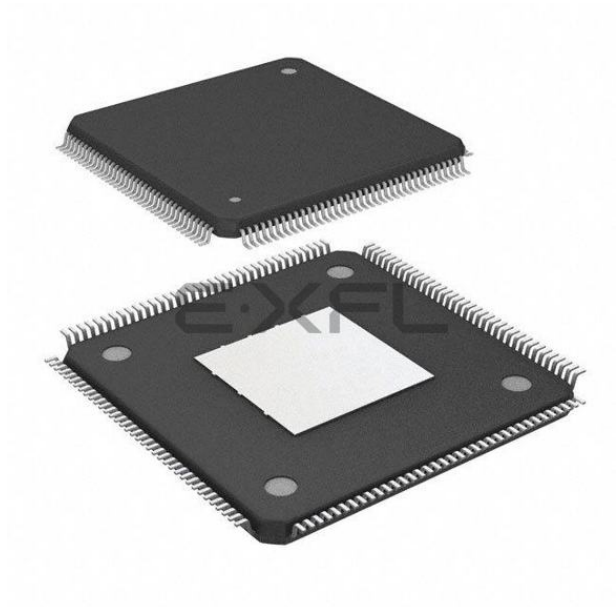


Welcome to E-XFL.COM

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.

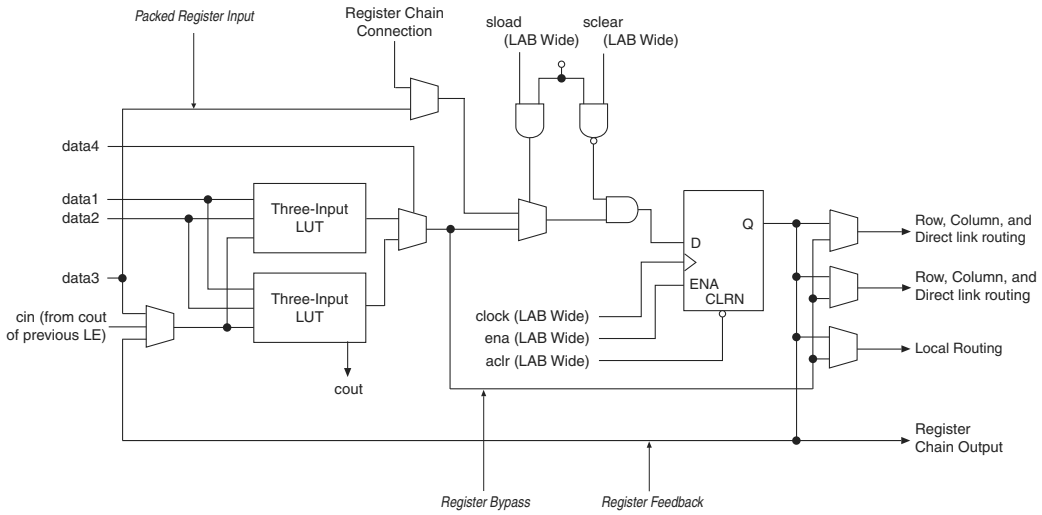
Details	
Product Status	Active
Number of LABs/CLBs	963
Number of Logic Elements/Cells	15408
Total RAM Bits	516096
Number of I/O	81
Number of Gates	-
Voltage - Supply	0.97V ~ 1.03V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	144-LQFP Exposed Pad
Supplier Device Package	144-EQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce15e22c9l

Arithmetic Mode

Arithmetic mode is ideal for implementing adders, counters, accumulators, and comparators. An LE in arithmetic mode implements a 2-bit full adder and basic carry chain (Figure 2–3). LEs in arithmetic mode can drive out registered and unregistered versions of the LUT output. Register feedback and register packing are supported when LEs are used in arithmetic mode.

Figure 2–3 shows LEs in arithmetic mode.

Figure 2–3. Cyclone IV Device LEs in Arithmetic Mode



The Quartus II Compiler automatically creates carry chain logic during design processing. You can also manually create the carry chain logic during design entry. Parameterized functions, such as LPM functions, automatically take advantage of carry chains for the appropriate functions.

The Quartus II Compiler creates carry chains longer than 16 LEs by automatically linking LABs in the same column. For enhanced fitting, a long carry chain runs vertically, which allows fast horizontal connections to M9K memory blocks or embedded multipliers through direct link interconnects. For example, if a design has a long carry chain in an LAB column next to a column of M9K memory blocks, any LE output can feed an adjacent M9K memory block through the direct link interconnect. If the carry chains run horizontally, any LAB which is not next to the column of M9K memory blocks uses other row or column interconnects to drive a M9K memory block. A carry chain continues as far as a full column.

Table 7–1 lists the number of DQS or DQ groups supported on each side of the Cyclone IV GX device.

Table 7–1. Cyclone IV GX Device DQS and DQ Bus Mode Support for Each Side of the Device

Device	Package	Side	Number ×8 Groups	Number ×9 Groups	Number ×16 Groups	Number ×18 Groups	Number ×32 Groups	Number ×36 Groups
EP4CGX15	169-pin FBGA	Right	1	0	0	0	—	—
		Top (1)	1	0	0	0	—	—
		Bottom (2)	1	0	0	0	—	—
EP4CGX22 EP4CGX30	169-pin FBGA	Right	1	0	0	0	—	—
		Top (1)	1	0	0	0	—	—
		Bottom (2)	1	0	0	0	—	—
	324-pin FBGA	Right	2	2	1	1	—	—
		Top	2	2	1	1	—	—
		Bottom	2	2	1	1	—	—
	484-pin FBGA (3)	Right	4	2	2	2	1	1
		Top	4	2	2	2	1	1
		Bottom	4	2	2	2	1	1
EP4CGX50 EP4CGX75	484-pin FBGA	Right	4	2	2	2	1	1
		Top	4	2	2	2	1	1
		Bottom	4	2	2	2	1	1
	672-pin FBGA	Right	4	2	2	2	1	1
		Top	4	2	2	2	1	1
		Bottom	4	2	2	2	1	1
EP4CGX110 EP4CGX150	484-pin FBGA	Right	4	2	2	2	1	1
		Top	4	2	2	2	1	1
		Bottom	4	2	2	2	1	1
	672-pin FBGA	Right	4	2	2	2	1	1
		Top	4	2	2	2	1	1
		Bottom	4	2	2	2	1	1
	896-pin FBGA	Right	6	3	2	2	1	1
		Top	6	3	3	3	1	1
		Bottom	6	3	3	3	1	1

Notes to Table 7–1:

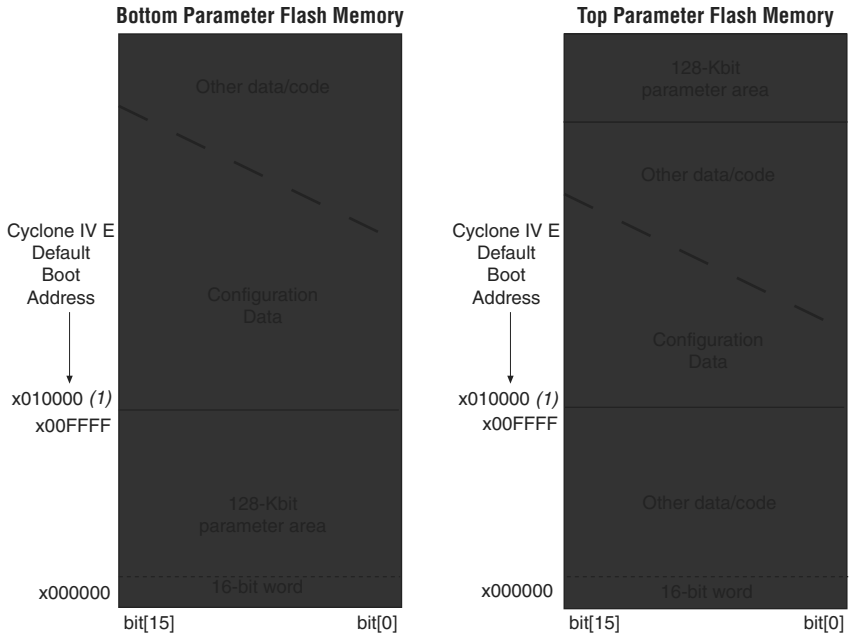
(1) Some of the DQ pins can be used as RUP and RDN pins. You cannot use these groups if you are using these pins as RUP and RDN pins for OCT calibration.

(2) Some of the DQ pins can be used as RUP pins while the DM pins can be used as RDN pins. You cannot use these groups if you are using the RUP and RDN pins for OCT calibration.

(3) Only available for EP4CGX30 device.

The default configuration boot address allows the system to use special parameter blocks in the flash memory map. Parameter blocks are at the top or bottom of the memory map. Figure 8–12 shows the configuration boot address in the AP configuration scheme. You can change the default configuration default boot address 0×010000 to any desired address using the APFC_BOOT_ADDR JTAG instruction. For more information about the APFC_BOOT_ADDR JTAG instruction, refer to "JTAG Instructions" on page 8–57.

Figure 8–12. Configuration Boot Address in AP Flash Memory Map



Note to Figure 8–12:

(1) The default configuration boot address is x010000 when represented in 16-bit word addressing.

PS Configuration

You can perform PS configuration on Cyclone IV devices with an external intelligent host, such as a MAX® II device, microprocessor with flash memory, or a download cable. In the PS scheme, an external host controls the configuration. Configuration data is clocked into the target Cyclone IV device through DATA[0] at each rising edge of DCLK.

If your system already contains a common flash interface (CFI) flash memory, you can use it for Cyclone IV device configuration storage as well. The MAX II PFL feature provides an efficient method to program CFI flash memory devices through the JTAG interface and the logic to control the configuration from the flash memory device to the Cyclone IV device.

For more information about the PFL, refer to *AN 386: Using the Parallel Flash Loader with the Quartus II Software*.

Cyclone IV devices do not support enhanced configuration devices for PS configuration.

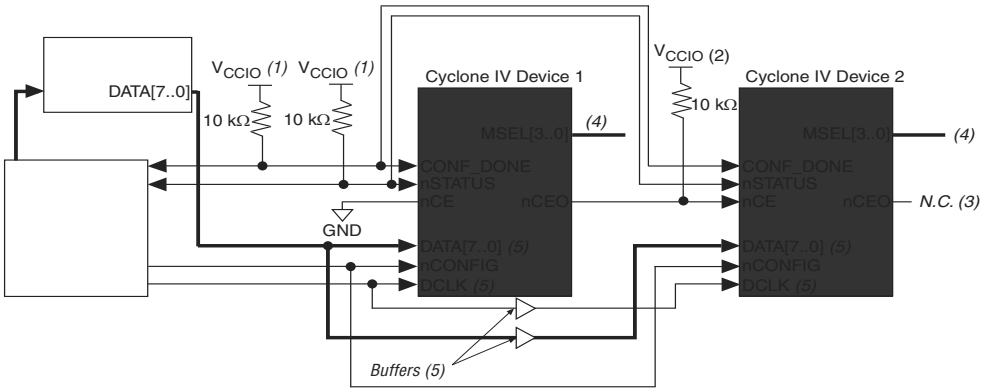
To ensure that DCLK and DATA[0] are not left floating at the end of the configuration, the MAX II device must drive them either high or low, whichever is convenient on your board. The DATA[0] pin is available as a user I/O pin after configuration. When you choose the FPP scheme in the Quartus II software, 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 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 the CONF_DONE and INIT_DONE pins 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 8–20 shows how to configure multiple devices with a MAX II device. This circuit is similar to the FPP configuration circuit for a single device, except the Cyclone IV devices are cascaded for multi-device configuration.

Figure 8–20. Multi-Device FPP Configuration Using an External Host



Notes to Figure 8–20:

(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 8–4 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[7..0] and DCLK must fit the maximum overshoot outlined in Equation 8–1 on page 8–5.

After the first device completes configuration in a multi-device configuration chain, its nCEO pin drives low to activate the nCE pin of the second device, which prompts the second device to begin configuration. The second device in the chain begins configuration in one clock cycle; therefore, the transfer of data destinations is transparent to the MAX II device. All other configuration pins (nCONFIG, nSTATUS,

Reconfiguration

After the configuration data is successfully written into the serial configuration device, the Cyclone IV device does not automatically start reconfiguration. The intelligent host issues the PULSE_NCONFIG JTAG instruction to initialize the reconfiguration process. During reconfiguration, the master device is reset and the SFL design no longer exists in the Cyclone IV device and the serial configuration device configures all the devices in the chain with the user design.

For more information about the SFL, refer to *AN 370: Using the Serial FlashLoader with Quartus II Software.*

JTAG Instructions

For more information about the JTAG binary instruction code, refer to the *JTAG Boundary-Scan Testing for Cyclone IV Devices* chapter.

I/O Reconfiguration

Use the CONFIG_IO instruction to reconfigure the I/O configuration shift register (IOCSR) chain. This instruction allows you to perform board-level testing prior to configuring the Cyclone IV device or waiting for a configuration device to complete configuration. After the configuration is interrupted and JTAG testing is complete, you must reconfigure the part through the PULSE_NCONFIG JTAG instruction or by pulsing the nCONFIG pin low.

You can issue the CONFIG_IO instruction any time during user mode.

You must meet the following timing restrictions when using the CONFIG_IO instruction:

- The CONFIG_IO instruction cannot be issued when the nCONFIG pin is low
- You must observe a 230 µs minimum wait time after any of the following conditions:
  - nCONFIG pin goes high
  - Issuing the PULSE_NCONFIG instruction
  - Issuing the ACTIVE_ENGAGE instruction, before issuing the CONFIG_IO instruction
- You must wait 230 µs after power up, with the nCONFIG pin high before issuing the CONFIG_IO instruction (or wait for the nSTATUS pin to go high)

- External configuration reset (nCONFIG) assertion
- User watchdog timer time out

Table 8–24 lists the contents of the current state logic in the status register, when the remote system upgrade master state machine is in factory configuration or application configuration accessing the factory information or application information, respectively. The status register bit in Table 8–24 lists the bit positions in a 32-bit logic.

Table 8–24. Remote System Upgrade Current State Logic Contents In Status Register

Remote System Upgrade Master State Machine	Status Register Bit	Definition	Description
Factory information (1)	31:30	Master state machine current state	The current state of the remote system upgrade master state machine
	29:24	Reserved bits	Padding bits that are set to all 0's
	23:0	Boot address	The current 24-bit boot address that was used by the configuration scheme as the start address to load the current configuration.
Application information 1 (2)	31:30	Master state machine current state	The current state of the remote system upgrade master state machine
	29	User watchdog timer enable bit	The current state of the user watchdog enable, which is active high
	28:0	User watchdog timer time-out value	The current entire 29-bit watchdog time-out value.
Application information 2 (2)	31:30	Master state machine current state	The current state of the remote system upgrade master state machine
	29:24	Reserved bits	Padding bits that are set to all 0's
	23:0	Boot address	The current 24-bit boot address that was used as the start address to load the current configuration

Notes to Table 8–24:

(1) The remote system upgrade master state machine is in factory configuration.

(2) The remote system upgrade master state machine is in application configuration.

The previous two application configurations are available in the previous state registers (previous state register 1 and previous state register 2), but only for debugging purposes.

IEEE Std. 1149.6 Boundary-Scan Register

The boundary-scan cell (BSC) for HSSI transmitters (GXB_TX[p,n]) and receivers (GXB_RX[p,n]) in Cyclone IV GX devices are different from the BSCs for I/O pins.

Figure 10–1 shows the Cyclone IV GX HSSI transmitter boundary-scan cell.

Figure 10–1. HSSI Transmitter BSC with IEEE Std. 1149.6 BST Circuitry for Cyclone IV GX Devices

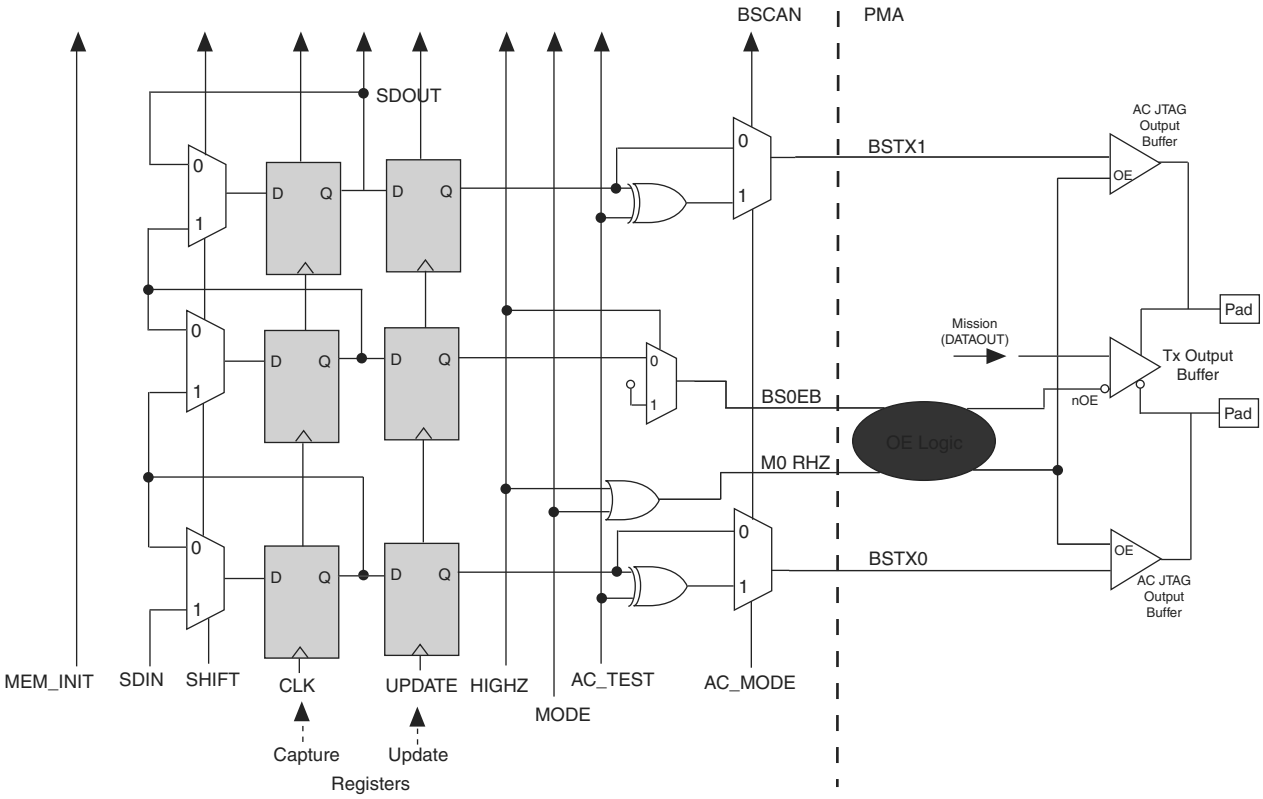


Table 11–1. Power Supply Descriptions for the Cyclone IV GX Devices (Part 2 of 2)

Power Supply Pin	Nominal Voltage Level (V)	Description
VCCL_GXB	1.2	Transceiver PMA and auxiliary power supply

Notes to Table 11–1:

(1) You must power up VCCA even if the phase-locked loop (PLL) is not used.

(2) I/O banks 3, 8, and 9 contain configuration pins. You can only power up the V_{CCIO} level of I/O banks 3 and 9 to 1.5 V, 1.8 V, 2.5 V, 3.0 V, or 3.3 V. For Fast Passive Parallel (FPP) configuration mode, you must power up the V_{CCIO} level of I/O bank 8 to 1.5 V, 1.8 V, 2.5 V, 3.0 V, or 3.3 V.

(3) All device packages of EP4CGX15, EP4CGX22, and device package F169 and F324 of EP4CGX30 devices have two VCC_CLKIN dedicated clock input I/O located at Banks 3A and 8A. Device package F484 of EP4CGX30, all device packages of EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 devices have four VCC_CLKIN dedicated clock input I/O bank located at banks 3A, 3B, 8A, and 8B.

(4) You must set VCC_CLKIN to 2.5V if the CLKIN is used as a high-speed serial interface (HSSI) transceiver refclk. When not used as a transceiver refclk, VCC_CLKIN supports 1.2 V/ 1.5 V/ 1.8 V/ 2.5 V/ 3.0 V/ 3.3V voltages.

Table 11–2. Power Supply Descriptions for the Cyclone IV E Devices

Power Supply Pin	Nominal Voltage Level (V)	Description
VCCINT	1.0, 1.2	Core voltage power supply
VCCA *(1)*	2.5	PLL analog power supply
VCCD_PLL	1.0, 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

Notes to Table 11–2:

(1) You must power up VCCA even if the PLL is not used.

(2) I/O banks 1, 6, 7, and 8 contain configuration pins.

Hot-Socketing Specifications

Cyclone IV devices are hot-socketing compliant without the need for any external components or special design requirements. Hot-socketing support in Cyclone IV devices has the following advantages:

- You can drive the device before power up without damaging the device.
- I/O pins remain tri-stated during power up. The device does not drive out before or during power-up. Therefore, it does not affect other buses in operation.

Devices Driven Before Power-Up

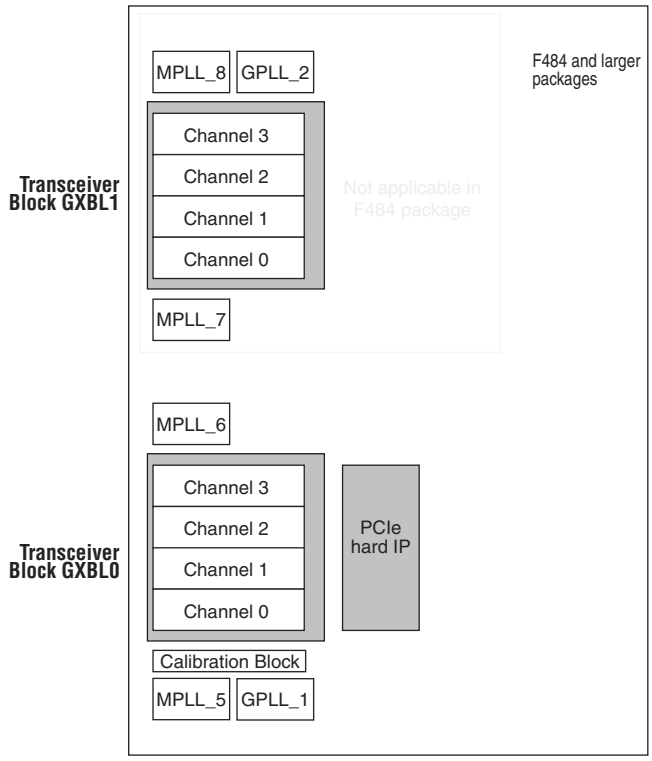
You can drive signals into regular Cyclone IV E I/O pins and transceiver Cyclone IV GX I/O pins before or during power up or power down without damaging the device. Cyclone IV devices support any power-up or power-down sequence to simplify system-level designs.

I/O Pins Remain Tri-stated During Power-Up

The output buffers of Cyclone IV devices are turned off during system power up or power down. Cyclone IV devices do not drive out until the device is configured and working in recommended operating conditions. The I/O pins are tri-stated until the device enters user 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`. Indicates command line commands and anything that must be typed exactly as it appears. For example, `c:\qdesigns\tutorial\chiptrip.gdf`. 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 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.

Figure 1–2. F484 and Larger Packages with Transceiver Channels for Cyclone IV GX Devices



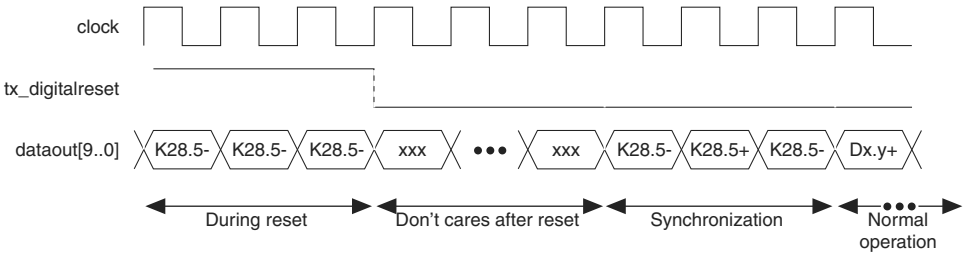
For more information about the transceiver architecture, refer to the following sections:

- "Architectural Overview" on page 1–4
- "Transmitter Channel Datapath" on page 1–5
- "Receiver Channel Datapath" on page 1–11
- "Transceiver Clocking Architecture" on page 1–26
- "Transceiver Channel Datapath Clocking" on page 1–29
- "FPGA Fabric-Transceiver Interface Clocking" on page 1–43
- "Calibration Block" on page 1–45
- "PCI-Express Hard IP Block" on page 1–46

The following describes the 8B/10B encoder behavior in reset condition (as shown in Figure 1–7):

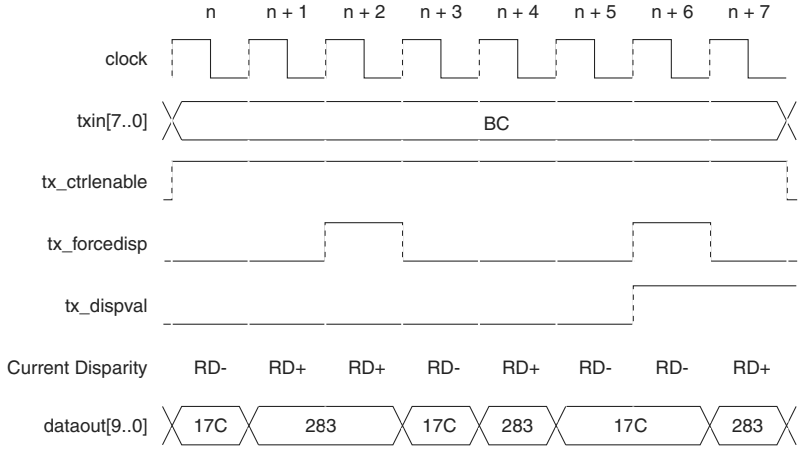
- During reset, the 8B/10B encoder ignores the inputs (`tx_datain` and `tx_ctrlenable` ports) from the FPGA fabric and outputs the K28.5 pattern from the RD- column continuously until the `tx_digitalreset` port is deasserted.
- Upon deassertion of the `tx_digitalreset` port, the 8B/10B encoder starts with a negative disparity and transmits three K28.5 code groups for synchronization before it starts encoding and transmitting data on its output.
- Due to some pipelining of the transmitter PCS, some "don't cares" (10'hxxx) are sent before the three synchronizing K28.5 code groups.

Figure 1–7. 8B/10B Encoder Behavior in Reset Condition



The encoder supports forcing the running disparity to either positive or negative disparity with `tx_forcedisp` and `tx_dispval` ports. Figure 1–8 shows an example of `tx_forcedisp` and `tx_dispval` port use, where data is shown in hexadecimal radix.

Figure 1–8. Force Running Disparity Operation



In this example, a series of K28.5 code groups are continuously sent. The stream alternates between a positive disparity K28.5 (RD+) and a negative disparity K28.5 (RD-) to maintain a neutral overall disparity. The current running disparity at time $n + 1$ indicates that the K28.5 in time $n + 2$ should be encoded with a negative disparity. Because `tx_forcedisp` is high at time $n + 2$, and `tx_dispval` is low, the K28.5

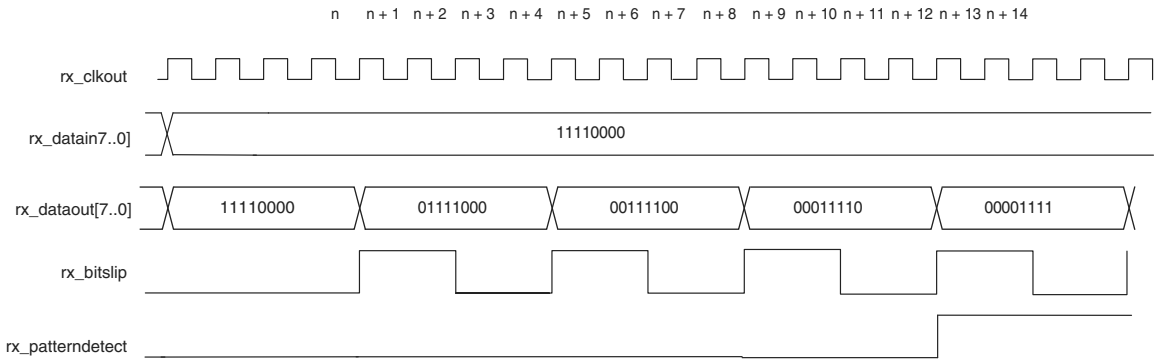
Bit-Slip Mode

In bit-slip mode, the rx_bitslip port controls the word aligner operation. At every rising edge of the rx_bitslip signal, the bit-slip circuitry slips one bit into the received data stream, effectively shifting the word boundary by one bit. When the received data after bit-slipping matches the programmed word alignment pattern, the rx_patterndetect signal is driven high for one parallel clock cycle.

☞ You can implement a bit-slip controller in the user logic that monitors either the rx_patterndetect signal or the receiver data output (rx_dataout), and controls the rx_bitslip port to achieve word alignment.

Figure 1–18 shows an example of the word aligner configured in bit-slip mode. For this example, consider that 8'b11110000 is received back-to-back and 16'b0000111100011110 is specified as the word alignment pattern. A rising edge on the rx_bitslip signal at time n + 1 slips a single bit 0 at the MSB position, forcing the rx_dataout to 8'b01111000. Another rising edge on the rx_bitslip signal at time n + 5 forces rx_dataout to 8'b00111100. Another rising edge on the rx_bitslip signal at time n + 9 forces rx_dataout to 8'b00011110. Another rising edge on the rx_bitslip signal at time n + 13 forces the rx_dataout to 8'b00001111. At this instance, rx_dataout in cycles n + 12 and n + 13 is 8'b00011110 and 8'b00001111, respectively, which matches the specified 16-bit alignment pattern 16'b0000111100011110. This results in the assertion of the rx_patterndetect signal.

Figure 1–18. Word Aligner Configured in Bit-Slip Mode



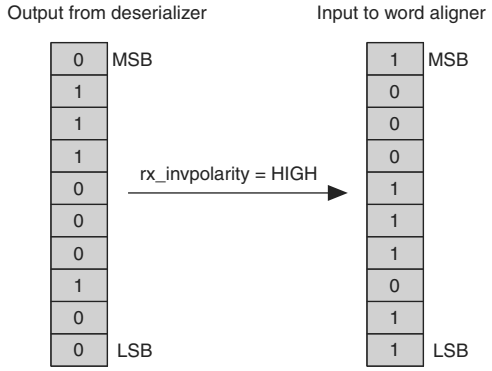
Automatic Synchronization State Machine Mode

In automatic synchronization state machine mode, the word aligner achieves synchronization after receiving a specific number of synchronization code groups, and falls out of synchronization after receiving a specific number of erroneous code groups. This mode provides hysteresis during link synchronization, which is required by protocols such as PCIe, GbE, XAUI, and Serial RapidIO.

☞ This mode is only supported using the 8B/10B encoded data with 10-bit input to the word aligner.

- Receiver polarity inversion—corrects accidental swapped positive and negative signals from the serial differential link during board layout. This feature works by inverting the polarity of every bit of the input data word to the word aligner, which has the same effect as swapping the positive and negative signals of the differential link. Inversion is dynamically controlled using `rx_invpolarity` port. Figure 1–19 shows the receiver polarity inversion feature.

Figure 1–19. Receiver Polarity Inversion



The generic receiver polarity inversion feature is different from the PCI Express (PIPE) 8B/10B polarity inversion feature. The generic receiver polarity inversion feature inverts the polarity of the data bits at the input of the word aligner and is not available in PCI Express (PIPE) mode. The PCI Express (PIPE) 8B/10B polarity inversion feature inverts the polarity of the data bits at the input of the 8B/10B decoder and is available only in PCI Express (PIPE) mode.

The `rx_invpolarity` signal is dynamic and might cause initial disparity errors in an 8B/10B encoded link. The downstream system must be able to tolerate these disparity errors.

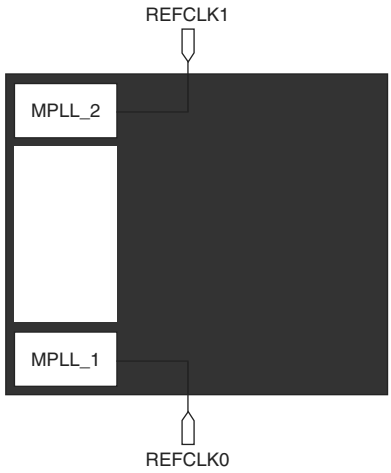
- Receiver bit reversal—by default, the Cyclone IV GX receiver assumes LSB to MSB transmission. If the link transmission order is MSB to LSB, the receiver forwards the incorrect reverse bit-ordered version of the parallel data to the FPGA fabric on the `rx_dataout` port. The receiver bit reversal feature is available to correct this situation. This feature is static in manual alignment and automatic

Input Reference Clocking

When used for transceiver, the left PLLs synthesize the input reference clock to generate the required clocks for the transceiver channels. Figure 1–25 and Figure 1–26 show the sources of input reference clocks for PLLs used in the transceiver operation.

☞ Clock output from PLLs in the FPGA core cannot feed into PLLs used by the transceiver as input reference clock.

Figure 1–25. PLL Input Reference Clocks in Transceiver Operation for F324 and Smaller Packages (1), (2)



Notes to Figure 1–25:

(1) The REFCLK0 and REFCLK1 pins are dual-purpose CLK, REFCLK, or DIFFCLK pins that reside in banks 3A and 8A respectively.

(2) Using any clock input pins other than the designated REFCLK pins as shown here to drive the MPLLs may have reduced jitter performance.

☞ In any configuration, a receiver channel cannot source CDR clocks from other PLLs beyond the two multipurpose PLLs directly adjacent to transceiver block where the channel resides.

The Cyclone IV GX transceivers support non-bonded (×1) and bonded (×2 and ×4) channel configurations. The two configurations differ in regards to clocking and phase compensation FIFO control. Bonded configuration provides a relatively lower channel-to-channel skew between the bonded channels than in non-bonded configuration. Table 1–8 lists the supported conditions in non-bonded and bonded channel configurations.

Table 1–8. Supported Conditions in Non-Bonded and Bonded Channel Configurations

Channel Configuration	Description	Supported Channel Operation Mode
Non-bonded (×1)	■ Low-speed clock in each channel is sourced independently ■ Phase compensation FIFO in each channel has its own pointers and control logic	■ Transmitter Only ■ Receiver Only ■ Transmitter and Receiver
Bonded (×2 and ×4)	■ Low-speed clock in each bonded channel is sourced from a common bonded clock path for lower channel-to-channel skew ■ Phase compensation FIFOs in bonded channels share common pointers and control logic for equal latency through the FIFOs in all bonded channels ■ ×2 bonded configuration is supported with channel 0 and channel 1 in a transceiver block ■ ×4 bonded configuration is supported with all four channels in a transceiver block	■ Transmitter Only ■ Transmitter and Receiver

Non-Bonded Channel Configuration

In non-bonded channel configuration, the high- and low-speed clocks for each channel are sourced independently. The phase compensation FIFOs in each channel has its own pointers and control logic. When implementing multi-channel serial interface in non-bonded channel configuration, the clock skew and unequal latency results in larger channel-to-channel skew.

☞ Altera recommends using bonded channel configuration (×2 or ×4) when implementing multi-channel serial interface for a lower channel-to-channel skew.

In a transceiver block, the high- and low-speed clocks for each channel are distributed primarily from one of the two multipurpose PLLs directly adjacent to the block. Transceiver channels for devices in F484 and larger packages support additional clocking flexibility. In these packages, some channels support high-speed and low-speed clock distribution from PLLs beyond the two multipurpose PLLs directly adjacent to the block.

Clock Frequency Compensation

In Serial RapidIO mode, the rate match FIFO compensates up to ±100 ppm (200 ppm total) difference between the upstream transmitter and the local receiver reference clock.

Rate matcher is an optional block available for selection in Serial RapidIO mode. However, this block is not fully compliant to the SRIO specification. When enabled in the ALTGX MegaWizard Plug-In Manager, the default settings are:

- control pattern 1 = K28.5 with positive disparity
- skip pattern 1 = K29.7 with positive disparity
- control pattern 2 = K28.5 with negative disparity
- skip pattern 2 = K29.7 with negative disparity

When enabled, the rate match FIFO operation begins after the link is synchronized (indicated by assertion of `rx_syncstatus` from the word aligner). When the rate matcher receives either of the two 10-bit control patterns followed by the respective 10-bit skip pattern, it inserts or deletes the 10-bit skip pattern as necessary to avoid the rate match FIFO from overflowing or under-running. The rate match FIFO can delete/insert a maximum of one skip pattern from a cluster.

☞ The rate match FIFO may perform multiple insertion or deletion if the ppm difference is more than the allowable 200 ppm range. Ensure that the ppm difference in your system is less than 200 ppm.

XAUI Mode

XAUI mode provides the bonded (×4) transceiver channel datapath configuration for XAUI protocol implementation. The Cyclone IV GX transceivers configured in XAUI mode provides the following functions:

- XGMII-to-PCS code conversion at transmitter datapath
- PCS-to-XGMII code conversion at receiver datapath
- channel deskewing of four lanes
- 8B/10B encoding and decoding
- IEEE P802.3ae-compliant synchronization state machine
- clock rate compensation

The XAUI is a self-managed interface to transparently extend the physical reach of the XGMII between the reconciliation sublayer and the PHY layer in the 10 Gbps LAN as shown in Figure 1–62. The XAUI interface consists of four lanes, each running at 3.125 Gbps with 8B/10B encoded data for a total of actual 10 Gbps data throughput. At the transmit side of the XAUI interface, the data and control characters are

Table 1–28. PIPE Interface Ports in ALTGX Megafunction for Cyclone IV GX (1) (Part 1 of 2)

Port Name	Input/ Output	Clock Domain	Description
fixedclk	Input	Clock signal	125-MHz clock for receiver detect and offset cancellation only in PIPE mode.
tx_detectrxloop	Input	Asynchronous signal	Receiver detect or reverse parallel loopback control. ■ A high level in the P1 power state and `tx_forceelecidle` signal asserted begins the receiver detection operation to determine if there is a valid receiver downstream. This signal must be deasserted when the `pipephydonestatus` signal indicates receiver detect completion. ■ A high level in the P0 power state with the `tx_forceelecidle` signal deasserted dynamically configures the channel to support reverse parallel loopback mode.
tx_forcedisp compliance	Input	Asynchronous signal	Force the 8B/10B encoder to encode with negative running disparity. ■ Assert only when transmitting the first byte of the PIPE-compliance pattern to force the 8B/10B encoder with a negative running disparity.
pipe8b10binvpolarity	Input	Asynchronous signal	Invert the polarity of every bit of the 10-bit input to the 8B/10B decoder
powerdn	Input	Asynchronous signal	PIPE power state control. ■ Signal is 2 bits wide and is encoded as follows: ■ 2'b00: P0 (Normal operation) ■ 2'b01: P0s (Low recovery time latency, low power state) ■ 2'b10: P1 (Longer recovery time latency, lower power state) ■ 2'b11: P2 (Lowest power state)
pipedatavalid	Output	N/A	Valid data and control on the `rx_dataout` and `rx_ctrldetect` ports indicator.
pipephydone status	Output	Asynchronous signal	PHY function completion indicator. ■ Asserted for one clock cycle to communicate completion of several PHY functions, such as power state transition and receiver detection.
pipeelecidle	Output	Asynchronous signal	Electrical idle detected or inferred at the receiver indicator. ■ When electrical idle inference is used, this signal is driven high when it infers an electrical idle condition ■ When electrical idle inference is not used, the `rx_signaldetect` signal is inverted and driven on this port.

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

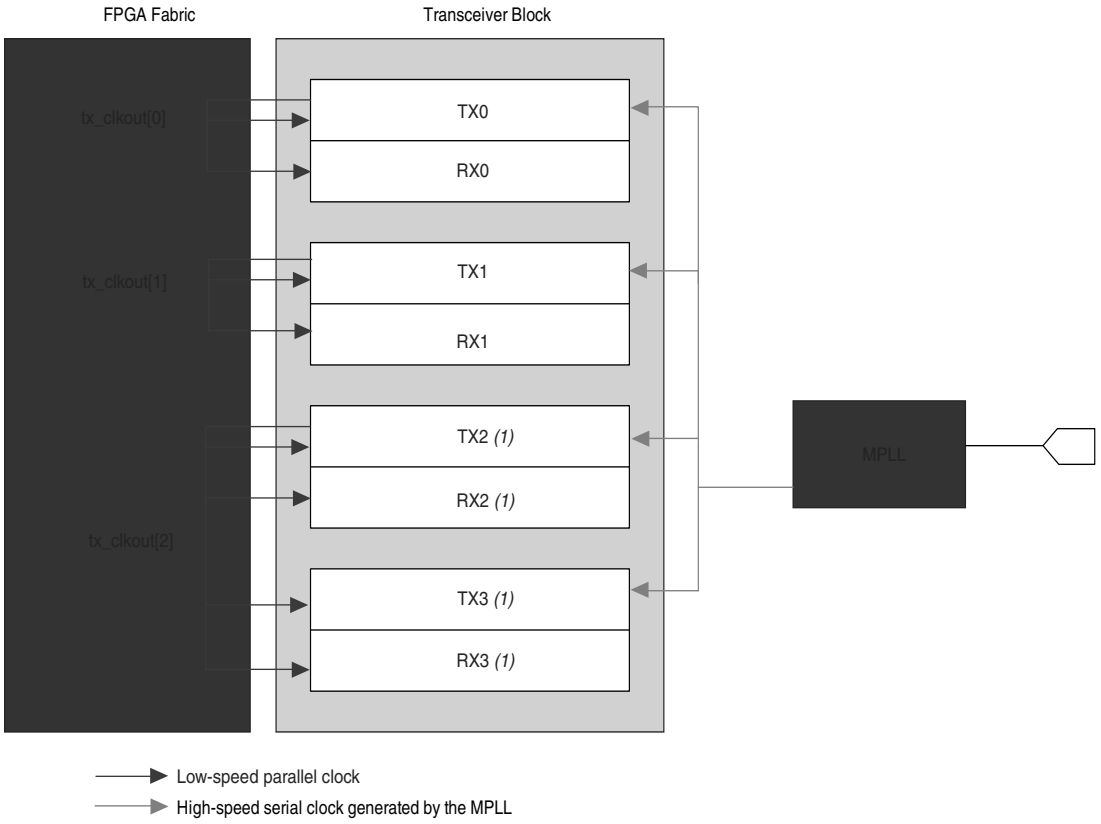
You can use the analog reconfiguration feature to dynamically reconfigure the transceivers channels setting in either the transmitter or the receivers in the PMA blocks. You can update the PMA controls on-the-fly based on the desired input. You can perform both read and write transaction separately for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

Option 2: Use the Respective Channel Transmitter Core Clocks

- Enable this option if you want the individual transmitter channel's `tx_clkout` signal to provide the read clock to its respective Receive Phase Compensation FIFO.
- This option is typically enabled when all the transceiver channels have rate matching enabled with different data rates and are reconfigured to another Basic or Protocol functional mode with rate matching enabled.

Figure 3–14 shows the respective `tx_clkout` of each channel clocking the respective channels of a transceiver block.

Figure 3–14. Option 2 for Receiver Core Clocking (Channel Reconfiguration Mode)



Note to Figure 3–14:

(1) Assuming channel 2 and 3 are running at the same data rate with rate matcher enabled and are reconfigured to another Basic or Protocol functional mode with rate matching enabled.

Table 1–25. PLL Specifications for Cyclone IV Devices (1), (2) (Part 2 of 2)

Symbol	Parameter	Min	Typ	Max	Unit
t_{DLOCK}	Time required to lock dynamically (after switchover, reconfiguring any non-post-scale counters/delays or areset is deasserted)	—	—	1	ms
$t_{OUTJITTER\_PERIOD\_DEDCLK}$ (6)	Dedicated clock output period jitter $F_{OUT} \geq 100$ MHz	—	—	300	ps
	$F_{OUT} < 100$ MHz	—	—	30	mUI
$t_{OUTJITTER\_CCJ\_DEDCLK}$ (6)	Dedicated clock output cycle-to-cycle jitter $F_{OUT} \geq 100$ MHz	—	—	300	ps
	$F_{OUT} < 100$ MHz	—	—	30	mUI
$t_{OUTJITTER\_PERIOD\_IO}$ (6)	Regular I/O period jitter $F_{OUT} \geq 100$ MHz	—	—	650	ps
	$F_{OUT} < 100$ MHz	—	—	75	mUI
$t_{OUTJITTER\_CCJ\_IO}$ (6)	Regular I/O cycle-to-cycle jitter $F_{OUT} \geq 100$ MHz	—	—	650	ps
	$F_{OUT} < 100$ MHz	—	—	75	mUI
t_{PLL_PSERR}	Accuracy of PLL phase shift	—	—	±50	ps
t_{ARESET}	Minimum pulse width on areset signal.	10	—	—	ns
$t_{CONFIGPLL}$	Time required to reconfigure scan chains for PLLs	—	3.5 (7)	—	SCANCLK cycles
$f_{SCANCLK}$	scanclk frequency	—	—	100	MHz
$t_{CASC\_OUTJITTER\_PERIOD\_DEDCLK}$ (8), (9)	Period jitter for dedicated clock output in cascaded PLLs ($F_{OUT} \geq 100$ MHz)	—	—	425	ps
	Period jitter for dedicated clock output in cascaded PLLs ($F_{OUT} < 100$ MHz)	—	—	42.5	mUI

Notes to Table 1–25:

(1) This table is applicable for general purpose PLLs and multipurpose PLLs.

(2) You must connect V_{CCD_PLL} to V_{CCINT} through the decoupling capacitor and ferrite bead.

(3) This parameter is limited in the Quartus II software by the I/O maximum frequency. The maximum I/O frequency is different for each I/O standard.

(4) The V_{CO} frequency reported by the Quartus II software in the PLL Summary section of the compilation report takes into consideration the V_{CO} post-scale counter K value. Therefore, if the counter K has a value of 2, the frequency reported can be lower than the f_{VCO} specification.

(5) A high input jitter directly affects the PLL output jitter. To have low PLL output clock jitter, you must provide a clean clock source that is less than 200 ps.

(6) Peak-to-peak jitter with a probability level of 10^{-12} (14 sigma, 99.99999999974404% confidence level). The output jitter specification applies to the intrinsic jitter of the PLL when an input jitter of 30 ps is applied.

(7) With 100-MHz scanclk frequency.

(8) The cascaded PLLs specification is applicable only with the following conditions:

- Upstream PLL—0.59 MHz ≤ Upstream PLL bandwidth < 1 MHz
- Downstream PLL—Downstream PLL bandwidth > 2 MHz

(9) PLL cascading is not supported for transceiver applications.