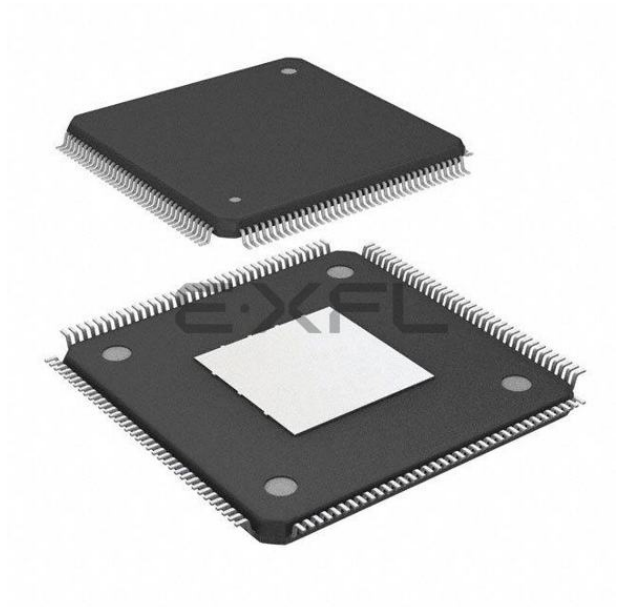


Welcome to E-XFL.COM

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.

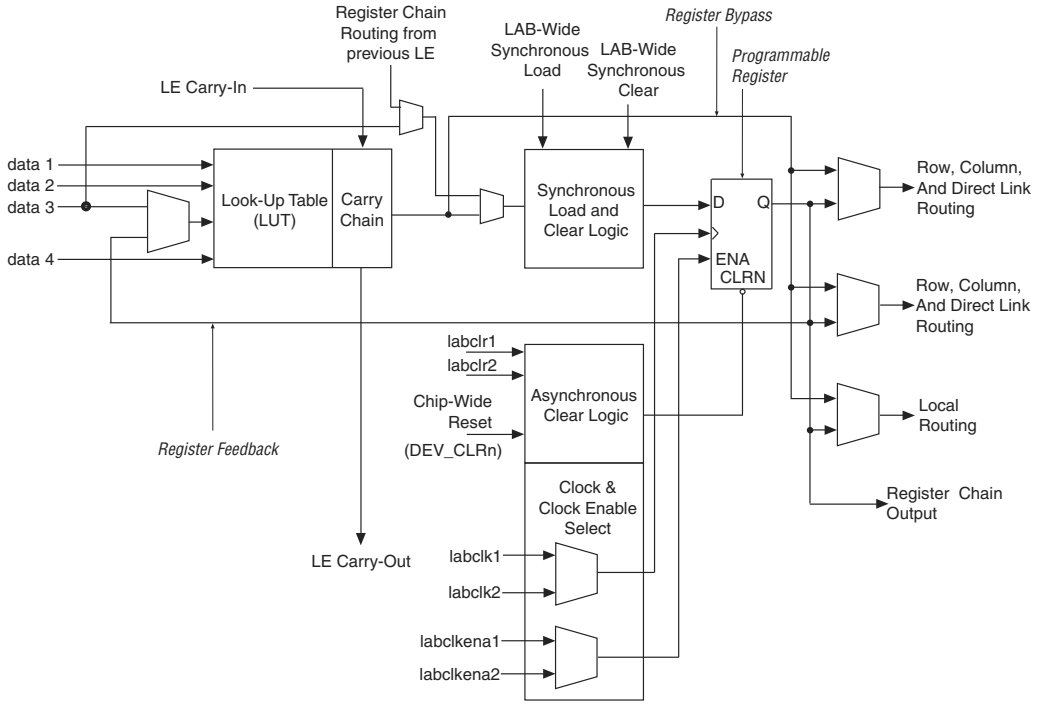
Applications of Embedded - FPGAs

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	-40°C ~ 100°C (TJ)
Package / Case	144-LQFP Exposed Pad
Supplier Device Package	144-EQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce22e22i8l

Figure 2–1 shows the LEs for Cyclone IV devices.

Figure 2–1. Cyclone IV Device LEs



LE Features

You can configure the programmable register of each LE for D, T, JK, or SR flipflop operation. Each register has data, clock, clock enable, and clear inputs. Signals that use the global clock network, general-purpose I/O pins, or any internal logic can drive the clock and clear control signals of the register. Either general-purpose I/O pins or the internal logic can drive the clock enable. For combinational functions, the LUT output bypasses the register and drives directly to the LE outputs.

Each LE has three outputs that drive the local, row, and column routing resources. The LUT or register output independently drives these three outputs. Two LE outputs drive the column or row and direct link routing connections, while one LE drives the local interconnect resources. This allows the LUT to drive one output while the register drives another output. This feature, called register packing, improves device utilization because the device can use the register and the LUT for unrelated functions. The LAB-wide synchronous load control signal is not available when using register packing. For more information about the synchronous load control signal, refer to "LAB Control Signals" on page 2–6.

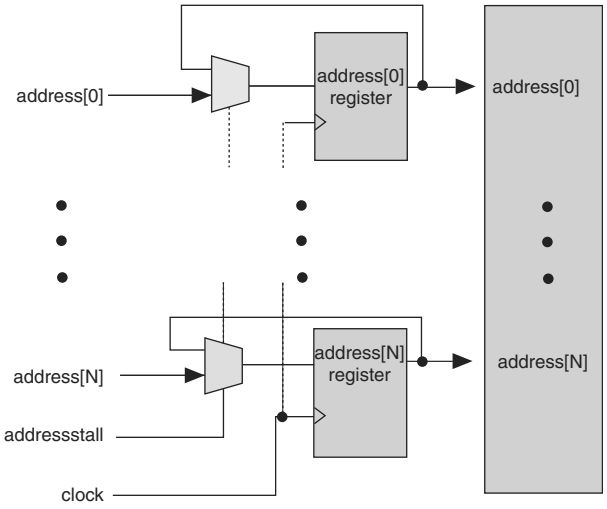
The register feedback mode allows the register output to feed back into the LUT of the same LE to ensure that the register is packed with its own fan-out LUT, providing another mechanism for improved fitting. The LE can also drive out registered and unregistered versions of the LUT output.

Address Clock Enable Support

Cyclone IV devices M9K memory blocks support an active-low address clock enable, which holds the previous address value for as long as the `addressstall` signal is high (`addressstall = '1'`). When you configure M9K memory blocks in dual-port mode, each port has its own independent address clock enable.

Figure 3–2 shows an address clock enable block diagram. The address register output feeds back to its input using a multiplexer. The multiplexer output is selected by the address clock enable (`addressstall`) signal.

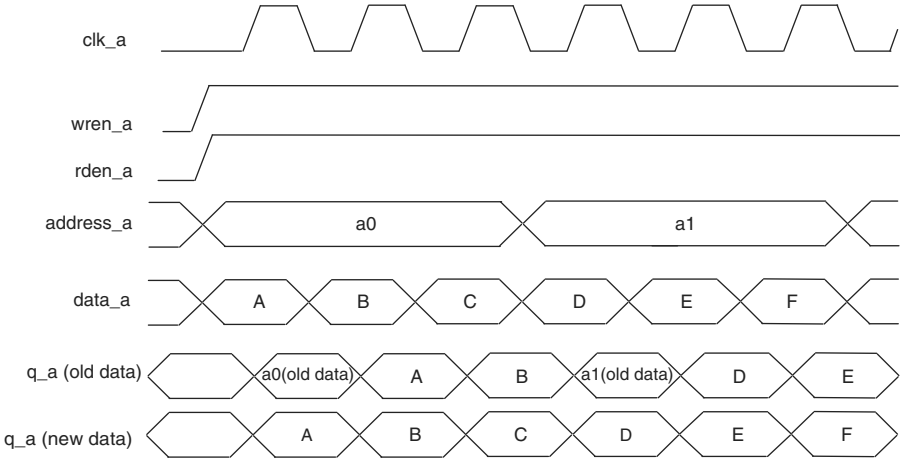
Figure 3–2. Cyclone IV Devices Address Clock Enable Block Diagram



The address clock enable is typically used to improve the effectiveness of cache memory applications during a cache-miss. The default value for the address clock enable signals is low.

Figure 3–7 shows a timing waveform for read and write operations in single-port mode with unregistered outputs. Registering the outputs of the RAM simply delays the q output by one clock cycle.

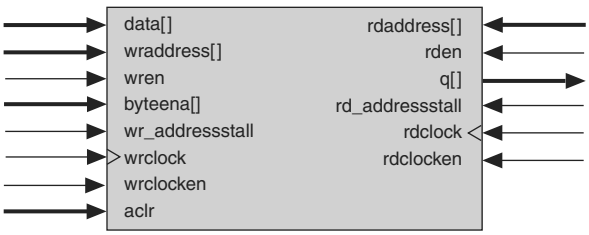
Figure 3–7. Cyclone IV Devices Single-Port Mode Timing Waveform



Simple Dual-Port Mode

Simple dual-port mode supports simultaneous read and write operations to different locations. Figure 3–8 shows the simple dual-port memory configuration.

Figure 3–8. Cyclone IV Devices Simple Dual-Port Memory *(1)*



Note to Figure 3–8:

(1) Simple dual-port RAM supports input or output clock mode in addition to the read or write clock mode shown.

Cyclone IV devices M9K memory blocks support mixed-width configurations, allowing different read and write port widths. Table 3–3 lists mixed-width configurations.

Table 3–3. Cyclone IV Devices M9K Block Mixed-Width Configurations (Simple Dual-Port Mode) ***(Part 1 of 2)***

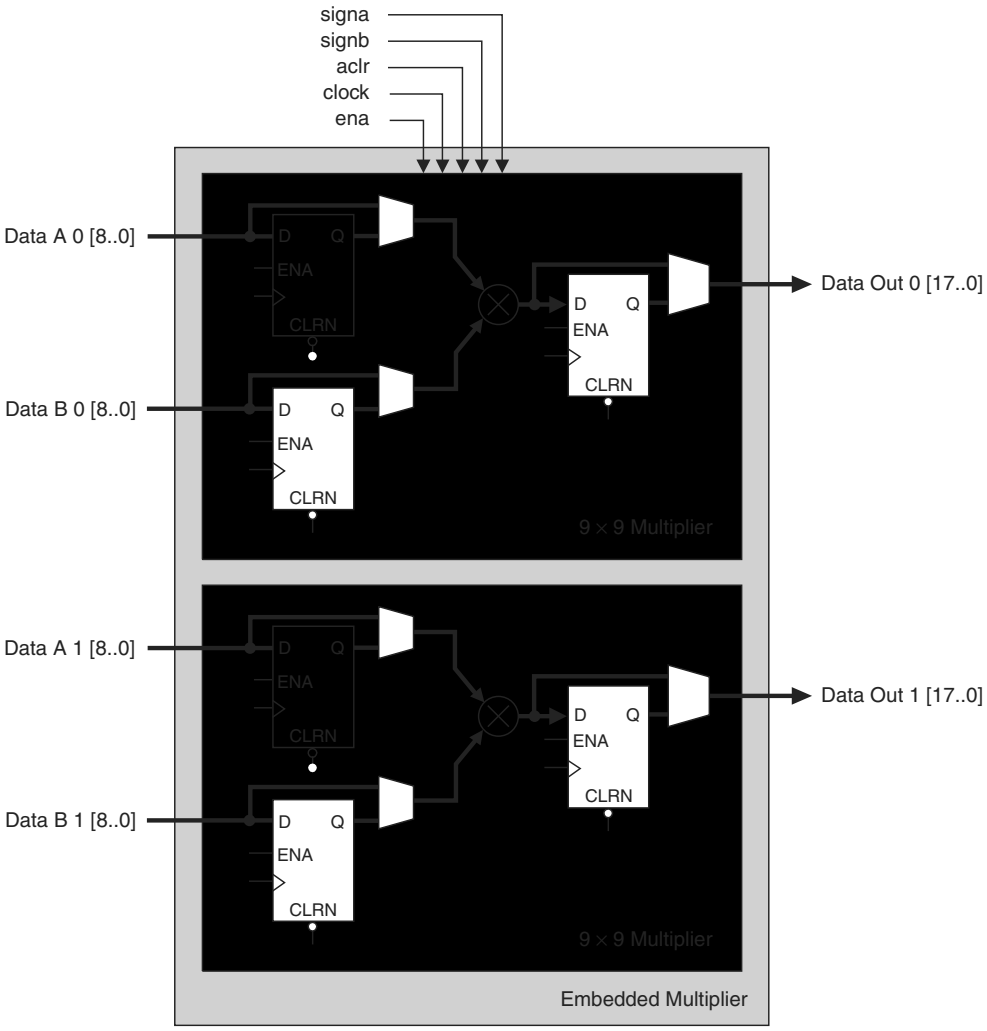
Read Port	Write Port								
	8192 × 1	4096 × 2	2048 × 4	1024 × 8	512 × 16	256 × 32	1024 × 9	512 × 18	256 × 36
8192 × 1	✓	✓	✓	✓	✓	✓	—	—	—
4096 × 2	✓	✓	✓	✓	✓	✓	—	—	—
2048 × 4	✓	✓	✓	✓	✓	✓	—	—	—
1024 × 8	✓	✓	✓	✓	✓	✓	—	—	—

9-Bit Multipliers

You can configure each embedded multiplier to support two 9 × 9 independent multipliers for input widths of up to 9 bits.

Figure 4–4 shows the embedded multiplier configured to support two 9-bit multipliers.

Figure 4–4. 9-Bit Multiplier Mode



All 9-bit multiplier inputs and results are independently sent through registers. The multiplier inputs can accept signed integers, unsigned integers, or a combination of both. Two 9 × 9 multipliers in the same embedded multiplier block share the same `signa` and `signb` signal. Therefore, all the `Data A` inputs feeding the same embedded multiplier must have the same sign representation. Similarly, all the `Data B` inputs feeding the same embedded multiplier must have the same sign representation.

Document Revision History

Table 4–3 lists the revision history for this chapter.

Table 4–3. Document Revision History

Date	Version	Changes
February 2010	1.1	Added Cyclone IV E devices in Table 4–1 for the Quartus II software version 9.1 SP1 release.
November 2009	1.0	Initial release.

Table 5–3. GCLK Network Connections for Cyclone IV E Devices (1) (Part 3 of 3)

GCLK Network Clock Sources	GCLK Networks																			
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
DPCLK2 (4) CDPCLK1, or CDPCLK2 (2), (5)	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
DPCLK5 (4) DPCLK7 (2)	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—
DPCLK4 (4) DPCLK6 (2)	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—
DPCLK6 (4) CDPCLK5, or CDPCLK6 (2), (5)	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—
DPCLK3 (4) CDPCLK4, or CDPCLK3 (2), (5)	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—
DPCLK8	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—
DPCLK11	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—
DPCLK9	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—
DPCLK10	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—
DPCLK5	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—
DPCLK2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—
DPCLK4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—
DPCLK3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓

Notes to Table 5–3:

(1) EP4CE6 and EP4CE10 devices only have GCLK networks 0 to 9.

(2) These pins apply to all Cyclone IV E devices except EP4CE6 and EP4CE10 devices.

(3) EP4CE6 and EP4CE10 devices only have PLL_1 and PLL_2.

(4) This pin applies only to EP4CE6 and EP4CE10 devices.

(5) Only one of the two CDPCLK pins can feed the clock control block. You can use the other pin as a regular I/O pin.

LVPECL I/O Support in Cyclone IV Devices

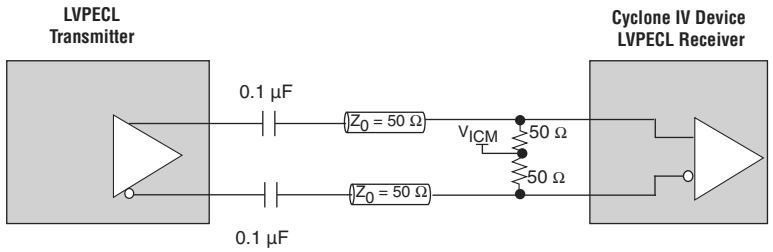
The LVPECL I/O standard is a differential interface standard that requires a 2.5-V V_{CCIO}. This standard is used in applications involving video graphics, telecommunications, data communications, and clock distribution. Cyclone IV devices support the LVPECL input standard at the dedicated clock input pins only. The LVPECL receiver requires an external 100-Ω termination resistor between the two signals at the input buffer.

For the LVPECL I/O standard electrical specification, refer to the *Cyclone IV Device Datasheet* chapter.

AC coupling is required when the LVPECL common mode voltage of the output buffer is higher than the Cyclone IV devices LVPECL input common mode voltage.

Figure 6–18 shows the AC-coupled termination scheme. The 50-Ω resistors used at the receiver are external to the device. DC-coupled LVPECL is supported if the LVPECL output common mode voltage is in the Cyclone IV devices LVPECL input buffer specification (refer to Figure 6–19).

Figure 6–18. LVPECL AC-Coupled Termination *(1)*

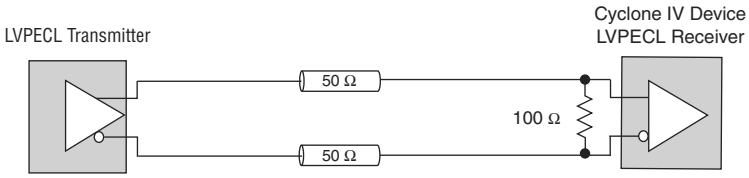


Note to Figure 6–18:

(1) The LVPECL AC-coupled termination is applicable only when an Altera FPGA transmitter is used.

Figure 6–19 shows the LVPECL DC-coupled termination.

Figure 6–19. LVPECL DC-Coupled Termination *(1)*



Note to Figure 6–19:

(1) The LVPECL DC-coupled termination is applicable only when an Altera FPGA transmitter is used.

Configuration Scheme

A configuration scheme with different configuration voltage standards is selected by driving the MSEL pins either high or low, as shown in Table 8–3, Table 8–4, and Table 8–5.

☞ Hardwire the MSEL pins to V_{CCA} or GND without pull-up or pull-down resistors to avoid problems detecting an incorrect configuration scheme. Do not drive the MSEL pins with a microprocessor or another device.

Table 8–3. Configuration Schemes for Cyclone IV GX Devices (EP4CGX15, EP4CGX22, and EP4CGX30 [except for F484 Package])

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

Notes to Table 8–3:

(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.

Table 8–4. Configuration Schemes for Cyclone IV GX Devices (EP4CGX30 [only for F484 package], EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150) (Part 1 of 2)

Configuration Scheme	MSEL3	MSEL2	MSEL1	MSEL0	POR Delay	Configuration Voltage Standard (V) (1)
AS	1	1	0	1	Fast	3.3
	1	0	1	1	Fast	3.0, 2.5
	1	0	0	1	Standard	3.3
	1	0	1	0	Standard	3.0, 2.5
PS	1	1	0	0	Fast	3.3, 3.0, 2.5
	1	1	1	0	Fast	1.8, 1.5
	1	0	0	0	Standard	3.3, 3.0, 2.5
	0	0	0	0	Standard	1.8, 1.5
FPP	0	0	1	1	Fast	3.3, 3.0, 2.5
	0	1	0	0	Fast	1.8, 1.5
	0	0	0	1	Standard	3.3, 3.0, 2.5
	0	0	1	0	Standard	1.8, 1.5

Altera recommends putting a buffer before the DATA and DCLK output from the master device to avoid signal strength and signal integrity issues. The buffer must not significantly change the DATA-to-DCLK relationships or delay them with respect to other AS signals (ASDI and nCS). Also, the buffer must only drive the slave devices to ensure that the timing between the master device and the serial configuration device is unaffected.

This configuration method supports both compressed and uncompressed **.sof**. Therefore, if the configuration bitstream size exceeds the capacity of a serial configuration device, you can enable the compression feature in the **.sof** or you can select a larger serial configuration device.

Guidelines for Connecting a Serial Configuration Device to Cyclone IV Devices for an AS Interface

For single- and multi-device AS configurations, the board trace length and loading between the supported serial configuration device and Cyclone IV device must follow the recommendations listed in Table 8–7.

Table 8–7. Maximum Trace Length and Loading for AS Configuration

Cyclone IV Device AS Pins	Maximum Board Trace Length from a Cyclone IV Device to a Serial Configuration Device (Inches)		Maximum Board Load (pF)
	Cyclone IV E	Cyclone IV GX	
DCLK	10	6	15
DATA[0]	10	6	30
nCSO	10	6	30
ASDO	10	6	30

Note to Table 8–7:

(1) For multi-devices AS configuration using Cyclone IV E with 1,0 V core voltage, the maximum board trace-length from the serial configuration device to the junction-split on both DCLK and Data0 line is 3.5 inches.

Estimating AS Configuration Time

AS configuration time is dominated by the time it takes to transfer data from the serial configuration device to the Cyclone IV device. This serial interface is clocked by the Cyclone IV device DCLK output (generated from a 40-MHz internal oscillator for Cyclone IV E devices, a 20- or 40-MHz internal oscillator, or an external CLKUSR of up to 40 MHz for Cyclone IV GX devices).

Equation 8–2 and Equation 8–3 show the configuration time calculations.

Equation 8–2.

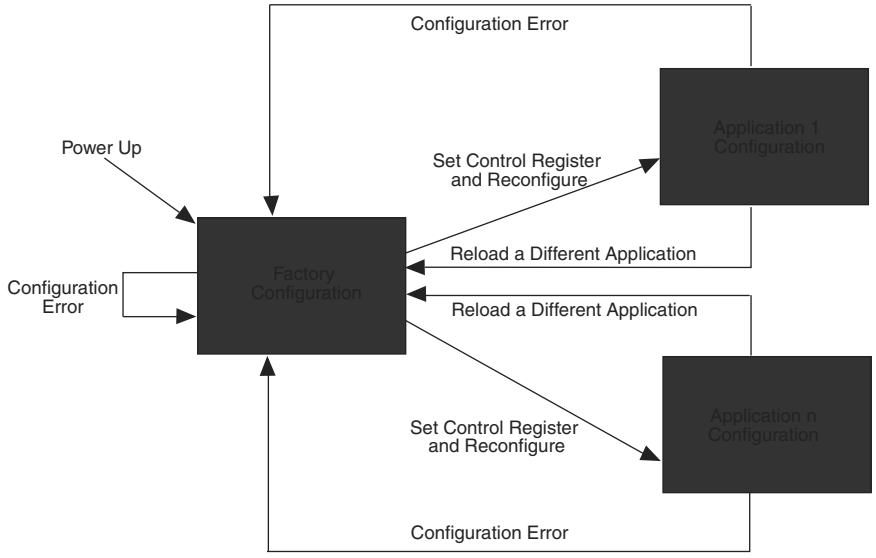
$$\text{Size} \times \left(\frac{\text{maximum DCLK period}}{\text{1 bit}}\right) = \text{estimated maximum configuration ti}$$

Equation 8–3.

$$9{,}600{,}000 \text{ bits} \times \left(\frac{50 \text{ ns}}{\text{1 bit}}\right) = 480 \text{ ms}$$

Figure 8–32 shows the transitions between the factory configuration and application configuration in remote update mode.

Figure 8–32. Transitions Between Configurations in Remote Update Mode



After power up or a configuration error, the factory configuration logic writes the remote system upgrade control register to specify the address of the application configuration to be loaded. The factory configuration also specifies whether or not to enable the user watchdog timer for the application configuration and, if enabled, specifies the timer setting.

☞ Only valid application configurations designed for remote update mode include the logic to reset the timer in user mode. For more information about the user watchdog timer, refer to the "User Watchdog Timer" on page 8–79.

If there is an error while loading the application configuration, the remote system upgrade status register is written by the dedicated remote system upgrade circuitry of the Cyclone IV device to specify the cause of the reconfiguration.

The following actions cause the remote system upgrade status register to be written:

- nSTATUS driven low externally
- Internal cyclical redundancy check (CRC) error
- User watchdog timer time-out
- A configuration reset (logic array nCONFIG signal or external nCONFIG pin assertion)

The Cyclone IV device automatically loads the factory configuration when an error occurs. This user-designed factory configuration reads the remote system upgrade status register to determine the reason for reconfiguration. Then the factory configuration takes the appropriate error recovery steps and writes to the remote system upgrade control register to determine the next application configuration to be loaded.

11. Power Requirements for Cyclone IV Devices

CYIV-51011-1.3

This chapter describes information about external power supply requirements, hot-socketing specifications, power-on reset (POR) requirements, and their implementation in Cyclone IV devices.

This chapter includes the following sections:

External Power Supply Requirements

This section describes the different external power supplies required to power Cyclone IV devices. Table 11–1 and Table 11–2 list the descriptions of external power supply pins for Cyclone IV GX and Cyclone IV E devices, respectively.

For each Altera recommended power supply's operating conditions, refer to the *Cyclone IV Device Datasheet* chapter.

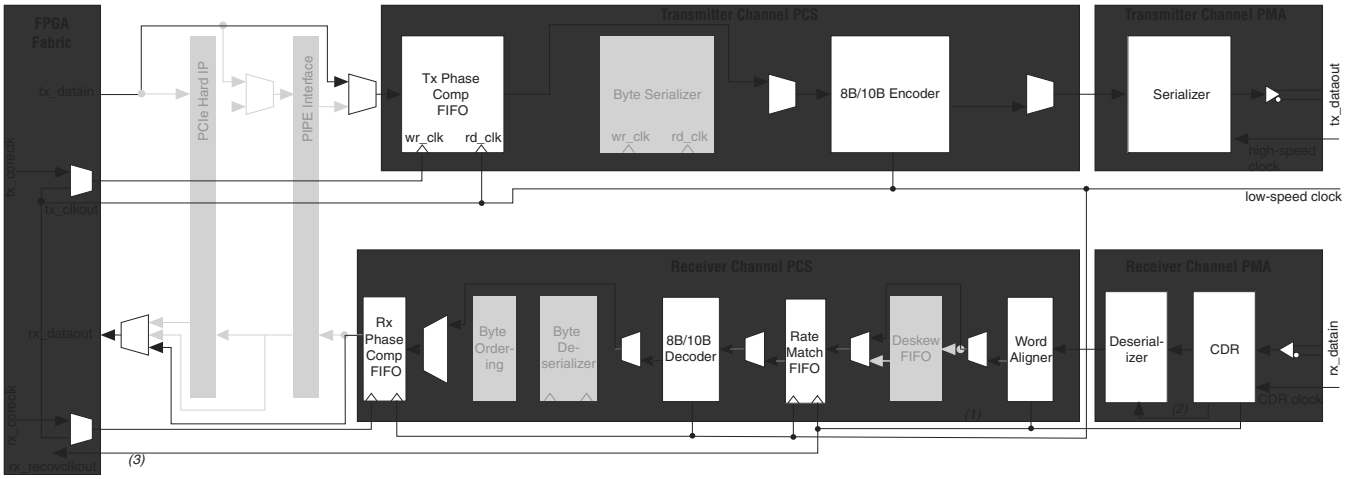
For power supply pin connection guidelines and power regulator sharing, refer to the *Cyclone IV Device Family Pin Connection Guidelines*.

Table 11–1. Power Supply Descriptions for the Cyclone IV GX Devices (Part 1 of 2)

Power Supply Pin	Nominal Voltage Level (V)	Description
VCCINT	1.2	Core voltage, PCI Express (PCIe) hard IP block, and transceiver physical coding sublayer (PCS) power supply
VCCA (1)	2.5	PLL analog power supply
VCCD_PLL	1.2	PLL digital power supply
VCCIO (2)	1.2, 1.5, 1.8, 2.5, 3.0, 3.3	I/O banks power supply
VCC_CLKIN (3), (4)	1.2, 1.5, 1.8, 2.5, 3.0, 3.3	Differential clock input pins power supply
VCCH_GXB	2.5	Transceiver output (TX) buffer power supply
VCCA_GXB	2.5	Transceiver physical medium attachment (PMA) and auxiliary power supply

Figure 1–55 shows the transceiver channel datapath and clocking when configured in GIGE mode.

Figure 1–55. Transceiver Channel Datapath and Clocking when Configured in GIGE Mode



Notes to Figure 1–55:

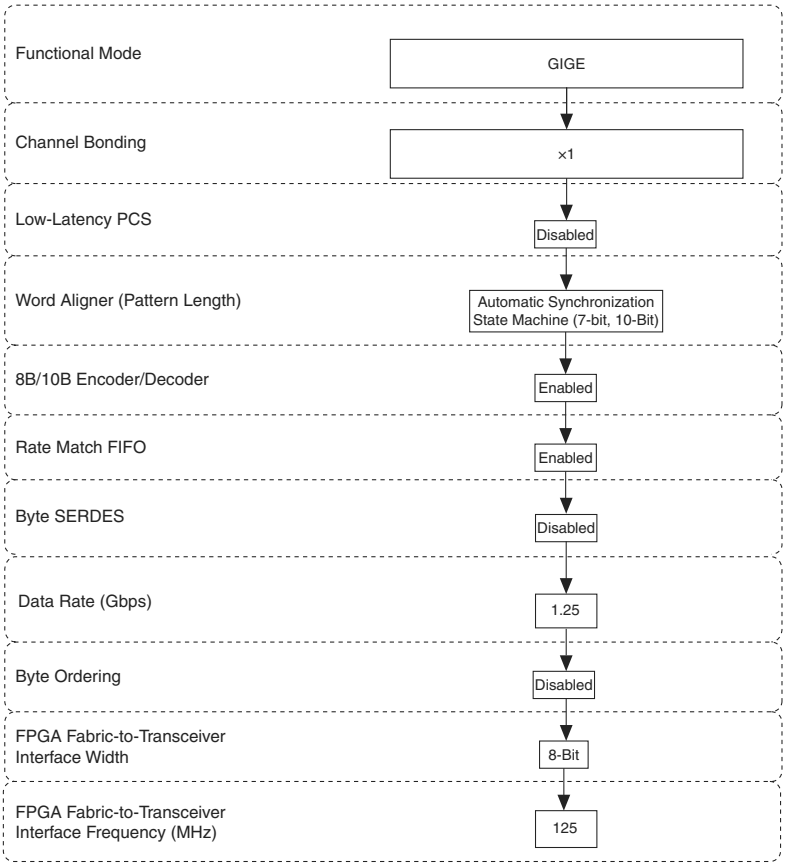
(1) Low-speed recovered clock.

(2) High-speed recovered clock.

(3) Optional `rx_recovclkout` port from CDR low-speed recovered clock is available for applications such as Synchronous Ethernet.

Figure 1–56 shows the transceiver configuration in GIGE mode.

Figure 1–56. Transceiver Configuration in GIGE Mode

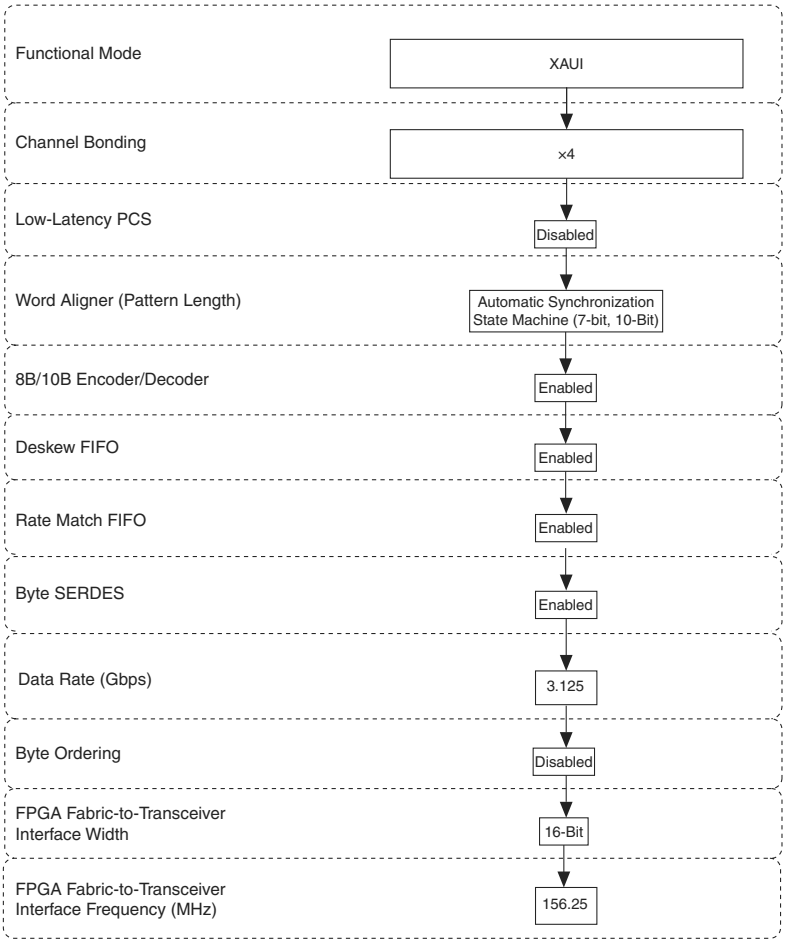


When configured in GIGE mode, three encoded comma (/K28.5/) code groups are transmitted automatically after deassertion of `tx_digitalreset` and before transmitting user data on the `tx_datain` port. This could affect the synchronization state machine behavior at the receiver.

Depending on when you start transmitting the synchronization sequence, there could be an even or odd number of encoded data (/Dx.y/) code groups transmitted between the last of the three automatically sent /K28.5/ code groups and the first /K28.5/ code group of the synchronization sequence. If there is an even number of /Dx.y/ code groups received between these two /K28.5/ code groups, the first /K28.5/ code group of the synchronization sequence begins at an odd code group boundary. An IEEE802.3-compliant GIGE synchronization state machine treats this as an error condition and goes into the Loss-of-Sync state.

Figure 1–64 shows the transceiver configuration in XAUI mode.

Figure 1–64. Transceiver Configuration in XAUI Mode



XGMII and PCS Code Conversions

In XAUI mode, the 8B/10B encoder in the transmitter datapath maps various 8-bit XGMII codes to 10-bit PCS code groups as listed in Table 1–21.

Table 1–21. XGMII Character to PCS Code Groups Mapping (Part 1 of 2)

XGMII TXC (1)	XGMII TXD (2), (3)	PCS Code Group	Description
0	00 through FF	Dxx,y	Normal data transmission
1	07	K28.0, K28.3, or K28.5	Idle in ||I||
1	07	K28.5	Idle in ||T||
1	9C	K28.4	Sequence
1	FB	K27.7	Start
1	FD	K29.7	Terminate
1	FE	K30.7	Error

Table 1–27. Receiver Ports in ALTGX Megafunction for Cyclone IV GX (Part 1 of 3)

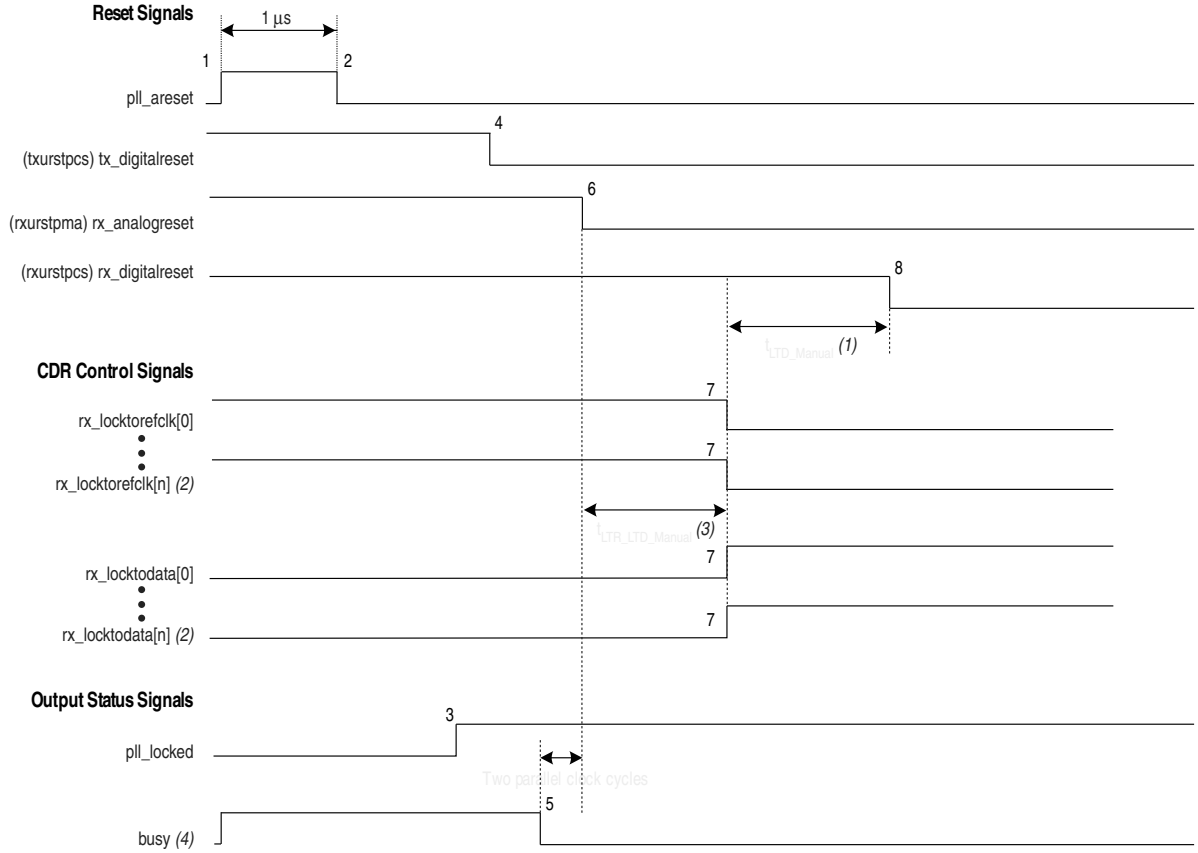
Block	Port Name	Input/ Output	Clock Domain	Description
RX PCS	rx_syncstatus	Output	Synchronous to `tx_clkout` (non-bonded modes with rate match FIFO), `rx_clkout` (non-bonded modes without rate match FIFO), `coreclkout` (bonded modes), or `rx_coreclk` (when using the optional `rx_coreclk` input)	Word alignment synchronization status indicator. This signal passes through the RX Phase Compensation FIFO. ■ Not available in bit-slip mode
	rx_patternde tect	Output	Synchronous to `tx_clkout` (non-bonded modes with rate match FIFO), `rx_clkout` (non-bonded modes without rate match FIFO), `coreclkout` (bonded modes), or `rx_coreclk` (when using the optional `rx_coreclk` input)	Indicates when the word alignment logic detects the alignment pattern in the current word boundary. This signal passes through the RX Phase Compensation FIFO.
	rx_bitslip	Input	Asynchronous signal. Minimum pulse width is two parallel clock cycles.	Bit-slip control for the word aligner configured in bit-slip mode. ■ At every rising edge, word aligner slips one bit into the received data stream, effectively shifting the word boundary by one bit.
	rx_rlv	Output	Asynchronous signal. Driven for a minimum of two recovered clock cycles in configurations without byte serializer and a minimum of three recovered clock cycles in configurations with byte serializer.	Run-length violation indicator. ■ A high pulse indicates that the number of consecutive 1s or 0s in the received data stream exceeds the programmed run length violation threshold.
	rx_invpolarity	Input	Asynchronous signal. Minimum pulse width is two parallel clock cycles.	Generic receiver polarity inversion control. ■ A high level to invert the polarity of every bit of the 8- or 10-bit data to the word aligner.
	rx_enapattern align	Input	Asynchronous signal.	Controls the word aligner operation configured in manual alignment mode.
	rx_rmfifodata inserted	Output	Synchronous to `tx_clkout` (non-bonded modes) or `coreclkout` (bonded modes)	Rate match FIFO insertion status indicator. ■ A high level indicates the rate match pattern byte is inserted to compensate for the ppm difference in the reference clock frequencies between the upstream transmitter and the local receiver.
	rx_rmfifodata deleted	Output	Synchronous to `tx_clkout` (non-bonded modes) or `coreclkout` (bonded modes)	Rate match FIFO deletion status indicator. ■ A high level indicates the rate match pattern byte is deleted to compensate for the ppm difference in the reference clock frequencies between the upstream transmitter and the local receiver.

4. For the receiver operation, after deassertion of `busy` signal, wait for two parallel clock cycles to deassert the `rx_analogreset` signal.
5. Wait for the `rx_freqlocked` signal from each channel to go high. The `rx_freqlocked` signal of each channel may go high at different times (indicated by the slashed pattern at marker 7).
6. In a bonded channel group, when the `rx_freqlocked` signals of all the channels has gone high, from that point onwards, wait for at least t_{LTD_Auto} time for the receiver parallel clock to be stable, then deassert the `rx_digitalreset` signal (marker 8). At this point, all the receivers are ready for data traffic.

Receiver and Transmitter Channel—Receiver CDR in Manual Lock Mode

This configuration contains both a transmitter and receiver channel. When the receiver CDR is in manual lock mode, use the reset sequence shown in Figure 2–5.

Figure 2–5. Sample Reset Sequence for Bonded Configuration Receiver and Transmitter Channels—Receiver CDR in Manual Lock Mode



Notes to Figure 2–5:

(1) For t_{LTD_Manual} duration, refer to the *Cyclone IV Device Datasheet* chapter.
(2) The number of `rx_locktorefclk[n]` and `rx_locktodata[n]` signals depend on the number of channels configured. `n`=number of channels.
(3) For $t_{LTR\_LTD\_Manual}$ duration, refer to the *Cyclone IV Device Datasheet* chapter.
(4) The `busy` signal is asserted and deasserted only during initial power up when offset cancellation occurs. In subsequent reset sequences, the `busy` signal is asserted and deasserted only if there is a read or write operation to the ALTGX_RECONFIG megafunction.

PCIe Initialization/Compliance Phase

After the device is powered up, a PCIe-compliant device goes through the compliance phase during initialization. The rx_digitalreset signal must be deasserted during this compliance phase to achieve transitions on the pipephydonestatus signal, as expected by the link layer. The rx_digitalreset signal is deasserted based on the assertion of the rx_freqlocked signal.

During the initialization/compliance phase, do not use the rx_freqlocked signal to trigger a deassertion of the rx_digitalreset signal. Instead, perform the following reset sequence:

1. After power up, assert pll_areset for a minimum period of 1 µs (the time between markers 1 and 2). Keep the tx_digitalreset, rx_analogreset, and rx_digitalreset signals asserted during this time period. After you deassert the pll_areset signal, the multipurpose PLL starts locking to the input reference clock.
2. After the multipurpose PLL locks, as indicated by the pll_locked signal going high (marker 3), deassert tx_digitalreset. For a receiver operation, after deassertion of busy signal, wait for two parallel clock cycles to deassert the rx_analogreset signal. After rx_analogreset is deasserted, the receiver CDR starts locking to the receiver input reference clock.
3. Deassert both the rx_analogreset signal (marker 6) and rx_digitalreset signal (marker 7) together, as indicated in Figure 2–10. After deasserting rx_digitalreset, the pipephydonestatus signal transitions from the transceiver channel to indicate the status to the link layer. Depending on its status, pipephydonestatus helps with the continuation of the compliance phase. After successful completion of this phase, the device enters into the normal operation phase.

PCIe Normal Phase

For the normal PCIe phase:

1. After completion of the Initialization/Compliance phase, during the normal operation phase at the Gen1 data rate, when the rx_freqlocked signal is deasserted (marker 9 in Figure 2–10).
2. Wait for the rx_freqlocked signal to go high again. In this phase, the received data is valid (not electrical idle) and the receiver CDR locks to the incoming data. Proceed with the reset sequence after assertion of the rx_freqlocked signal.
3. After the rx_freqlocked signal goes high, wait for at least t_{LTD_Manual} before asserting rx_digitalreset (marker 12 in Figure 2–10) for two parallel receive clock cycles so that the receiver phase compensation FIFO is initialized. For bonded PCIe Gen 1 mode (×2 and ×4), wait for all the rx_freqlocked signals to go high, then wait for t_{LTD_Manual} before asserting rx_digitalreset for 2 parallel clock cycles.

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

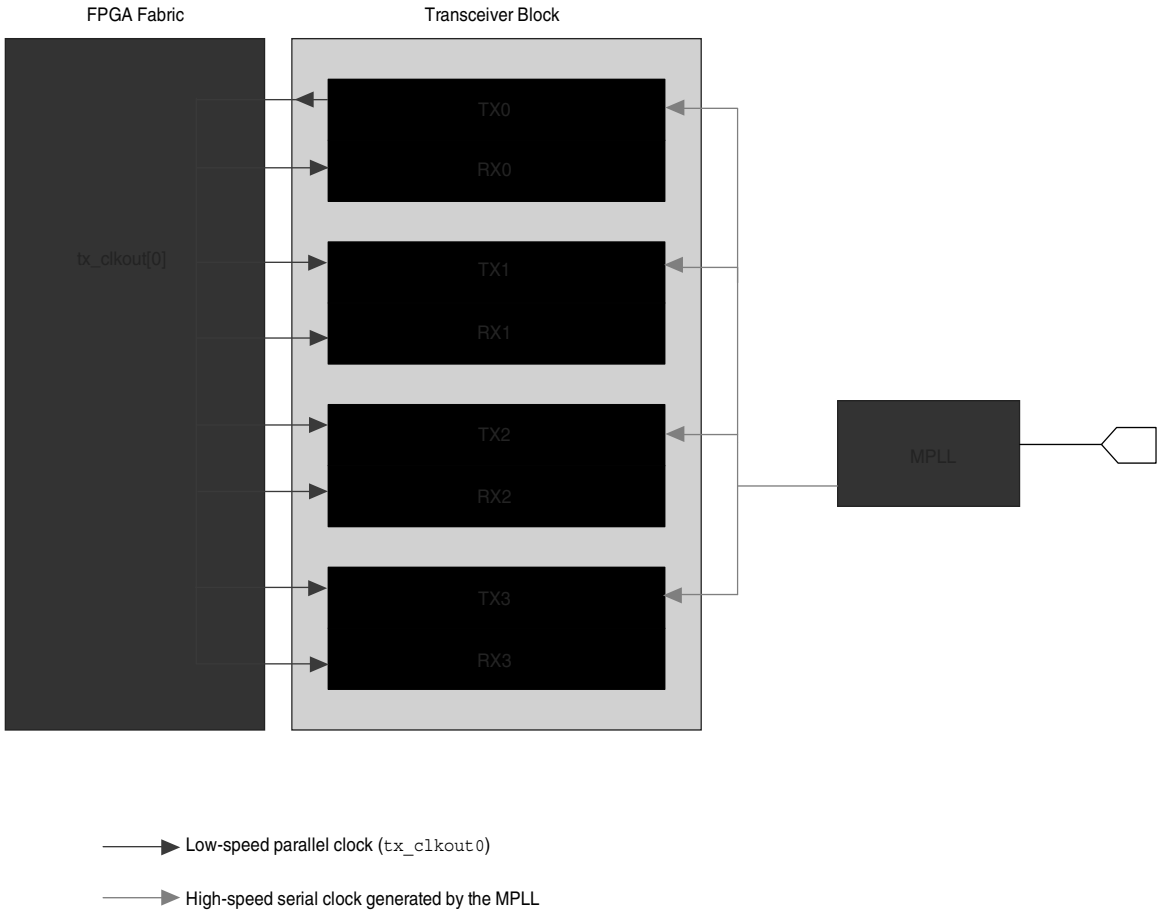
You can use the analog reconfiguration feature to dynamically reconfigure the transceivers channels setting in either the transmitter or the receivers in the PMA blocks. You can update the PMA controls on-the-fly based on the desired input. You can perform both read and write transaction separately for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

Option 1: Share a Single Transmitter Core Clock Between Receivers

- Enable this option if you want `tx_clkout` of the first channel (channel 0) of the transceiver block to provide the read clock to the Receive Phase Compensation FIFOs of the remaining receiver channels in the transceiver block.
- This option is typically enabled when all the channels of a transceiver block are in a Basic or Protocol configuration with rate matching enabled and are reconfigured to another Basic or Protocol configuration with rate matching enabled.

Figure 3–13 shows the sharing of channel 0's `tx_clkout` between all four channels of a transceiver block.

Figure 3–13. Option 1 for Receiver Core Clocking (Channel Reconfiguration 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.