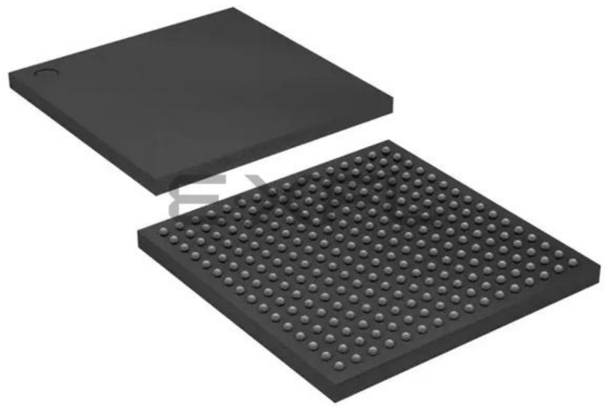
Intel - EP4CE22F17C8 Datasheet



Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

Applications of Embedded - FPGAs

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

Details	
Product Status	Active
Number of LABs/CLBs	1395
Number of Logic Elements/Cells	22320
Total RAM Bits	608256
Number of I/O	153
Number of Gates	-
Voltage - Supply	1.15V ~ 1.25V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	256-LBGA
Supplier Device Package	256-FBGA (17x17)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce22f17c8

© 2016 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at www.altera.com/common/legal.html. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered

Read or Write Clock Mode

Cyclone IV devices M9K memory blocks can implement read or write clock mode for FIFO and simple dual-port memories. In this mode, a write clock controls the data inputs, write address, and `wren` registers. Similarly, a read clock controls the data outputs, read address, and `rden` registers. M9K memory blocks support independent clock enables for both the read and write clocks.

When using read or write mode, if you perform a simultaneous read or write to the same address location, the output read data is unknown. If you require the output data to be a known value, use either single-clock mode, input clock mode, or output clock mode and choose the appropriate read-during-write behavior in the MegaWizard Plug-In Manager.

Single-Clock Mode

Cyclone IV devices M9K memory blocks can implement single-clock mode for FIFO, ROM, true dual-port, simple dual-port, and single-port memories. In this mode, you can control all registers of the M9K memory block with a single clock together with clock enable.

Design Considerations

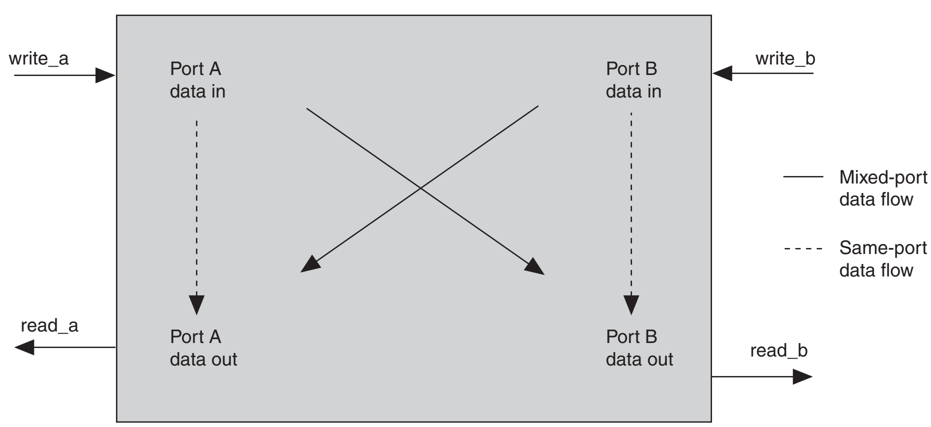
This section describes designing with M9K memory blocks.

Read-During-Write Operations

"Same-Port Read-During-Write Mode" on page 3–16 and "Mixed-Port Read-During-Write Mode" on page 3–16 describe the functionality of the various RAM configurations when reading from an address during a write operation at that same address.

There are two read-during-write data flows: same-port and mixed-port. Figure 3–13 shows the difference between these flows.

Figure 3–13. Cyclone IV Devices Read-During-Write Data Flow



Power-Up Conditions and Memory Initialization

The M9K memory block outputs of Cyclone IV devices power up to zero (cleared) regardless of whether the output registers are used or bypassed. All M9K memory blocks support initialization using a **.mif**. You can create **.mif**s in the Quartus II software and specify their use using the RAM MegaWizard Plug-In Manager when instantiating memory in your design. Even if memory is pre-initialized (for example, using a **.mif**), it still powers up with its outputs cleared. Only the subsequent read after power up outputs the pre-initialized values.

For more information about **.mif**s, refer to the *RAM Megafunction User Guide* and the *Quartus II Handbook*.

Power Management

The M9K memory block clock enables of Cyclone IV devices allow you to control clocking of each M9K memory block to reduce AC power consumption. Use the `rden` signal to ensure that read operations only occur when necessary. If your design does not require read-during-write, reduce power consumption by deasserting the `rden` signal during write operations or any period when there are no memory operations. The Quartus II software automatically powers down any unused M9K memory blocks to save static power.

Document Revision History

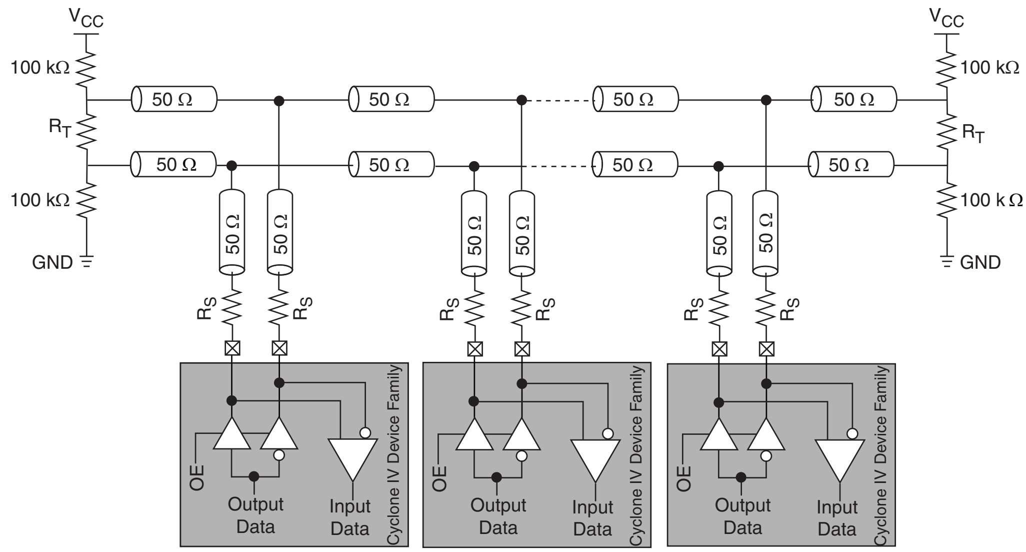
Table 3–6 shows the revision history for this chapter.

Table 3–6. Document Revision History

Date	Version	Changes
November 2011	1.1	Updated the "Byte Enable Support" section.
November 2009	1.0	Initial release.

Figure 6–14 shows a typical BLVDS topology with multiple transmitter and receiver pairs.

Figure 6–14. BLVDS Topology with Cyclone IV Devices Transmitters and Receivers



The BLVDS I/O standard is supported on the top, bottom, and right I/O banks of Cyclone IV devices. The BLVDS transmitter uses two single-ended output buffers with the second output buffer programmed as inverted, while the BLVDS receiver uses a true LVDS input buffer. The transmitter and receiver share the same pins. An output-enabled (`OE`) signal is required to tristate the output buffers when the LVDS input buffer receives a signal.

For more information, refer to the *Cyclone IV Device Datasheet* chapter.

Designing with BLVDS

The BLVDS bidirectional communication requires termination at both ends of the bus in BLVDS. The termination resistor (R_T) must match the bus differential impedance, which in turn depends on the loading on the bus. Increasing the load decreases the bus differential impedance. With termination at both ends of the bus, termination is not required between the two signals at the input buffer. A single series resistor (R_S) is required at the output buffer to match the output buffer impedance to the transmission line impedance. However, this series resistor affects the voltage swing at the input buffer. The maximum data rate achievable depends on many factors.

Altera recommends that you perform simulation using the IBIS model while considering factors such as bus loading, termination values, and output and input buffer location on the bus to ensure that the required performance is achieved.

For more information about BLVDS interface support in Altera devices, refer to *AN 522: Implementing Bus LVDS Interface in Supported Altera Device Families*.

RSDS, Mini-LVDS, and PPDS I/O Standard Support in Cyclone IV Devices

The RSDS, mini-LVDS, and PPDS I/O standards are used in chip-to-chip applications between the timing controller and the column drivers on the display panels such as LCD monitor panels and LCD televisions. Cyclone IV devices meet the National Semiconductor Corporation RSDS Interface Specification, Texas Instruments mini-LVDS Interface Specification, and National Semiconductor Corporation PPDS Interface Specification to support RSDS, mini-LVDS and PPDS output standards, respectively.

For Cyclone IV devices RSDS, mini-LVDS, and PPDS output electrical specifications, refer to the *Cyclone IV Device Datasheet* chapter.

For more information about the RSDS I/O standard, refer to the RSDS specification from the National Semiconductor website (www.national.com).

Designing with RSDS, Mini-LVDS, and PPDS

Cyclone IV I/O banks support RSDS, mini-LVDS, and PPDS output standards. The right I/O banks support true RSDS, mini-LVDS, and PPDS transmitters. On the top and bottom I/O banks, RSDS, mini-LVDS, and PPDS transmitters are supported using two single-ended output buffers with external resistors. The two single-ended output buffers are programmed to have opposite polarity.

Figure 6–15 shows an RSDS, mini-LVDS, or PPDS interface with a true output buffer.

Figure 6–15. Cyclone IV Devices RSDS, Mini-LVDS, or PPDS Interface with True Output Buffer on the Right I/O Banks

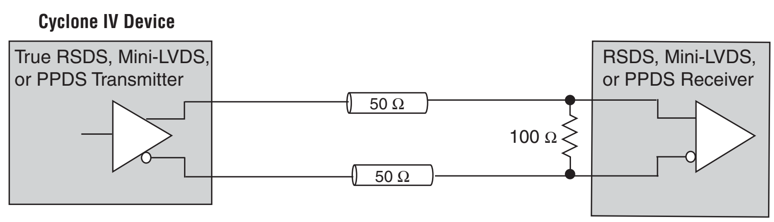
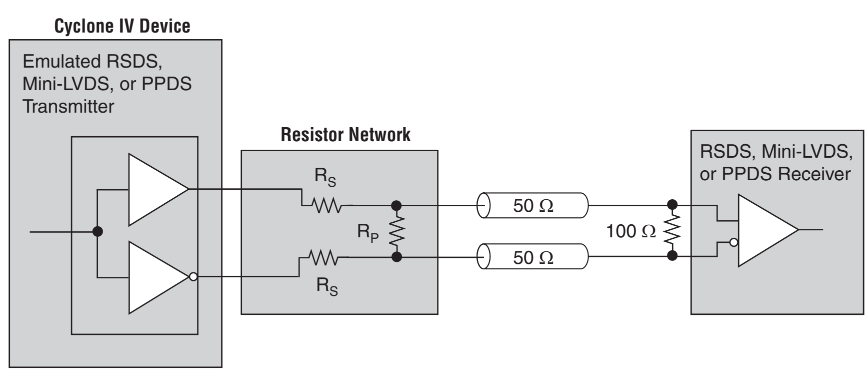


Figure 6–16 shows an RSDS, mini-LVDS, or PPDS interface with two single-ended output buffers and external resistors.

Figure 6–16. RSDS, Mini-LVDS, or PPDS Interface with External Resistor Network on the Top and Bottom I/O Banks *(1)*

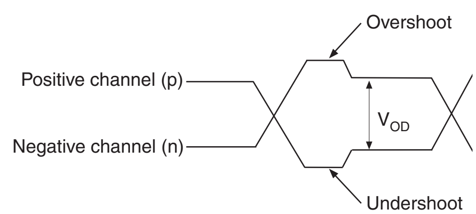


before the next edge; this may lead to pattern-dependent jitter. With pre-emphasis, the output current is momentarily boosted during switching to increase the output slew rate. The overshoot produced by this extra switching current is different from the overshoot caused by signal reflection. This overshoot 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: $\mathrm{RSKM} = \frac{(\mathrm{TUI} - \mathrm{SW} - \mathrm{TCCS})}{2}$

Table 8–2. Uncompressed Raw Binary File (.rbf) Sizes for Cyclone IV Devices (Part 2 of 2)

Device		Data Size (bits)
Cyclone IV GX	EP4CGX15	3,805,568
	EP4CGX22	7,600,040
	EP4CGX30	7,600,040
		22,010,888 *(1)*
	EP4CGX50	22,010,888
	EP4CGX75	22,010,888
	EP4CGX110	39,425,016
	EP4CGX150	39,425,016

Note to Table 8–2:

(1) Only for the F484 package.

Use the data in Table 8–2 to estimate the file size before design compilation. Different configuration file formats, such as Hexadecimal (**.hex**) or Tabular Text File (**.ttf**) formats, have different file sizes. However, for any specific version of the Quartus II software, any design targeted for the same device has the same uncompressed configuration file size. If you use compression, the file size varies after each compilation, because the compression ratio depends on the design.

For more information about setting device configuration options or creating configuration files, refer to the *Software Settings* section in volume 2 of the *Configuration Handbook*.

Configuration and JTAG Pin I/O Requirements

Cyclone IV devices are manufactured using the TSMC 60-nm low-k dielectric process. Although Cyclone IV devices use TSMC 2.5-V transistor technology in the I/O buffers, the devices are compatible and able to interface with 2.5, 3.0, and 3.3-V configuration voltage standards by following specific requirements.

All I/O inputs must maintain a maximum AC voltage of 4.1 V. When using a serial configuration device in an AS configuration scheme, you must connect a 25-Ω series resistor for the `DATA[0]` pin. When cascading the Cyclone IV device family in a multi-device configuration for AS, AP, FPP, and PS configuration schemes, you must connect the repeater buffers between the master and slave devices for the `DATA` and `DCLK` pins. When using the JTAG configuration scheme in a multi-device configuration, connect 25-Ω resistors on both ends of the `TDO-TDI` path if the `TDO` output driver is a non-Cyclone IV device.

The output resistance of the repeater buffers and the `TDO` path for all cases must fit the maximum overshoot equation shown in Equation 8–1.

Equation 8–1. *(1)*

$$0.8Z_O \le R_E \le 1.8Z_O$$

Note to Equation 8–1:

(1) Z_O is the transmission line impedance and R_E is the equivalent resistance of the output buffer.

Table 8–4. Configuration Schemes for Cyclone IV GX Devices (EP4CGX30 [only for F484 package], EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150) (Part 2 of 2)

Configuration Scheme	MSEL3	MSEL2	MSEL1	MSEL0	POR Delay	Configuration Voltage Standard (V) *(1)*
JTAG-based configuration *(2)*	*(3)*	*(3)*	*(3)*	*(3)*	—	—

Notes to Table 8–4:

(1) Configuration voltage standard applied to the V_{CCIO} supply of the bank in which the configuration pins reside.

(2) JTAG-based configuration takes precedence over other configuration schemes, which means the `MSEL` pin settings are ignored.

(3) Do not leave the `MSEL` pins floating. Connect them to V_{CCA} or GND. These pins support the non-JTAG configuration scheme used in production. Altera recommends connecting the `MSEL` pins to GND if your device is only using JTAG configuration.

Smaller Cyclone IV E devices or package options (E144 and F256 packages) do not have the `MSEL[3]` pin. The AS Fast POR configuration scheme at 3.0- or 2.5-V configuration voltage standard and the AP configuration scheme are not supported in Cyclone IV E devices without the `MSEL[3]` pin. To configure these devices with other supported configuration schemes, select `MSEL[2..0]` pins according to the `MSEL` settings in Table 8–5.

Table 8–5. Configuration Schemes for Cyclone IV E Devices

Configuration Scheme	MSEL3	MSEL2	MSEL1	MSEL0	POR Delay	Configuration Voltage Standard (V) *(1)*
AS	1	1	0	1	Fast	3.3
	0	1	0	0	Fast	3.0, 2.5
	0	0	1	0	Standard	3.3
	0	0	1	1	Standard	3.0, 2.5
AP	0	1	0	1	Fast	3.3
	0	1	1	0	Fast	1.8
	0	1	1	1	Standard	3.3
	1	0	1	1	Standard	3.0, 2.5
	1	0	0	0	Standard	1.8
PS	1	1	0	0	Fast	3.3, 3.0, 2.5
	0	0	0	0	Standard	3.3, 3.0, 2.5
FPP	1	1	1	0	Fast	3.3, 3.0, 2.5
	1	1	1	1	Fast	1.8, 1.5
JTAG-based configuration *(2)*	*(3)*	*(3)*	*(3)*	*(3)*	—	—

Notes to Table 8–5:

(1) Configuration voltage standard applied to the V_{CCIO} supply of the bank in which the configuration pins reside.

(2) JTAG-based configuration takes precedence over other configuration schemes, which means the `MSEL` pin settings are ignored.

(3) Do not leave the `MSEL` pins floating. Connect them to V_{CCA} or GND. These pins support the non-JTAG configuration scheme used in production. Altera recommends connecting the `MSEL` pins to GND if your device is only using JTAG configuration.

The first Cyclone IV device in the chain is the configuration master and it controls the configuration of the entire chain. Other Altera devices that support PS configuration can also be part of the chain as configuration slaves.

☞ In the multi-device AS configuration, the board trace length between the serial configuration device and the master device of the Cyclone IV device must follow the recommendations in Table 8–7 on page 8–18.

The nSTATUS and CONF_DONE pins on all target devices are connected together with external pull-up resistors, as shown in Figure 8–3 on page 8–13. These pins are open-drain bidirectional pins on the devices. When the first device asserts nCEO (after receiving all its configuration data), it releases its CONF_DONE pin. However, the subsequent devices in the chain keep this shared CONF_DONE line low until they receive their configuration data. When all target devices in the chain receive their configuration data and release CONF_DONE, the pull-up resistor drives a high level on CONF_DONE line and all devices simultaneously enter initialization mode.

☞ Although you can cascade Cyclone IV devices, serial configuration devices cannot be cascaded or chained together.

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 device's configuration bitstream.

Configuring Multiple Cyclone IV Devices with the Same Design

Certain designs require that you configure multiple Cyclone IV devices with the same design through a configuration bitstream, or a **.sof**. You can do this through the following methods:

- Multiple **.sof**
- Single **.sof**

☞ For both methods, the serial configuration devices cannot be cascaded or chained together.

Multiple SRAM Object Files

Two copies of the **.sof** are stored in the serial configuration device. Use the first copy to configure the master device of the Cyclone IV device and the second copy to configure all remaining slave devices concurrently. All slave devices must have the same density and package. The setup is similar to Figure 8–3 on page 8–13.

To configure four identical Cyclone IV devices with the same **.sof**, you must set up the chain similar to the example shown in Figure 8–4. The first device is the master device and its MSEL pins must be set to select AS configuration. The other three slave devices are set up for concurrent configuration and their MSEL pins must be set to select PS configuration. The nCEO pin from the master device drives the nCE input pins on all three slave devices, as well as the DATA and DCLK pins that connect in parallel to all

EN_ACTIVE_CLK

The `EN_ACTIVE_CLK` instruction causes the `CLKUSR` pin signal to replace the internal oscillator as the clock source. When using the `EN_ACTIVE_CLK` instruction, you must enable the internal oscillator for the clock change to occur. After this instruction is issued, other JTAG instructions can be issued while the `CLKUSR` pin signal remains as the clock source. The clock source is only reverted back to the internal oscillator by issuing the `DIS_ACTIVE_CLK` instruction or a POR.

DIS_ACTIVE_CLK

The `DIS_ACTIVE_CLK` instruction breaks the `CLKUSR` enable latch set by the `EN_ACTIVE_CLK` instruction and causes the clock source to revert back to the internal oscillator. After the `DIS_ACTIVE_CLK` instruction is issued, you must continue to clock the `CLKUSR` pin for 10 clock cycles.

Changing the Start Boot Address of the AP Flash

In the AP configuration scheme (for Cyclone IV E devices only), you can change the default configuration boot address of the parallel flash memory to any desired address using the `APFC_BOOT_ADDR` JTAG instruction.

APFC_BOOT_ADDR

The `APFC_BOOT_ADDR` instruction is for Cyclone IV E devices only and allows you to define a start boot address for the parallel flash memory in the AP configuration scheme.

This instruction shifts in a start boot address for the AP flash. When this instruction becomes the active instruction, the `TDI` and `TDO` pins are connected through a 22-bit active boot address shift register. The shifted-in boot address bits get loaded into the 22-bit AP boot address update register, which feeds into the AP controller. The content of the AP boot address update register can be captured and shifted-out of the active boot address shift register from `TDO`.

The boot address in the boot address shift register and update register are shifted to the right (in the LSB direction) by two bits versus the intended boot address. The reason for this is that the two LSB of the address are not accessible. When this boot address is fed into the AP controller, two 0s are attached in the end as LSB, thereby pushing the shifted-in boot address to the left by two bits, which become the actual AP boot address the AP controller gets.

If you have enabled the remote update feature, the `APFC_BOOT_ADDR` instruction sets the boot address for the factory configuration only.

☞ The `APFC_BOOT_ADDR` instruction is retained after reconfiguration while the system board is still powered on. However, you must reprogram the instruction whenever you restart the system board.

Document Revision History

Table 10–3 lists the revision history for this chapter.

Table 10–3. Document Revision History

Date	Version	Changes
December 2013	1.3	■ Updated the "EXTEST_PULSE" section.
November 2011	1.2	■ Updated the "BST Operation Control" section. ■ Updated Table 10–2.
February 2010	1.1	■ Added Cyclone IV E devices in Table 10–1 and Table 10–2 for the Quartus II software version 9.1 SP1 release. ■ Updated Figure 10–1 and Figure 10–2. ■ Minor text edits.
November 2009	1.0	Initial release.

Contents

Section I. Transceivers

Chapter 1. Cyclone IV Transceivers Architecture

Table 1–4 lists the synchronization state machine parameters for the word aligner in this mode.

Table 1–4. Synchronization State Machine Parameters

Parameter	Allowed Values
Number of erroneous code groups received to lose synchronization	1–64
Number of continuous good code groups received to reduce the error count by one	1–256

After deassertion of the `rx_digitalreset` signal in automatic synchronization state machine mode, the word aligner starts looking for the synchronization code groups, word alignment pattern or its complement in the received data stream. When the programmed number of valid synchronization code groups or ordered sets are received, the `rx_syncstatus` signal is driven high to indicate that synchronization is acquired. The `rx_syncstatus` signal is constantly driven high until the programmed number of erroneous code groups are received without receiving intermediate good groups; after which the `rx_syncstatus` signal is driven low. The word aligner indicates loss of synchronization (`rx_syncstatus` signal remains low) until the programmed number of valid synchronization code groups are received again.

In addition to restoring word boundaries, the word aligner supports the following features:

- Programmable run length violation detection—detects consecutive 1s or 0s in the data stream, and asserts run length violation signal (`rx_rlv`) when a preset run length threshold (maximum number of consecutive 1s or 0s) is detected. The `rx_rlv` signal in each channel is clocked by its parallel recovered clock and is asserted for a minimum of two recovered clock cycles to ensure that the FPGA fabric clock can latch the `rx_rlv` signal reliably because the FPGA fabric clock might have phase differences, ppm differences (in asynchronous systems), or both, with the recovered clock. Table 1–5 lists the run length violation circuit detection capabilities.

Table 1–5. Run Length Violation Circuit Detection Capabilities

Supported Data Width	Detector Range		Increment Step Settings
	Minimum	Maximum	
8-bit	4	128	4
10-bit	5	160	5

Figure 1–35 shows the datapath clocking in the transmitter and receiver operation mode with the rate match FIFO. The receiver datapath clocking in configuration without the rate match FIFO is identical to Figure 1–34.

In configuration with the rate match FIFO, the CDR unit in the receiver channel recovers the clock from 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:

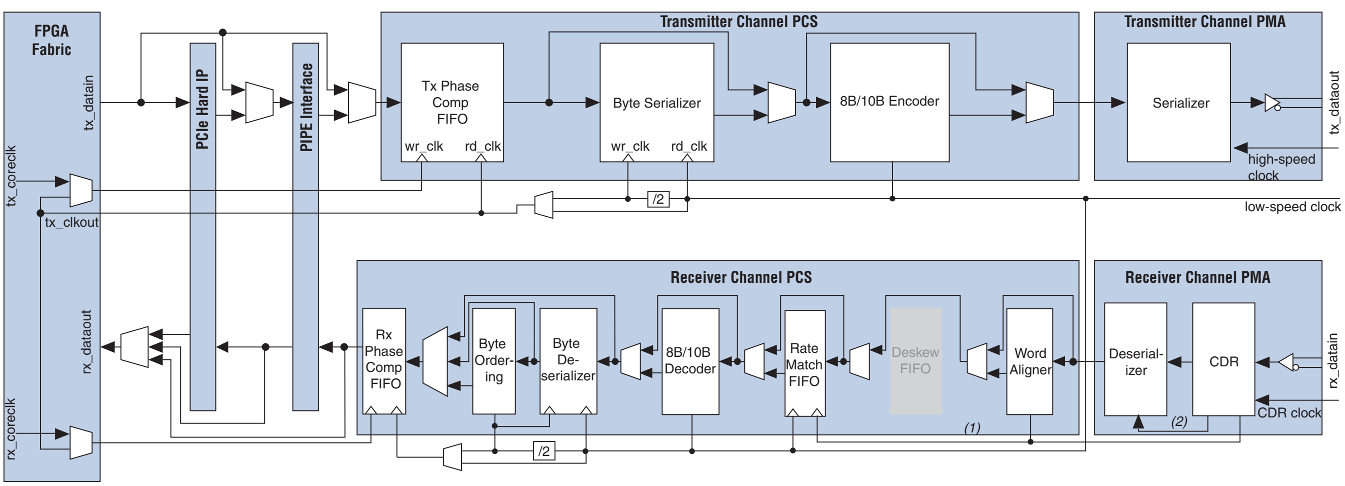
- word aligner
- write clock of rate match FIFO

The low-speed clock that is used in the transmitter PCS datapath feeds the following blocks in the receiver PCS:

- read clock of rate match FIFO
- 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 clock frequency is halved before feeding into the write clock of RX 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, and capture receiver data and status signals.

Figure 1–35. Transmitter and Receiver Datapath Clocking with Rate Match FIFO in Non-Bonded Channel Configuration



Notes to Figure 1–35:

(1) Low-speed recovered clock.

(2) High-speed recovered clock.

Table 1–11. FPGA Fabric-Transceiver Interface Clocks (Part 2 of 2)

Clock Name	Clock Description	Interface Direction
cal_blk_clk (2)	Transceiver calibration block clock	FPGA fabric to transceiver

Notes to Table 1–11:

(1) Offset cancellation process that is executed after power cycle requires `reconfig_clk` clock. The `reconfig_clk` must be driven with a free-running clock and not derived from the transceiver blocks.

(2) For the supported clock frequency range, refer to the *Cyclone IV Device Data Sheet*.

In the transmitter datapath, TX phase compensation FIFO forms the FPGA fabric-transmitter interface. Data and control signals for the transmitter are clocked with the FIFO write clock. The FIFO write clock supports automatic clock selection by the Quartus II software (depending on channel configuration), or user-specified clock from `tx_coreclk` port. Table 1–12 details the automatic TX phase compensation FIFO write clock selection by the Quartus II software.

The Quartus II software assumes automatic clock selection for TX phase compensation FIFO write clock if you do not enable the `tx_coreclk` port.

Table 1–12. Automatic TX Phase Compensation FIFO Write Clock Selection

Channel Configuration	Quartus II Selection
Non-bonded	`tx_clkout` clock feeds the FIFO write clock. `tx_clkout` is forwarded through the transmitter channel from low-speed clock, which also feeds the FIFO read clock.
Bonded	`coreclkout` clock feeds the FIFO write clock for the bonded channels. `coreclkout` clock is the common bonded low-speed clock, which also feeds the FIFO read clock in the bonded channels.

When using user-specified clock option, ensure that the clock feeding `tx_coreclk` port has 0 ppm difference with the TX phase compensation FIFO read clock.

In the receiver datapath, RX phase compensation FIFO forms the receiver-FPGA fabric interface. Data and status signals from the receiver are clocked with the FIFO read clock. The FIFO read clock supports automatic clock selection by the Quartus II software (depending on channel configuration), or user-specified clock from `rx_coreclk` port. Table 1–13 details the automatic RX phase compensation FIFO read clock selection by the Quartus II software.

The Quartus II software assumes automatic clock selection for RX phase compensation FIFO read clock if you do not enable the `rx_coreclk` port.

Table 1–13. Automatic RX Phase Compensation FIFO Read Clock Selection (Part 1 of 2)

Channel Configuration		Quartus II Selection
Non-bonded	With rate match FIFO (1)	`tx_clkout` clock feeds the FIFO read clock. `tx_clkout` is forwarded through the receiver channel from low-speed clock, which also feeds the FIFO write clock and transmitter PCS.
	Without rate match FIFO	`rx_clkout` clock feeds the FIFO read clock. `rx_clkout` is forwarded through the receiver channel from low-speed recovered clock, which also feeds the FIFO write clock.

Table 1–14. Transceiver Functional Modes for Protocol Implementation (Part 2 of 2)

Functional Mode	Protocol	Key Feature	Reference
Deterministic Latency	Proprietary, CPRI, OBSAI	TX PLL phase frequency detector (PFD) feedback, registered mode FIFO, TX bit-slip control	"Deterministic Latency Mode" on page 1–73
SDI	SDI	High-speed SERDES, CDR	"SDI Mode" on page 1–76

Basic Mode

The Cyclone IV GX transceiver channel datapath is highly flexible in Basic mode to implement proprietary protocols. SATA, V-by-One, and Display Port protocol implementations in Cyclone IV GX transceiver are supported with Basic mode. Figure 1–44 shows the transceiver channel datapath supported in Basic mode.

Figure 1–44. Transceiver Channel Datapath in Basic Mode

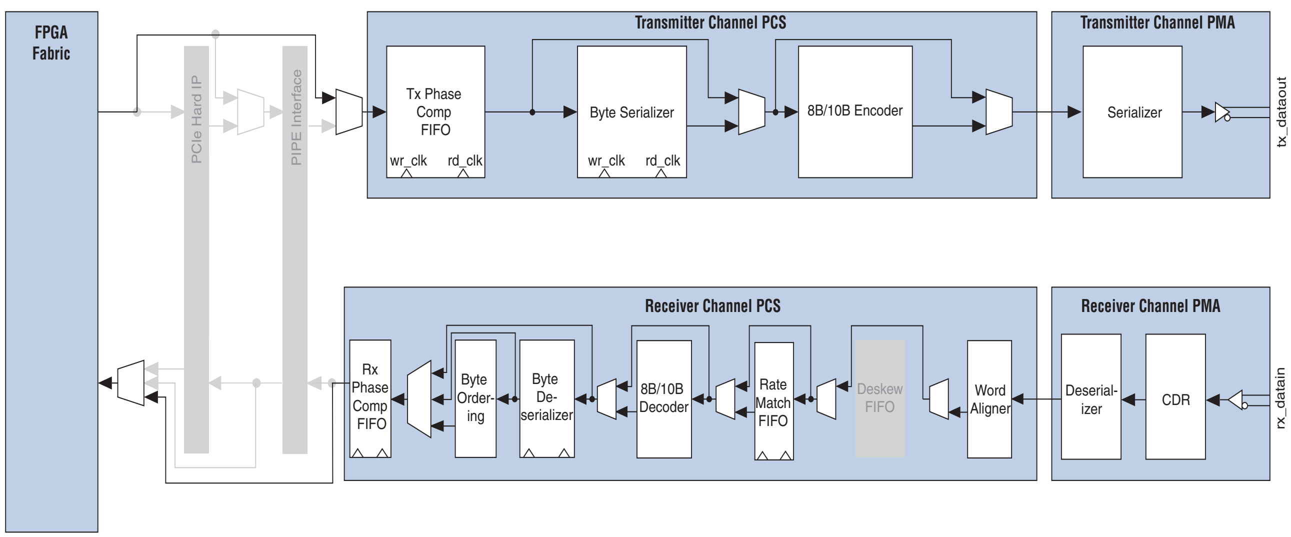
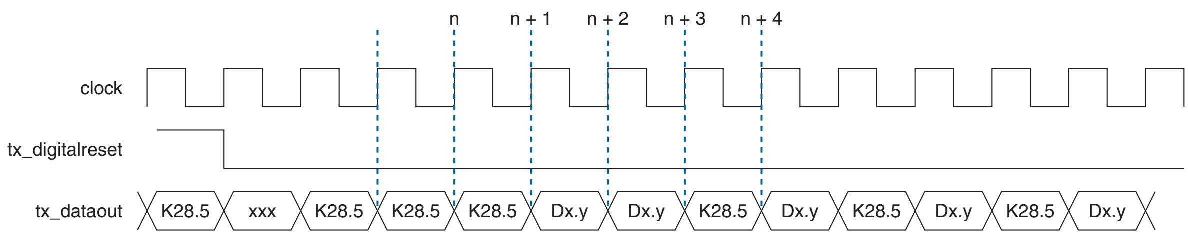


Figure 1–57 shows an example of even numbers of /Dx.y/ between the last automatically sent /K28.5/ and the first user-sent /K28.5/. The first user-sent /K28.5/ code group received at an odd code group boundary in cycle $n + 3$ takes the receiver synchronization state machine in Loss-of-Sync state. The first synchronization ordered-set /K28.5/Dx.y/ in cycles $n + 3$ and $n + 4$ is discounted and three additional ordered sets are required for successful synchronization.

Figure 1–57. Example of Reset Condition in GIGE Mode



Running Disparity Preservation with Idle Ordered Set

During idle ordered sets transmission in GIGE mode, the transmitter ensures a negative running disparity at the end of an idle ordered set. Any /Dx.y/, except for /D21.5/ (part of /C1/ ordered set) or /D2.2/ (part of /C2/ ordered set) following a /K28.5/ is automatically replaced with either of the following:

- A /D5.6/ (/I1/ ordered set) if the running disparity before /K28.5/ is positive
- A /D16.2/ (/I2/ ordered set) if the running disparity before /K28.5/ is negative

Lane Synchronization

In GIGE mode, the word aligner is configured in automatic synchronization state machine mode that complies with the IEEE P802.3ae standard. A synchronization ordered set is a /K28.5/ code group followed by an odd number of valid /Dx.y/ code groups. Table 1–19 lists the synchronization state machine parameters that implements the GbE-compliant synchronization.

Table 1–19. Synchronization State Machine Parameters (1)

Parameter	Value
Number of valid synchronization ordered sets received to achieve synchronization	3
Number of erroneous code groups received to lose synchronization	4
Number of continuous good code groups received to reduce the error count by one	4

Note to Table 1–19:

(1) The word aligner supports 7-bit and 10-bit pattern lengths in GIGE mode.

Functional Simulation of the Dynamic Reconfiguration Process

This section describes the points to be considered during functional simulation of the dynamic reconfiguration process.

- You must connect the ALTGX_RECONFIG instance to the ALTGX_instance/ALTGX instances in your design for functional simulation.
- The functional simulation uses a reduced timing model of the dynamic reconfiguration controller. The duration of the offset cancellation process is 16 `reconfig_clk` clock cycles for functional simulation only.
- The `gxb_powerdown` signal must not be asserted during the offset cancellation sequence (for functional simulation and silicon).

Document Revision History

Table 3–8 lists the revision history for this chapter.

Table 3–8. Document Revision History

Date	Version	Changes
November 2011	2.1	■ Updated "Dynamic Reconfiguration Controller Architecture", "PMA Controls Reconfiguration Mode", "PLL Reconfiguration Mode", and "Error Indication During Dynamic Reconfiguration" sections. ■ Updated Table 3–2 and Table 3–4.
December 2010	2.0	■ Updated for the Quartus II software version 10.1 release. ■ Updated Table 3–1, Table 3–2, Table 3–3, Table 3–4, Table 3–5, and Table 3–6. ■ Added Table 3–7. ■ Updated Figure 3–1, Figure 3–11, Figure 3–13, and Figure 3–14. ■ Updated "Offset Cancellation Feature", "Error Indication During Dynamic Reconfiguration", "Data Rate Reconfiguration Mode Using RX Local Divider", "PMA Controls Reconfiguration Mode", and "Control and Status Signals for Channel Reconfiguration" sections.
July 2010	1.0	Initial release.

Power Consumption

Use the following methods to estimate power for a design:

- the Excel-based EPE
- the Quartus® II PowerPlay power analyzer feature

The interactive Excel-based EPE is used prior to designing the device to get a magnitude estimate of the device power. The Quartus II PowerPlay power analyzer provides better quality estimates based on the specifics of the design after place-and-route is complete. The PowerPlay power analyzer can apply a combination of user-entered, simulation-derived, and estimated signal activities that, combined with detailed circuit models, can yield very accurate power estimates.

For more information about power estimation tools, refer to the *Early Power Estimator User Guide* and the *PowerPlay Power Analysis* chapter in volume 3 of the *Quartus II Handbook*.

Switching Characteristics

This section provides performance characteristics of Cyclone IV core and periphery blocks for commercial grade devices.

These characteristics can be designated as Preliminary or Final.

- Preliminary characteristics are created using simulation results, process data, and other known parameters. The upper-right hand corner of these tables show the designation as “Preliminary”.
- Final numbers are based on actual silicon characterization and testing. The numbers reflect the actual performance of the device under worst-case silicon process, voltage, and junction temperature conditions. There are no designations on finalized tables.