock pins can access the GCLK networks.

(7) These clock pins are only available in EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 devices and cannot access the GCLK networks. CLK[17,19,20,21]p can be used as single-ended clock input pins.

(8) Only applicable if the input clock jitter is in the input jitter tolerance specifications.

(9) The smallest phase shift is determined by the voltage-controlled oscillator (VCO) period divided by eight. For degree increments, Cyclone IV GX devices can shift all output frequencies in increments of at least 45°. Smaller degree increments are possible depending on the frequency and divide parameters.

(10) This is applicable to the EP4CGX30, EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 devices in F484 package.

Table 5–6 lists the features available in Cyclone IV E PLLs.

**Table 5–6. Cyclone IV E PLL Features (Part 1 of 2)**

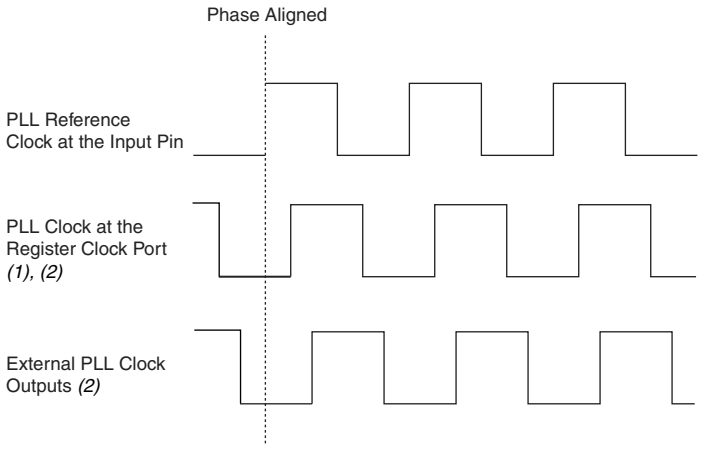
| Hardware Features | Availability |
|---|---|
| C (output counters) | 5 |
| M, N, C counter sizes | 1 to 512 (1) |
| Dedicated clock outputs | 1 single-ended or 1 differential pair |
| Clock input pins | 4 single-ended or 2 differential pairs |
| Spread-spectrum input clock tracking | ✓ (2) |
| PLL cascading | Through GCLK |
| Compensation modes | Source-Synchronous Mode, No Compensation Mode, Normal Mode, and Zero Delay Buffer Mode |
| Phase shift resolution | Down to 96-ps increments (3) |
| Programmable duty cycle | ✓ |
| Output counter cascading | ✓ |
| Input clock switchover | ✓ |
| User mode reconfiguration | ✓ |

## No Compensation Mode

In no compensation mode, the PLL does not compensate for any clock networks. This provides better jitter performance because clock feedback into the PFD does not pass through as much circuitry. Both the PLL internal and external clock outputs are phase shifted with respect to the PLL clock input.

Figure 5–13 shows a waveform example of the phase relationship of the PLL clock in this mode.

**Figure 5–13. Phase Relationship Between PLL Clocks in No Compensation Mode**




**Notes to Figure 5–13:**

(1) Internal clocks fed by the PLL are phase-aligned to each other.

(2) The PLL clock outputs can lead or lag the PLL input clocks.

## Normal Mode

An internal clock in normal mode is phase-aligned to the input clock pin. The external clock output pin has a phase delay relative to the clock input pin if connected in this mode. The Quartus II software timing analyzer reports any phase difference between the two. In normal mode, the PLL fully compensates the delay introduced by the GCLK network.

**Table 6–3. Cyclone IV Devices Supported I/O Standards and Constraints (Part 2 of 3)**

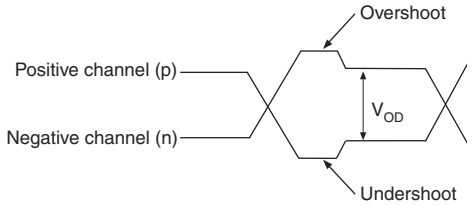
| I/O Standard | Type | Standard Support | $V_{CCIO}$ Level (in V) | | Column I/O Pins | | | Row I/O Pins (1) | |
|---|---|---|---|---|---|---|---|---|---|
| | | | Input | Output | CLK, DQS | PLL_OUT | User I/O Pins | CLK, DQS | User I/O Pins |
| 2.5-V LVTTL / LVCMOS | Single-ended | JESD8-5 | 3.3/3.0/2.5 (3) | 2.5 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 1.8-V LVTTL / LVCMOS | Single-ended | JESD8-7 | 1.8/1.5 (3) | 1.8 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 1.5-V LVCMOS | Single-ended | JESD8-11 | 1.8/1.5 (3) | 1.5 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 1.2-V LVCMOS (4) | Single-ended | JESD8-12A | 1.2 | 1.2 | ✓ | ✓ | ✓ | ✓ | ✓ |
| SSTL-2 Class I, SSTL-2 Class II | voltage-referenced | JESD8-9A | 2.5 | 2.5 | ✓ | ✓ | ✓ | ✓ | ✓ |
| SSTL-18 Class I, SSTL-18 Class II | voltage-referenced | JESD815 | 1.8 | 1.8 | ✓ | ✓ | ✓ | ✓ | ✓ |
| HSTL-18 Class I, HSTL-18 Class II | voltage-referenced | JESD8-6 | 1.8 | 1.8 | ✓ | ✓ | ✓ | ✓ | ✓ |
| HSTL-15 Class I, HSTL-15 Class II | voltage-referenced | JESD8-6 | 1.5 | 1.5 | ✓ | ✓ | ✓ | ✓ | ✓ |
| HSTL-12 Class I | voltage-referenced | JESD8-16A | 1.2 | 1.2 | ✓ | ✓ | ✓ | ✓ | ✓ |
| HSTL-12 Class II (9) | voltage-referenced | JESD8-16A | 1.2 | 1.2 | ✓ | ✓ | ✓ | — | — |
| PCI and PCI-X | Single-ended | — | 3.0 | 3.0 | ✓ | ✓ | ✓ | ✓ | ✓ |
| Differential SSTL-2 Class I or Class II | Differential (5) | JESD8-9A | — | 2.5 | — | ✓ | — | — | — |
| | | | 2.5 | — | ✓ | — | — | ✓ | — |
| Differential SSTL-18 Class I or Class II | Differential (5) | JESD815 | — | 1.8 | — | ✓ | — | — | — |
| | | | 1.8 | — | ✓ | — | — | ✓ | — |
| Differential HSTL-18 Class I or Class II | Differential (5) | JESD8-6 | — | 1.8 | — | ✓ | — | — | — |
| | | | 1.8 | — | ✓ | — | — | ✓ | — |
| Differential HSTL-15 Class I or Class II | Differential (5) | JESD8-6 | — | 1.5 | — | ✓ | — | — | — |
| | | | 1.5 | — | ✓ | — | — | ✓ | — |
| Differential HSTL-12 Class I or Class II | Differential (5) | JESD8-16A | — | 1.2 | — | ✓ | — | — | — |
| | | | 1.2 | — | ✓ | — | — | ✓ | — |
| PPDS (6) | Differential | — | — | 2.5 | — | ✓ | ✓ | — | ✓ |
| LVDS (10) | Differential | ANSI/TIA/EIA-644 | 2.5 | 2.5 | ✓ | ✓ | ✓ | ✓ | ✓ |
| RSDS and mini-LVDS (6) | Differential | — | — | 2.5 | — | ✓ | ✓ | — | ✓ |
| BLVDS (8) | Differential | — | 2.5 | 2.5 | — | — | ✓ | — | ✓ |

before the next edge; this may lead to pattern-dependent jitter. With pre-emphasis, the output current is momentarily boosted during switching to increase the output slew rate. The overshoot produced by this extra switching current is different from the overshoot caused by signal reflection. This overshoot happens only during switching, and does not produce ringing.

The Quartus II software allows two settings for programmable pre-emphasis control—**0** and **1**, in which **0** is pre-emphasis off and **1** is pre-emphasis on. The default setting is **1**. The amount of pre-emphasis needed depends on the amplification of the high-frequency components along the transmission line. You must adjust the setting to suit your designs, as pre-emphasis decreases the amplitude of the low-frequency component of the output signal.

Figure 6–20 shows the differential output signal with pre-emphasis.

**Figure 6–20. The Output Signal with Pre-Emphasis**




# High-Speed I/O Timing

This section discusses the timing budget, waveforms, and specifications for source-synchronous signaling in Cyclone IV devices. Timing for source-synchronous signaling is based on skew between the data and clock signals.

High-speed differential data transmission requires timing parameters provided by IC vendors and requires you to consider the board skew, cable skew, and clock jitter. This section provides information about high-speed I/O standards timing parameters in Cyclone IV devices.

Table 6–11 defines the parameters of the timing diagram shown in Figure 6–21.

**Table 6–11. High-Speed I/O Timing Definitions (Part 1 of 2)**

| Parameter | Symbol | Description |
|---|---|---|
| Transmitter channel-to-channel skew *(1)* | TCCS | The timing difference between the fastest and slowest output edges, including $t_{CO}$ variation and clock skew. The clock is included in the TCCS measurement. |
| Sampling window | SW | The period of time during which the data must be valid in order for you to capture it correctly. The setup and hold times determine the ideal strobe position in the sampling window. $T_{SW} = T_{SU} + T_{hd}$ + PLL jitter. |
| Time unit interval | TUI | The TUI is the data-bit timing budget allowed for skew, propagation delays, and data sampling window. |
| Receiver input skew margin | RSKM | RSKM is defined by the total margin left after accounting for the sampling window and TCCS. The RSKM equation is: $RSKM = \frac{(TUI - SW - TCCS}{2}$ |

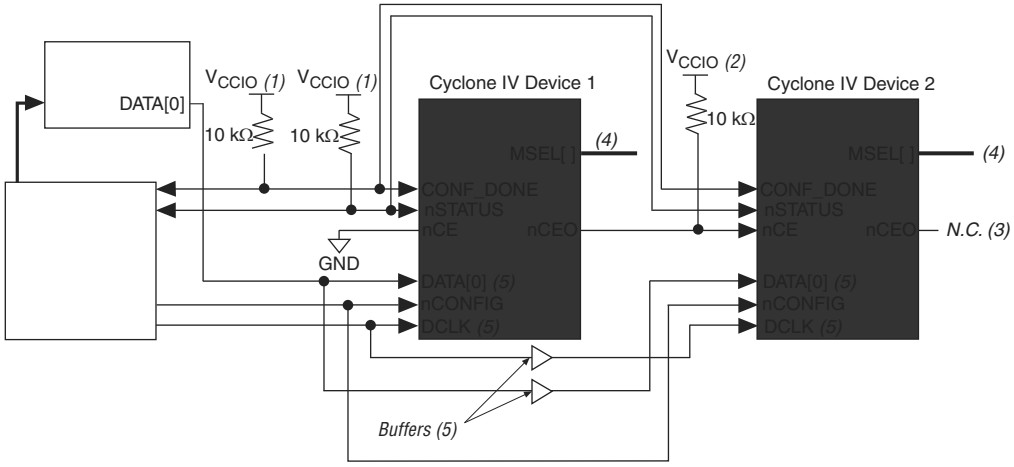
To ensure DCLK and DATA[0] are not left floating at the end of configuration, the MAX II device must drive them either high or low, whichever is convenient on your board. The DATA[0] pin is available as a user I/O pin after configuration. In the PS scheme, the DATA[0] pin is tri-stated by default in user mode and must be driven by the external host device. To change this default option in the Quartus II software, select the **Dual-Purpose Pins** tab of the **Device and Pin Options** dialog box.

The configuration clock (DCLK) speed must be below the specified system frequency to ensure correct configuration. No maximum DCLK period exists, which means you can pause configuration by halting DCLK for an indefinite amount of time.

The external host device can also monitor CONF_DONE and INIT_DONE to ensure successful configuration. The CONF_DONE pin must be monitored by the external device to detect errors and to determine when programming is complete. If all configuration data is sent, but CONF_DONE or INIT_DONE has not gone high, the external device must reconfigure the target device.

Figure 8–14 shows how to configure multiple devices using an external host device. This circuit is similar to the PS configuration circuit for a single device, except that Cyclone IV devices are cascaded for multi-device configuration.

**Figure 8–14. Multi-Device PS Configuration Using an External Host**




**Notes to Figure 8–14:**

(1) The pull-up resistor must be connected to a supply that provides an acceptable input signal for all devices in the chain. $V_{CC}$ must be high enough to meet the $V_{IH}$ specification of the I/O on the device and the external host.

(2) Connect the pull-up resistor to the $V_{CCIO}$ supply voltage of the I/O bank in which the nCE pin resides.

(3) The nCEO pin is left unconnected or used as a user I/O pin when it does not feed the nCE pin of another device.

(4) The MSEL pin settings vary for different configuration voltage standards and POR time. To connect the MSEL pins, refer to Table 8–3 on page 8–8, Table 8–4 on page 8–8, and Table 8–5 on page 8–9. Connect the MSEL pins directly to $V_{CCA}$ or GND.

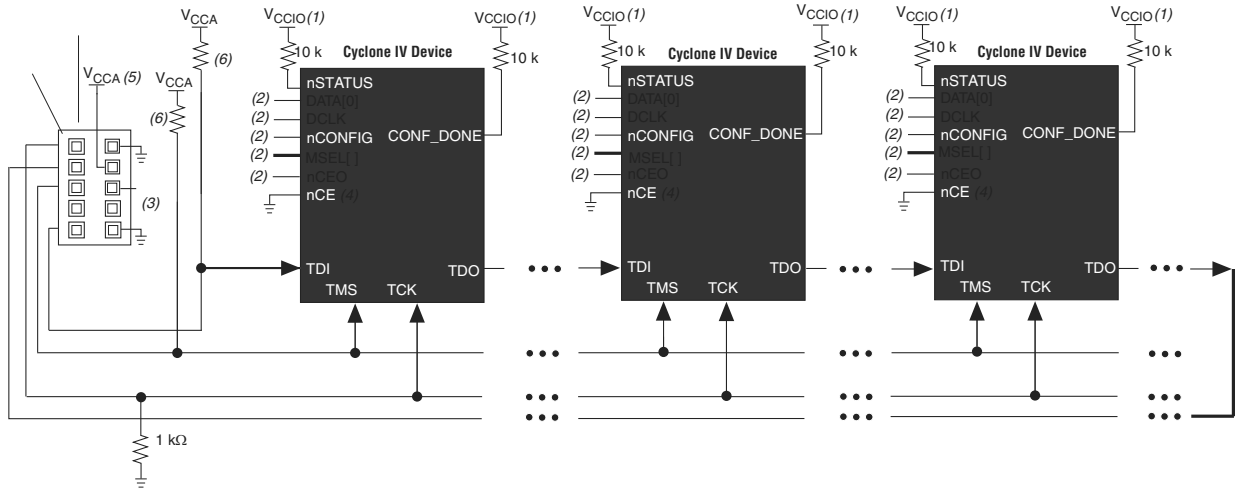
(5) All I/O inputs must maintain a maximum AC voltage of 4.1 V. DATA[0] and DCLK must fit the maximum overshoot outlined in Equation 8–1 on page 8–5.

When programming a JTAG device chain, one JTAG-compatible header is connected to several devices. The number of devices in the JTAG chain is limited only by the drive capability of the download cable. When four or more devices are connected in a JTAG chain, Altera recommends buffering the TCK, TDI, and TMS pins with an on-board buffer.

JTAG-chain device programming is ideal when the system contains multiple devices, or when testing your system with JTAG BST circuitry. Figure 8–25 and Figure 8–26 show multi-device JTAG configuration.

For devices using 2.5-, 3.0-, and 3.3-V $V_{CCIO}$ supply, you must refer to Figure 8–25. All I/O inputs must maintain a maximum AC voltage of 4.1 V because JTAG pins do not have the internal PCI clamping diodes to prevent voltage overshoot when using 2.5-, 3.0-, and 3.3- V $V_{CCIO}$ supply. You must power up the $V_{CC}$ of the download cable with a 2.5-V $V_{CCA}$ supply. For device using $V_{CCIO}$ of 1.2, 1.5 V, and 1.8 V, refer to Figure 8–26. You can power up the $V_{CC}$ of the download cable with the supply from $V_{CCIO}$.

**Figure 8–25. JTAG Configuration of Multiple Devices Using a Download Cable (2.5, 3.0, and 3.3-V $V_{CCIO}$ Powering the JTAG Pins)**




**Notes to Figure 8–25:**

(1) Connect these pull-up resistors to the $V_{CCIO}$ supply of the bank in which the pin resides.

(2) Connect the nCONFIG and MSEL pins to support a non-JTAG configuration scheme. If you only use a JTAG configuration, connect the nCONFIG pin to logic-high and the MSEL pins to GND. In addition, pull DCLK and DATA[0] to either high or low, whichever is convenient on your board.

(3) Pin 6 of the header is a $V_{IO}$ reference voltage for the MasterBlaster output driver. $V_{IO}$ must match the $V_{CCA}$ of the device. For this value, refer to the *MasterBlaster Serial/USB Communications Cable User Guide*. In the ByteBlasterMV cable, this pin is a no connect. In the USB-Blaster and ByteBlaster II cables, this pin is connected to nCE when it is used for AS programming, otherwise it is a no connect.

(4) You must connect the nCE pin to GND or driven low for successful JTAG configuration.

(5) Power up the $V_{CC}$ of the ByteBlaster II, USB-Blaster, or ByteBlasterMV cable with a 2.5-V supply from $V_{CCA}$. Third-party programmers must switch to 2.5 V. Pin 4 of the header is a $V_{CC}$ power supply for the MasterBlaster cable. The MasterBlaster cable can receive power from either 5.0- or 3.3-V circuit boards, DC power supply, or 5.0 V from the USB cable. For this value, refer to the *MasterBlaster Serial/USB Communications Cable User Guide*.

(6) Resistor value can vary from 1 kΩ to 10 kΩ.

The user watchdog timer is disabled in factory configurations and during the configuration cycle of the application configuration. It is enabled after the application configuration enters user mode.

## Quartus II Software Support

Implementation in your design requires a remote system upgrade interface between the Cyclone IV device logic array and remote system upgrade circuitry. You must also generate configuration files for production and remote programming of the system configuration memory. The Quartus II software provides these features.

The two implementation options, ALTREMOTE_UPDATE megafunction and remote system upgrade atom, are for the interface between the remote system upgrade circuitry and the device logic array interface. Using the megafunction block instead of creating your own logic saves design time and offers more efficient logic synthesis and device implementation.

For more information about the ALTREMOTE_UPDATE megafunction, refer to the *Remote Update Circuitry (ALTREMOTE_UPDATE) Megafunction User Guide*.

# Document Revision History

Table 8–28 lists the revision history for this chapter.

**Table 8–28. Document Revision History (Part 1 of 2)**

| Date | Version | Changes |
|---|---|---|
| May 2013 | 1.7 | ■ Added Table 8–6.<br>■ Updated Table 8–9 to add new device options and packages.<br>■ Updated Figure 8–16 and Figure 8–22 to include user mode.<br>■ Updated the “Dedicated” column for DATA[0] and DCLK in Table 8–19.<br>■ Updated the “User Mode” and “Pin Type” columns for DCLK in Table 8–20. |
| February 2013 | 1.6 | Updated Table 8–9 to add new device options and packages. |
| October 2012 | 1.5 | ■ Updated “AP Configuration Supported Flash Memories”, “Configuration Data Decompression”, and “Overriding the Internal Oscillator” sections.<br>■ Updated Figure 8–3, Figure 8–4, Figure 8–5, Figure 8–7, Figure 8–8, Figure 8–9, Figure 8–10, and Figure 8–11.<br>■ Updated Table 8–2, Table 8–8, Table 8–12, Table 8–13, Table 8–18, and Table 8–19. |
| November 2011 | 1.4 | ■ Added information about how to gain control of EPCS pins.<br>■ Updated “Reset”, “Single-Device AS Configuration”, “Single-Device AP Configuration”, and “Overriding the Internal Oscillator” sections.<br>■ Added Table 8–7.<br>■ Updated Table 8–6 and Table 8–19.<br>■ Updated Figure 8–3, Figure 8–4, and Figure 8–5. |
| December 2010 | 1.3 | ■ Updated for the Quartus II software version 10.1 release.<br>■ Added Cyclone IV E new device package information.<br>■ Updated Table 8–7, Table 8–10, and Table 8–11.<br>■ Minor text edits. |

When the byte serializer is enabled, the low-speed clock frequency is halved before feeding into the read clock of TX phase compensation FIFO. The low-speed clock is available in the FPGA fabric as `tx_clkout` port, which can be used in the FPGA fabric to send transmitter data and control signals.

**Figure 1–33. Transmitter Only Datapath Clocking in Non-Bonded Channel Configuration**

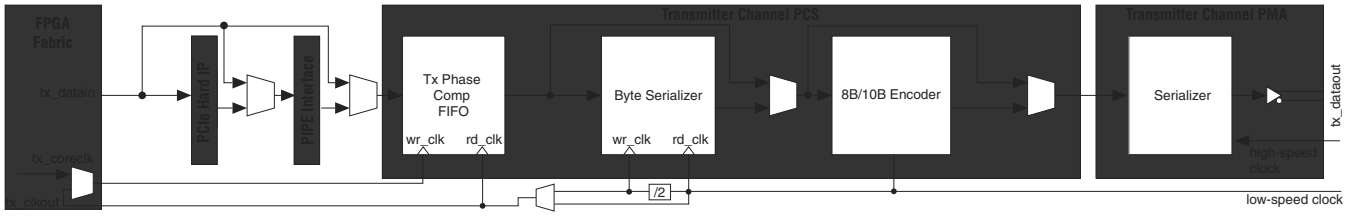
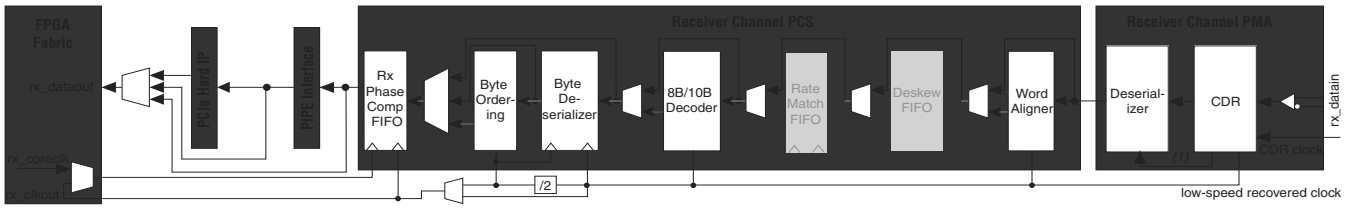



Figure 1–34 shows the datapath clocking in receiver only operation. In this mode, the receiver PCS supports configuration without the rate match FIFO. The CDR unit in the channel recovers the clock from the received serial data and generates the high-speed recovered clock for the deserializer, and low-speed recovered clock for forwarding to the receiver PCS. The low-speed recovered clock feeds to the following blocks in the receiver PCS:

- word aligner
- 8B/10B decoder
- write clock of byte deserializer
- byte ordering
- write clock of RX phase compensation FIFO

When the byte deserializer is enabled, the low-speed recovered clock frequency is halved before feeding into the write clock of the RX phase compensation FIFO. The low-speed recovered clock is available in the FPGA fabric as `rx_clkout` port, which can be used in the FPGA fabric to capture receiver data and status signals.

**Figure 1–34. Receiver Only Datapath Clocking without Rate Match FIFO in Non-Bonded Channel Configuration**




**Note to Figure 1–34:**

(1) High-speed recovered clock.

When the transceiver is configured for transmitter and receiver operation in non-bonded channel configuration, the receiver PCS supports configuration with and without the rate match FIFO. The difference is only at the receiver datapath clocking. The transmitter datapath clocking is identical to transmitter only operation mode as shown in Figure 1–33.

# User Reset and Power-Down Signals

Each transceiver channel in the Cyclone IV GX device has individual reset signals to reset its physical coding sublayer (PCS) and physical medium attachment (PMA). The transceiver block also has a power-down signal that affects the multipurpose phase-locked loops (PLLs), general purpose PLLs, and all the channels in the transceiver block.

☞ All reset and power-down signals are asynchronous.

Table 2–1 lists the reset signals available for each transceiver channel.

**Table 2–1. Transceiver Channel Reset Signals**

| Signal | ALTGX MegaWizard Plug-In Manager Configurations | Description |
|---|---|---|
| tx_digitalreset (1) | ■ Transmitter Only<br>■ Receiver and Transmitter | Provides asynchronous reset to all digital logic in the transmitter PCS, including the XAUI transmit state machine.<br>The minimum pulse width for this signal is two parallel clock cycles. |
| rx_digitalreset (1) | ■ Receiver Only<br>■ Receiver and Transmitter | Resets all digital logic in the receiver PCS, including:<br>■ XAUI receiver state machines<br>■ GIGE receiver state machines<br>■ XAUI channel alignment state machine<br>■ BIST-PRBS verifier<br>■ BIST-incremental verifier<br>The minimum pulse width for this signal is two parallel clock cycles. |
| rx_analogreset | ■ Receiver Only<br>■ Receiver and Transmitter | Resets the receiver CDR present in the receiver channel.<br>The minimum pulse width is two parallel clock cycles. |

**Note to Table 2–1:**

(1) Assert this signal until the clocks coming out of the multipurpose PLL and receiver CDR are stabilized. Stable parallel clocks are essential for proper operation of transmitter and receiver phase-compensation FIFOs in the PCS.

**Table 3–3. Cyclone IV GX Supported Dynamic Reconfiguration Mode (Part 2 of 2)**

| Dynamic Reconfiguration Supported Mode | Operational Mode | | | Quartus II Instances | | | .mif Requirements |
|---|---|---|---|---|---|---|---|
| | Transmitter Only | Receiver Only | Transmitter and Receiver Only | ALTGX | ALTGX_ RECONFIG | ALTPLL_ RECONFIG | |
| Channel Reconfiguration | | | | | | | |
| Channel Interface | ✓ | ✓ | ✓ | ✓ | ✓ | — | ✓ |
| Data Rate Division in Receiver Channel | — | ✓ | ✓ | ✓ | ✓ | — | ✓ |
| PLL Reconfiguration | ✓ | ✓ | ✓ | ✓ | — | ✓ | ✓ |

The following modes are available for dynamically reconfiguring the Cyclone IV transceivers:

- "PMA Controls Reconfiguration Mode" on page 3–13
- "Transceiver Channel Reconfiguration Mode" on page 3–21
  - Channel interface (**.mif** based)
  - Data rate division in receiver channel (**.mif** based)

The following sections describe each of these modes in detail.

The following modes are unsupported for dynamic reconfiguration:

- Dynamically enable/disable PRBS or BIST
- Switch between a receiver-only channel and a transmitter-only channel
- Switch between a ×1 mode to a bonded ×4 mode

## PMA Controls Reconfiguration Mode

You can dynamically reconfigure the following PMA controls for all supported transceiver configurations channels as configured in the ALTGX instances:

- Pre-emphasis settings
- Equalization settings (channel reconfiguration mode does not support equalization settings)
- DC gain settings
- $V_{OD}$ settings

You can use the analog reconfiguration feature to dynamically reconfigure the transceivers channels setting in either the transmitter or the receivers in the PMA blocks. You can update the PMA controls on-the-fly based on the desired input. You can perform both read and write transaction separately for this analog reconfiguration mode.

The following are the channel reconfiguration mode options:

- Channel interface reconfiguration
- Data rate division at receiver channel

## Channel Interface Reconfiguration Mode

Enable this option if the reconfiguration of the transceiver channel involves the following changes:

- The reconfigured channel has a changed FPGA fabric-Transceiver channel interface data width
- The reconfigured channel has changed input control signals and output status signals
- The reconfigured channel has enabled and disabled the static PCS blocks of the transceiver channel

The following are the new input signals available when you enable this option:

- `tx_datainfull`—the width of this input signal depends on the number of channels you set up in the ALTGX MegaWizard Plug-In Manager. It is 22 bits wide per channel. This signal is available only for **Transmitter only** and **Receiver and Transmitter** configurations. This port replaces the existing `tx_datain` port.
- `rx_dataoutfull`—the width of this output signal depends on the number of channels you set up in the ALTGX MegaWizard Plug-In Manager. It is 32 bits wide per channel. This signal is available only for **Receiver only** and **Receiver and Transmitter** configurations. This port replaces the existing `rx_dataout` port.

  The Quartus II software has legality checks for the connectivity of `tx_datainfull` and `rx_dataoutfull` and the various control and status signals you enable in the **Clocking/Interface** screen. For example, the Quartus II software allows you to select and connect the `pipestatus` and `powerdn` signals. It assumes that you are planning to switch to and from PCI Express (PIPE) functional mode.

| Visual Cue | Meaning |
|---|---|
| `Courier type` | Indicates signal, port, register, bit, block, and primitive names. For example, `data1`, `tdi`, and `input`. The suffix `n` denotes an active-low signal. For example, `resetn`.<br>Indicates command line commands and anything that must be typed exactly as it appears. For example, `c:\qdesigns\tutorial\chiptrip.gdf`.<br>Also indicates sections of an actual file, such as a Report File, references to parts of files (for example, the AHDL keyword `SUBDESIGN`), and logic function names (for example, `TRI`). |
| ↵ | An angled arrow instructs you to press the Enter key. |
| 1., 2., 3., and<br>a., b., c., and so on | Numbered steps indicate a list of items when the sequence of the items is important, such as the steps listed in a procedure. |
| ■ ■ ■ | Bullets indicate a list of items when the sequence of the items is not important. |
|  | The hand points to information that requires special attention. |
|  | The question mark directs you to a software help system with related information. |
|  | The feet direct you to another document or website with related information. |
|  | The multimedia icon directs you to a related multimedia presentation. |
|  | A caution calls attention to a condition or possible situation that can damage or destroy the product or your work. |
|  | A warning calls attention to a condition or possible situation that can cause you injury. |
| | The envelope links to the Email Subscription Management Center page of the Altera website, where you can sign up to receive update notifications for Altera documents. |

The OCT resistance may vary with the variation of temperature and voltage after calibration at device power-up. Use Table 1–10 and Equation 1–1 to determine the final OCT resistance considering the variations after calibration at device power-up. Table 1–10 lists the change percentage of the OCT resistance with voltage and temperature.

**Table 1–10. OCT Variation After Calibration at Device Power-Up for Cyclone IV Devices** *(1)*

| Nominal Voltage | dR/dT (%/°C) | dR/dV (%/mV) |
|---|---|---|
| 3.0 | 0.262 | –0.026 |
| 2.5 | 0.234 | –0.039 |
| 1.8 | 0.219 | –0.086 |
| 1.5 | 0.199 | –0.136 |
| 1.2 | 0.161 | –0.288 |

**Note to Table 1–10:**

(1) This specification is not applicable to EP4CGX15, EP4CGX22, and EP4CGX30 devices.

**Equation 1–1. Final OCT Resistance** *(1), (2), (3), (4), (5), (6)*

$$\Delta R_V = (V_2 - V_1) \times 1000 \times dR/dV \quad (7)$$

$$\Delta R_T = (T_2 - T_1) \times dR/dT \quad (8)$$

$$\text{For } \Delta R_x < 0;\ MF_x = 1/ (|\Delta R_x|/100 + 1) \quad (9)$$

$$\text{For } \Delta R_x > 0;\ MF_x = \Delta R_x/100 + 1 \quad (10)$$

$$MF = MF_V \times MF_T \quad (11)$$

$$R_{final} = R_{initial} \times MF \quad (12)$$

**Notes to Equation 1–1:**

(1) $T_2$ is the final temperature.
(2) $T_1$ is the initial temperature.
(3) MF is multiplication factor.
(4) $R_{final}$ is final resistance.
(5) $R_{initial}$ is initial resistance.
(6) Subscript $_x$ refers to both $_V$ and $_T$.
(7) $\Delta R_V$ is a variation of resistance with voltage.
(8) $\Delta R_T$ is a variation of resistance with temperature.
(9) dR/dT is the change percentage of resistance with temperature after calibration at device power-up.
(10) dR/dV is the change percentage of resistance with voltage after calibration at device power-up.
(11) $V_2$ is final voltage.
(12) $V_1$ is the initial voltage.