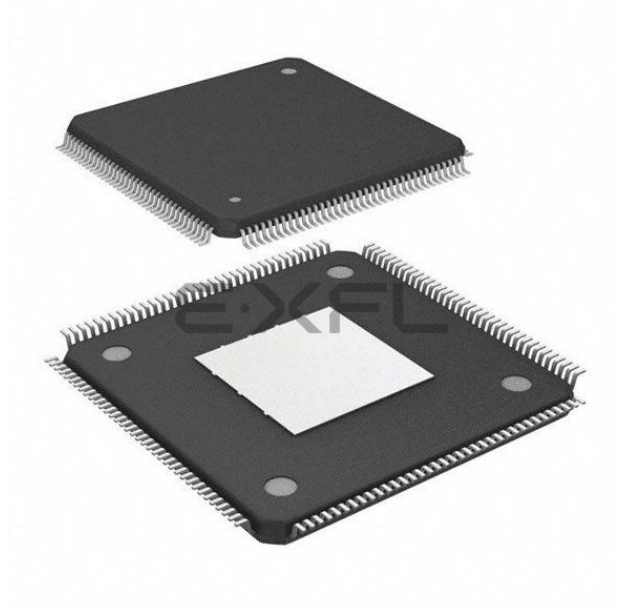


Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

Applications of Embedded - FPGAs

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

Details	
Product Status	Active
Number of LABs/CLBs	1395
Number of Logic Elements/Cells	22320
Total RAM Bits	608256
Number of I/O	79
Number of Gates	-
Voltage - Supply	0.97V ~ 1.03V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	144-LQFP Exposed Pad
Supplier Device Package	144-EQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce22e22c9ln

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.

Table 1–2 lists Cyclone IV GX device resources.

Table 1–2. Resources for the Cyclone IV GX Device Family

Resources	EP4CGX15	EP4CGX22	EP4CGX30 (1)	EP4CGX30 (2)	EP4CGX50 (3)	EP4CGX75 (3)	EP4CGX110 (3)	EP4CGX150 (3)
Logic elements (LEs)	14,400	21,280	29,440	29,440	49,888	73,920	109,424	149,760
Embedded memory (Kbits)	540	756	1,080	1,080	2,502	4,158	5,490	6,480
Embedded 18 × 18 multipliers	0	40	80	80	140	198	280	360
General purpose PLLs	1	2	2	4 (4)	4 (4)	4 (4)	4 (4)	4 (4)
Multipurpose PLLs	2 (5)	2 (5)	2 (5)	2 (5)	4 (5)	4 (5)	4 (5)	4 (5)
Global clock networks	20	20	20	30	30	30	30	30
High-speed transceivers (6)	2	4	4	4	8	8	8	8
Transceiver maximum data rate (Gbps)	2.5	2.5	2.5	3.125	3.125	3.125	3.125	3.125
PCIe (PIPE) hard IP blocks	1	1	1	1	1	1	1	1
User I/O banks	9 (7)	9 (7)	9 (7)	11 (8)	11 (8)	11 (8)	11 (8)	11 (8)
Maximum user I/O (9)	72	150	150	290	310	310	475	475

Notes to Table 1–2:

(1) Applicable for the F169 and F324 packages.

(2) Applicable for the F484 package.

(3) Only two multipurpose PLLs for F484 package.

(4) Two of the general purpose PLLs are able to support transceiver clocking. For more information, refer to the *Clock Networks and PLLs in Cyclone IV Devices* chapter.

(5) You can use the multipurpose PLLs for general purpose clocking when they are not used to clock the transceivers. For more information, refer to the *Clock Networks and PLLs in Cyclone IV Devices* chapter.

(6) If PCIe ×1, you can use the remaining transceivers in a quad for other protocols at the same or different data rates.

(7) Including one configuration I/O bank and two dedicated clock input I/O banks for HSSI reference clock input.

(8) Including one configuration I/O bank and four dedicated clock input I/O banks for HSSI reference clock input.

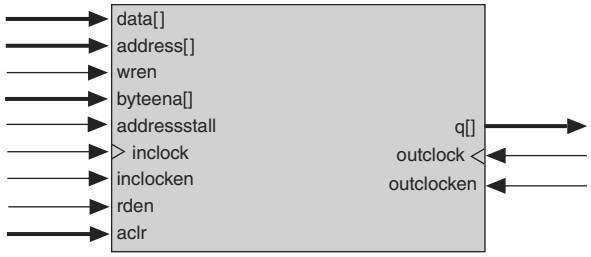
(9) The user I/Os count from pin-out files includes all general purpose I/O, dedicated clock pins, and dual purpose configuration pins. Transceiver pins and dedicated configuration pins are not included in the pin count.

☞ Violating the setup or hold time on the M9K memory block input registers may corrupt memory contents. This applies to both read and write operations.

Single-Port Mode

Single-port mode supports non-simultaneous read and write operations from a single address. Figure 3–6 shows the single-port memory configuration for Cyclone IV devices M9K memory blocks.

Figure 3–6. Single-Port Memory (1), (2)



Notes to Figure 3–6:

(1) You can implement two single-port memory blocks in a single M9K block.
(2) For more information, refer to "Packed Mode Support" on page 3–4.

During a write operation, the behavior of the RAM outputs is configurable. If you activate `rden` during a write operation, the RAM outputs show either the new data being written or the old data at that address. If you perform a write operation with `rden` deactivated, the RAM outputs retain the values they held during the most recent active `rden` signal.

To choose the desired behavior, set the **Read-During-Write** option to either **New Data** or **Old Data** in the RAM MegaWizard Plug-In Manager in the Quartus II software. For more information about read-during-write mode, refer to "Read-During-Write Operations" on page 3–15.

The port width configurations for M9K blocks in single-port mode are as follow:

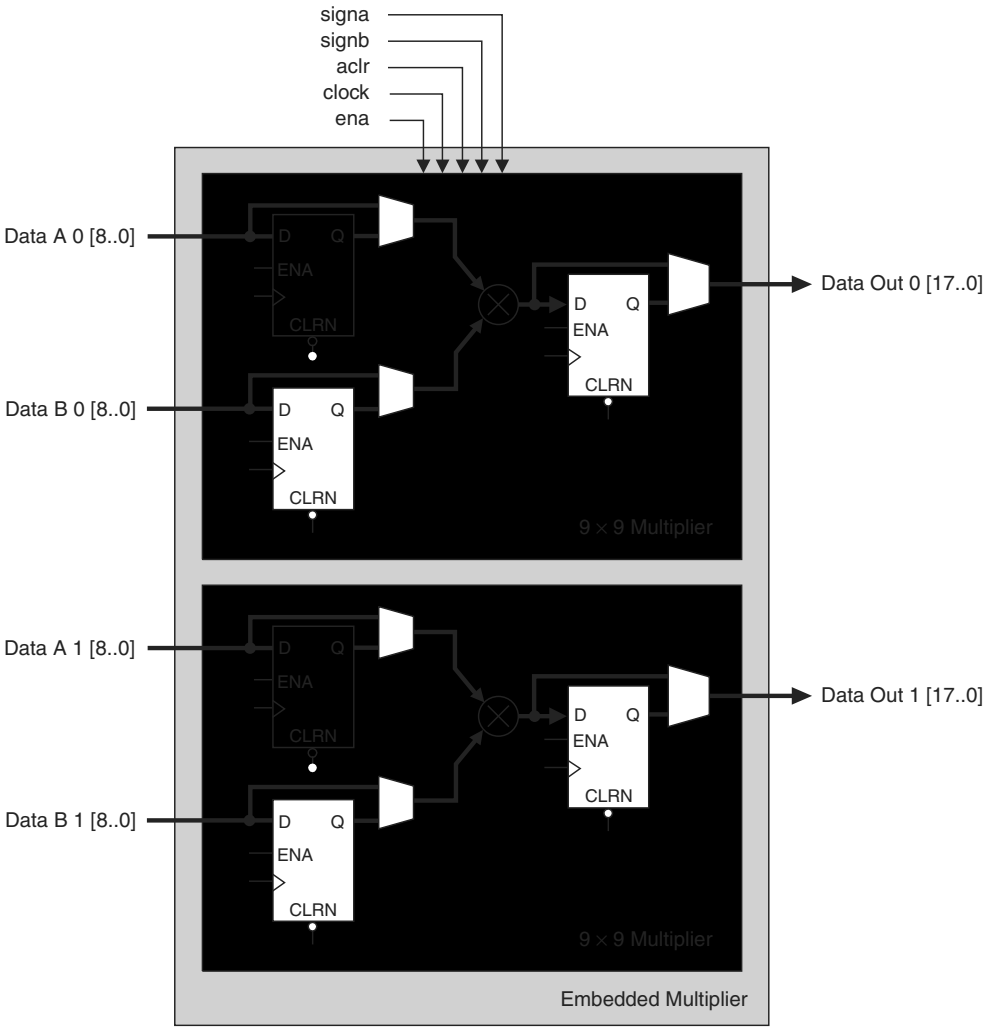
- 8192 × 1
- 4096 × 2
- 2048 × 4
- 1024 × 8
- 1024 × 9
- 512 × 16
- 512 × 18
- 256 × 32
- 256 × 36

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.

PLLs in Cyclone IV Devices

Cyclone IV GX devices offer two variations of PLLs: general purpose PLLs and multipurpose PLLs. Cyclone IV E devices only have the general purpose PLLs.

The general purpose PLLs are used for general-purpose applications in the FPGA fabric and periphery such as external memory interfaces. The multipurpose PLLs are used for clocking the transceiver blocks. When the multipurpose PLLs are not used for transceiver clocking, they can be used for general-purpose clocking.

For more details about the multipurpose PLLs used for transceiver clocking, refer to the *Cyclone IV Transceivers* chapter.

Cyclone IV GX devices contain up to eight general purpose PLLs and multipurpose PLLs while Cyclone IV E devices have up to four general purpose PLLs that provide robust clock management and synthesis for device clock management, external system clock management, and high-speed I/O interfaces.

For more information about the number of general purpose PLLs and multipurpose PLLs in each device density, refer to the *Cyclone IV Device Family Overview* chapter.

The general I/O pins cannot drive the PLL clock input pins.

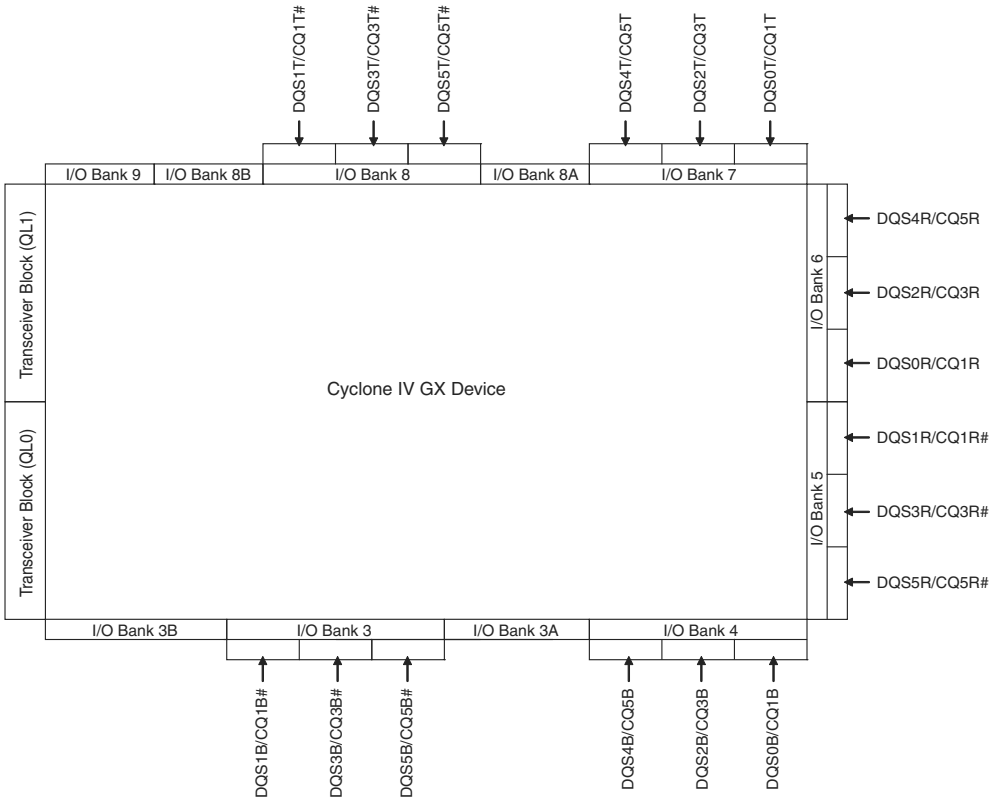
Table 5–5 lists the features available in Cyclone IV GX PLLs.

Table 5–5. Cyclone IV GX PLL Features (Part 1 of 2)

Features	Availability									
	General Purpose PLLs				Multipurpose PLLs					
	PLL_1 (1), (10)	PLL_2 (1), (10)	PLL_3 (2)	PLL_4 (3)	PLL_1 (4)	PLL_2 (4)	PLL_5 (1), (10)	PLL_6 (1), (10)	PLL_7 (1)	PLL_8 (1)
C (output counters)	5									
M, N, C counter sizes	1 to 512 (5)									
Dedicated clock outputs	1 single-ended or 1 differential pair									
Clock input pins	12 single-ended or 6 differential pairs (6) and 4 differential pairs (7)									
Spread-spectrum input clock tracking	✓ (8)									
PLL cascading	Through GCLK									
Source-Synchronous Mode	✓	✓	✓	✓	✓	✓	✓	—	—	✓
No Compensation Mode	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Normal Mode	✓	✓	✓	✓	✓	✓	✓	—	—	✓
Zero Delay Buffer Mode	✓	✓	✓	✓	✓	✓	✓	—	—	✓
Deterministic Latency Compensation Mode	✓	✓	—	—	✓	✓	✓	✓	✓	✓
Phase shift resolution (9)	Down to 96 ps increments									
Programmable duty cycle	✓									
Output counter cascading	✓									

Figure 7–2 shows the location and numbering of the DQS, DQ, or CQ# pins in the Cyclone IV GX I/O banks.

Figure 7–2. DQS, CQ, or CQ# Pins in Cyclone IV GX I/O Banks *(1)*



Note to Figure 7–2:

(1) The DQS, CQ, or CQ# pin locations in this diagram apply to all packages in Cyclone IV GX devices except devices in 169-pin FBGA and 324-pin FBGA.

You can use the Quartus II software with the APU and the appropriate configuration device programming adapter to program serial configuration devices. All serial configuration devices are offered in an 8- or 16-pin small outline integrated circuit (SOIC) package.

In production environments, serial configuration devices are programmed using multiple methods. Altera programming hardware or other third-party programming hardware is used to program blank serial configuration devices before they are mounted onto PCBs. Alternatively, you can use an on-board microprocessor to program the serial configuration device in-system by porting the reference C-based SRunner software driver provided by Altera.

A serial configuration device is programmed in-system by an external microprocessor with the SRunner software driver. The SRunner software driver is a software driver developed for embedded serial configuration device programming, which is easily customized to fit in different embedded systems. The SRunner software driver is able to read a Raw Programming Data (**.rpd**) file and write to serial configuration devices. The serial configuration device programming time, using the SRunner software driver, is comparable to the programming time with the Quartus II software.

For more information about the SRunner software driver, refer to *AN 418: SRunner: An Embedded Solution for Serial Configuration Device Programming* and the source code at the Altera website.

AP Configuration (Supported Flash Memories)

The AP configuration scheme is only supported in Cyclone IV E devices. In the AP configuration scheme, Cyclone IV E devices are configured using commodity 16-bit parallel flash memory. These external non-volatile configuration devices are industry standard microprocessor flash memories. The flash memories provide a fast interface to access configuration data. The speed up in configuration time is mainly due to the 16-bit wide parallel data bus, which is used to retrieve data from the flash memory.

Some of the smaller Cyclone IV E devices or package options do not support the AP configuration scheme. Table 8–9 lists the supported AP configuration scheme for each Cyclone IV E devices.

Table 8–9. Supported AP Configuration Scheme for Cyclone IV E Devices

Device	Package Options								
	E144	M164	M256	U256	F256	F324	U484	F484	F780
EP4CE6	—	—	—	—	—	—	—	—	—
EP4CE10	—	—	—	—	—	—	—	—	—
EP4CE15	—	—	—	—	—	—	—	✓	—
EP4CE22	—	—	—	—	—	—	—	—	—
EP4CE30	—	—	—	—	—	✓	—	✓	✓
EP4CE40	—	—	—	—	—	✓	✓	✓	✓
EP4CE55	—	—	—	—	—	—	✓	✓	✓
EP4CE75	—	—	—	—	—	—	✓	✓	✓
EP4CE115	—	—	—	—	—	—	—	✓	✓

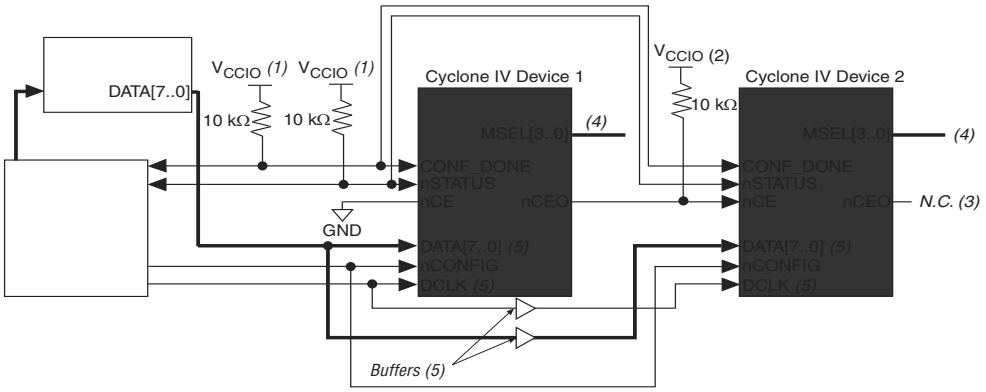
To ensure that DCLK and DATA[0] are not left floating at the end of the configuration, the MAX II device must drive them either high or low, whichever is convenient on your board. The DATA[0] pin is available as a user I/O pin after configuration. When you choose the FPP scheme in the Quartus II software, 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,

You can perform JTAG testing on Cyclone IV devices before, during, and after configuration. Cyclone IV devices support the BYPASS, IDCODE, and SAMPLE instructions during configuration without interrupting configuration. All other JTAG instructions can only be issued by first interrupting configuration and reprogramming I/O pins with the ACTIVE_DISENGAGE and CONFIG_IO instructions.

The CONFIG_IO instruction allows you to configure the I/O buffers through the JTAG port and interrupts configuration when issued after the ACTIVE_DISENGAGE instruction. 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. Prior to issuing the CONFIG_IO instruction, you must issue the ACTIVE_DISENGAGE instruction. This is because in Cyclone IV devices, the CONFIG_IO instruction does not hold nSTATUS low until reconfiguration, so you must disengage the active configuration mode controller when active configuration is interrupted. The ACTIVE_DISENGAGE instruction places the active configuration mode controllers in an idle state prior to JTAG programming. Additionally, the ACTIVE_ENGAGE instruction allows you to re-engage a disengaged active configuration mode controller.

☞ You must follow a specific flow when executing the ACTIVE_DISENGAGE, CONFIG_IO, and ACTIVE_ENGAGE JTAG instructions in Cyclone IV devices.

The chip-wide reset (DEV_CLRn) and chip-wide output enable (DEV_OE) pins in Cyclone IV devices do not affect JTAG boundary-scan or programming operations. Toggling these pins do not affect JTAG operations (other than the usual boundary-scan operation).

When designing a board for JTAG configuration of Cyclone IV devices, consider the dedicated configuration pins. Table 8–15 describes how you must connect these pins during JTAG configuration.

Table 8–15. Dedicated Configuration Pin Connections During JTAG Configuration

Signal	Description
nCE	On all Cyclone IV devices in the chain, nCE must be driven low by connecting it to GND, pulling it low through a resistor, or driving it by some control circuitry. For devices that are also in multi-device AS, AP, PS, or FPP configuration chains, you must connect the nCE pins to GND during JTAG configuration or JTAG configured in the same order as the configuration chain.
nCEO	On all Cyclone IV devices in the chain, nCEO is left floating or connected to the nCE of the next device.
MSEL	These pins must not be left floating. These pins support whichever non-JTAG configuration that you used in production. If you only use JTAG configuration, tie these pins to GND.
nCONFIG	Driven high by connecting to the V_{CCIO} supply of the bank in which the pin resides and pulling up through a resistor or driven high by some control circuitry.
nSTATUS	Pull to the V_{CCIO} supply of the bank in which the pin resides through a 10-kΩ resistor. When configuring multiple devices in the same JTAG chain, each nSTATUS pin must be pulled up to the V_{CCIO} individually.
CONF_DONE	Pull to the V_{CCIO} supply of the bank in which the pin resides through a 10-kΩ resistor. When configuring multiple devices in the same JTAG chain, each CONF_DONE pin must be pulled up to V_{CCIO} supply of the bank in which the pin resides individually. CONF_DONE going high at the end of JTAG configuration indicates successful configuration.
DCLK	Must not be left floating. Drive low or high, whichever is more convenient on your board.

Table 8–20. Dedicated Configuration Pins on the Cyclone IV Device (Part 4 of 4)

Pin Name	User Mode	Configuration Scheme	Pin Type	Description
DATA[7..2]	I/O	FPP, AP (2)	Inputs (FPP). Bidirectional (AP) (2)	In an AS or PS configuration scheme, DATA[7..2] function as user I/O pins during configuration, which means they are tri-stated. After FPP configuration, DATA[7..2] are available as user I/O pins and the state of these pin depends on the **Dual-Purpose Pin** settings. In an AP configuration scheme, for Cyclone IV E devices only, the byte-wide or word-wide configuration data is presented to the target Cyclone IV E device on DATA[7..0] or DATA[15..0], respectively. After AP configuration, DATA[7..2] are dedicated bidirectional pins with optional user control. (2)
DATA[15..8]	I/O	AP (2)	Bidirectional	Data inputs. Word-wide configuration data is presented to the target Cyclone IV E device on DATA[15..0]. In a PS, FPP, or AS configuration scheme, DATA[15:8] function as user I/O pins during configuration, which means they are tri stated. After AP configuration, DATA[15:8] are dedicated bidirectional pins with optional user control.
PADD[23..0]	I/O	AP (2)	Output	In AP mode, it is a 24-bit address bus from the Cyclone IV E device to the parallel flash. Connects to the A[24:1] bus on the Micron P30 or P33 flash.
nRESET	I/O	AP (2)	Output	Active-low reset output. Driving the nRESET pin low resets the parallel flash. Connects to the RST# pin on the Micron P30 or P33 flash.
nAVD	I/O	AP (2)	Output	Active-low address valid output. Driving the nAVD pin low during read or write operation indicates to the parallel flash that a valid address is present on the PADD[23..0] address bus. Connects to the ADV# pin on the Micron P30 or P33 flash.
nOE	I/O	AP (2)	Output	Active-low output enable to the parallel flash. During the read operation, driving the nOE pin low enables the parallel flash outputs (DATA[15..0]). Connects to the OE# pin on the Micron P30 or P33 flash.
nWE	I/O	AP (2)	Output	Active-low write enable to the parallel flash. During the write operation, driving the nWE pin low indicates to the parallel flash that data on the DATA[15..0] bus is valid. Connects to the WE# pin on the Micron P30 or P33 flash.

Note to Table 8–20:

(1) If you are accessing the EPCS device with the ALTASMI_PARALLEL megafunction or your own user logic in user mode, in the **Device and Pin Options** window of the Quartus II software, in the **Dual-Purpose Pins** category, select **Use as regular I/O** for this pin.

(2) The AP configuration scheme is for Cyclone IV E devices only.

Document Revision History

Table 9–8 lists the revision history for this chapter.

Table 9–8. Document Revision History

Date	Version	Changes
May 2013	1.3	Updated “CRC_ERROR Pin Type” in Table 9–2.
October 2012	1.2	Updated Table 9–2.
February 2010	1.1	Updated for the Quartus II software version 9.1 SP1 release: ■ Updated “Configuration Error Detection” section. ■ Updated Table 9–6. ■ Added Cyclone IV E devices in Table 9–6.
November 2009	1.0	Initial release.

In some applications, it is necessary for a device to wake up very quickly to begin operation. Cyclone IV devices offer the Fast-On feature to support fast wake-up time applications. The MSEL pin settings determine the POR time (t_{POR}) of the device.

For more information about the MSEL pin settings, refer to the *Configuration and Remote System Upgrades in Cyclone IV Devices* chapter.

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

Document Revision History

Table 11–3 lists the revision history for this chapter.

Table 11–3. Document Revision History

Date	Version	Changes
May 2013	1.3	Updated Note (4) in Table 11–1.
July 2010	1.2	■ Updated for the Quartus II software version 10.0 release. ■ Updated "I/O Pins Remain Tri-stated During Power-Up" section. ■ Updated Table 11–1.
February 2010	1.1	Updated Table 11–1 and Table 11–2 for the Quartus II software version 9.1 SP1 release.
November 2009	1.0	Initial release.

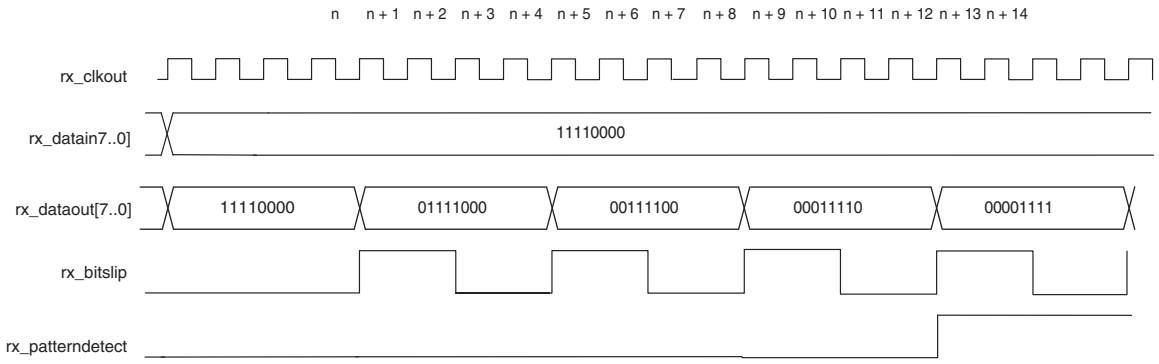
Bit-Slip Mode

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

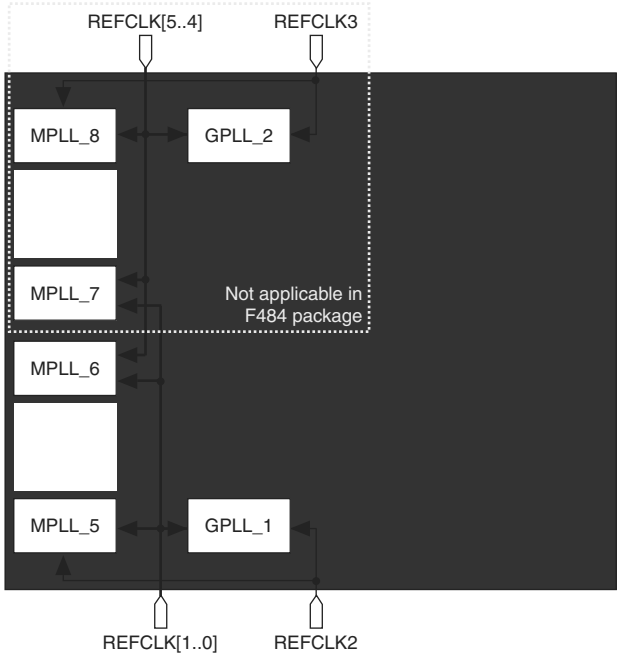


Automatic Synchronization State Machine Mode

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.

Figure 1–26. PLL Input Reference Clocks in Transceiver Operation for F484 and Larger Packages (1), (2), (3)



Notes to Figure 1–26:

(1) The REFCLK2 and REFCLK3 pins are dual-purpose CLKIO, REFCLK, or DIFFCLK pins that reside in banks 3A and 8A respectively.

(2) The REFCLK[1..0] and REFCLK[5..4] pins are dual-purpose differential REFCLK or DIFFCLK pins that reside in banks 3B and 8B respectively. These clock input pins do not have access to the clock control blocks and GCLK networks. For more details, refer to the *Clock Networks and PLLs in Cyclone IV Devices* chapter.

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

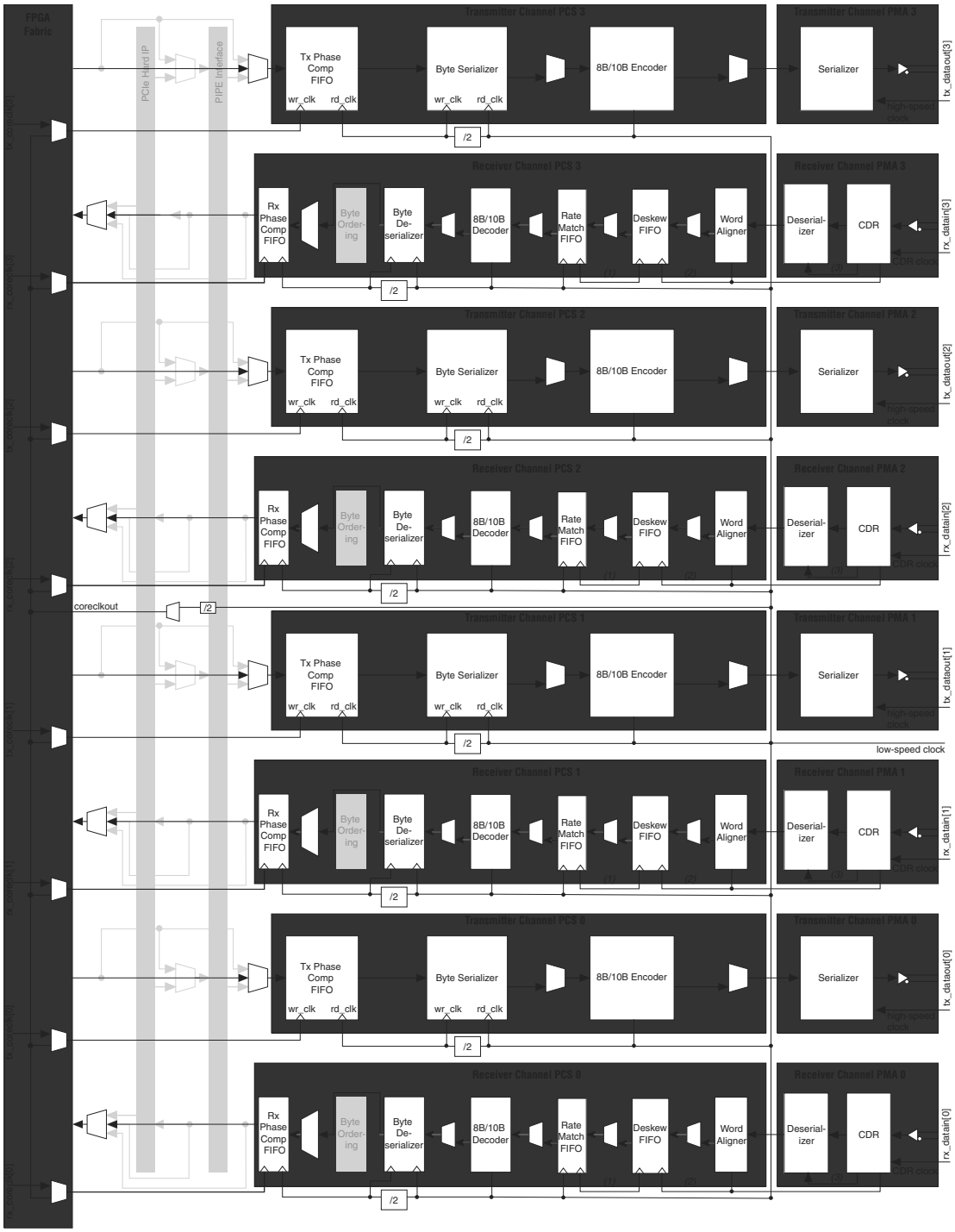
The input reference clocks reside in banks 3A, 3B, 8A, and 8B have dedicated $V_{CC\_CLKIN3A}$, $V_{CC\_CLKIN3B}$, $V_{CC\_CLKIN8A}$, and $V_{CC\_CLKIN8B}$ power supplies separately in their respective I/O banks to avoid the different power level requirements in the same bank for general purpose I/Os (GPIOs). Table 1–6 lists the supported I/O standard for the REFCLK pins.

Table 1–6. REFCLK I/O Standard Support

I/O Standard	HSSI Protocol	Coupling	Termination	VCC_CLKIN Level		I/O Pin Type		
				Input	Output	Column I/O	Row I/O	Supported Banks
LVDS	ALL	Differential AC (Needs off-chip resistor to restore V_{CM})	Off-chip	2.5 V	Not Supported	Yes	No	3A, 3B, 8A, 8B
LVPECL	ALL		Off-chip	2.5 V	Not Supported	Yes	No	3A, 3B, 8A, 8B
1.2 V, 1.5 V, 3.3 V PCML	ALL		Off-chip	2.5 V	Not Supported	Yes	No	3A, 3B, 8A, 8B
	ALL		Off-chip	2.5 V	Not Supported	Yes	No	3A, 3B, 8A, 8B
	ALL		Off-chip	2.5 V	Not Supported	Yes	No	3A, 3B, 8A, 8B
HCSL	PCIe	Differential DC	Off-chip	2.5 V	Not Supported	Yes	No	3A, 3B, 8A, 8B

Figure 1–63 shows the transceiver channel datapath and clocking when configured in XAUI mode.

Figure 1–63. Transceiver Channel Datapath and Clocking when Configured in XAUI Mode



Notes to Figure 1–63:

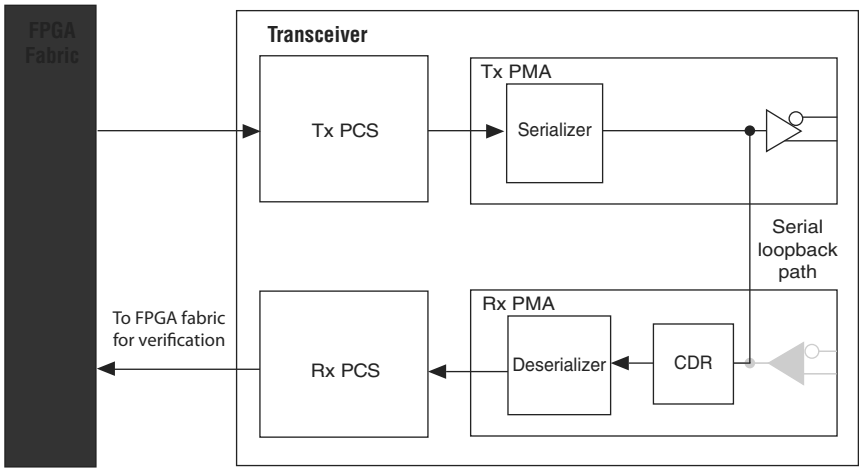
(1) Channel 1 low-speed recovered clock.

(2) Low-speed recovered clock.

(3) High-speed recovered clock.

Serial loopback mode can only be dynamically enabled or disabled during user mode by performing a dynamic channel reconfiguration.

Figure 1–71. Serial Loopback Path (1)



Note to Figure 1–71:

(1) Grayed-Out Blocks are Not Active in this mode.

Reverse Serial Loopback

The reverse serial loopback mode is available for all functional modes except for XAUI mode. The two reverse serial loopback options from the receiver to the transmitter are:

- Pre-CDR mode where data received through the RX input buffer is looped back to the TX output buffer using the **Reverse serial loopback (pre-CDR)** option
- Post-CDR mode where retimed data through the receiver CDR from the RX input buffer is looped back to the TX output buffer using the **Reverse serial loopback** option

The received data is also available to the FPGA logic. In the transmitter channel, only the transmitter buffer is active.

The transmitter pre-emphasis feature is not available in reverse serial loopback (pre-CDR) mode.

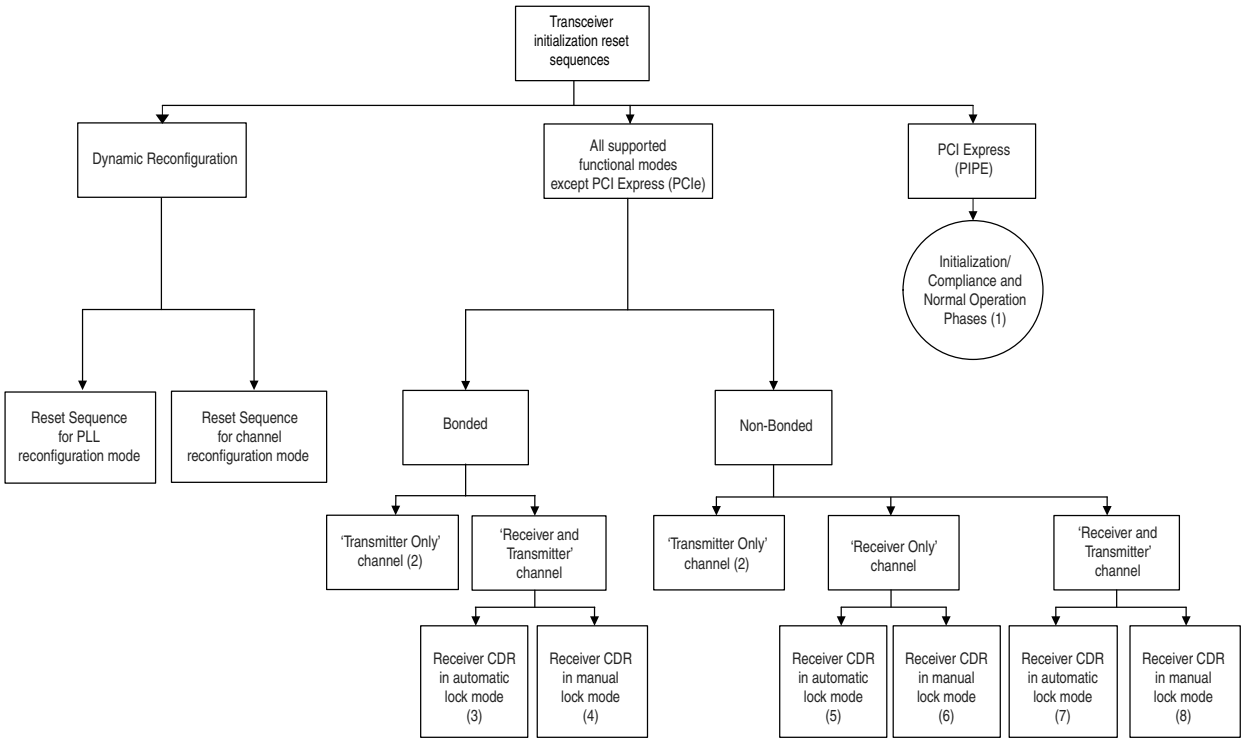
Reverse serial loopback modes can only be dynamically enabled or disabled during user mode by performing a dynamic channel reconfiguration.

The busy signal remains low for the first reconfig_clk clock cycle. It then gets asserted from the second reconfig_clk clock cycle. Subsequent deassertion of the busy signal indicates the completion of the offset cancellation process. This busy signal is required in transceiver reset sequences except for transmitter only channel configurations. Refer to the reset sequences shown in Figure 2–2 and the associated references listed in the notes for the figure.

Altera strongly recommends adhering to these reset sequences for proper operation of the Cyclone IV GX transceiver.

Figure 2–2 shows the transceiver reset sequences for Cyclone IV GX devices.

Figure 2–2. Transceiver Reset Sequences Chart



Notes to Figure 2–2:

(1) Refer to the Timing Diagram in Figure 2–10.
(2) Refer to the Timing Diagram in Figure 2–3.
(3) Refer to the Timing Diagram in Figure 2–4.
(4) Refer to the Timing Diagram in Figure 2–5.
(5) Refer to the Timing Diagram in Figure 2–6.
(6) Refer to the Timing Diagram in Figure 2–7.
(7) Refer to the Timing Diagram in Figure 2–8.
(8) Refer to the Timing Diagram in Figure 2–9.

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)