

Welcome to [E-XFL.COM](https://www.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

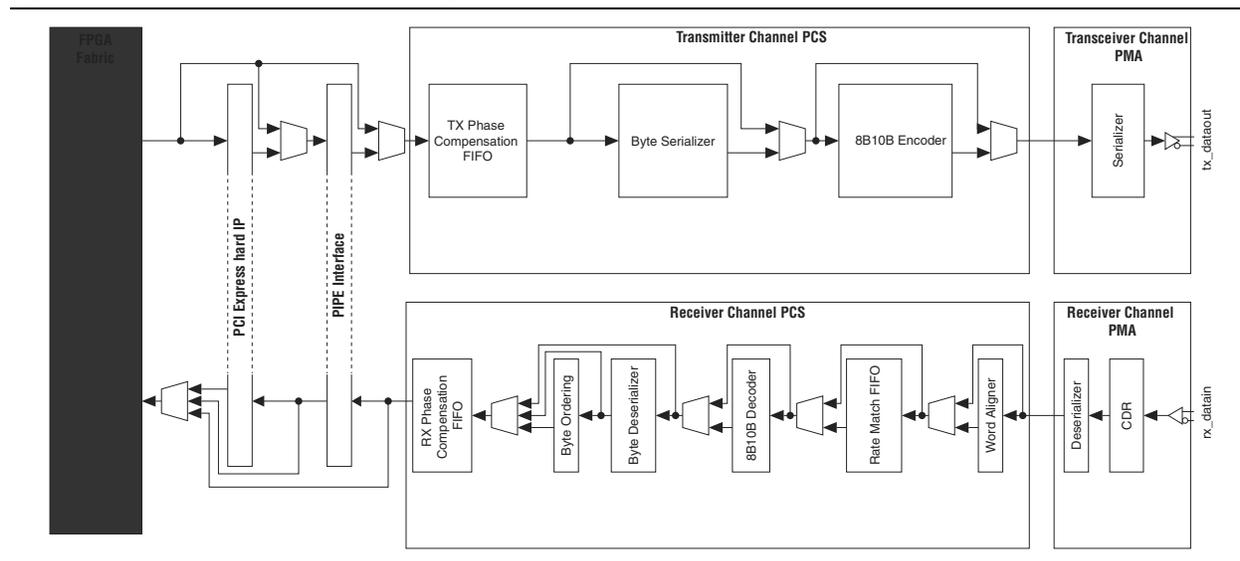
The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details

Product Status	Active
Number of LABs/CLBs	1803
Number of Logic Elements/Cells	28848
Total RAM Bits	608256
Number of I/O	328
Number of Gates	-
Voltage - Supply	1.15V ~ 1.25V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	484-BGA
Supplier Device Package	484-FBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce30f23c7

Figure 1-1 shows the structure of the Cyclone IV GX transceiver.

Figure 1-1. Transceiver Channel for the Cyclone IV GX Device



For more information, refer to the *Cyclone IV Transceivers Architecture* chapter.

Hard IP for PCI Express (Cyclone IV GX Devices Only)

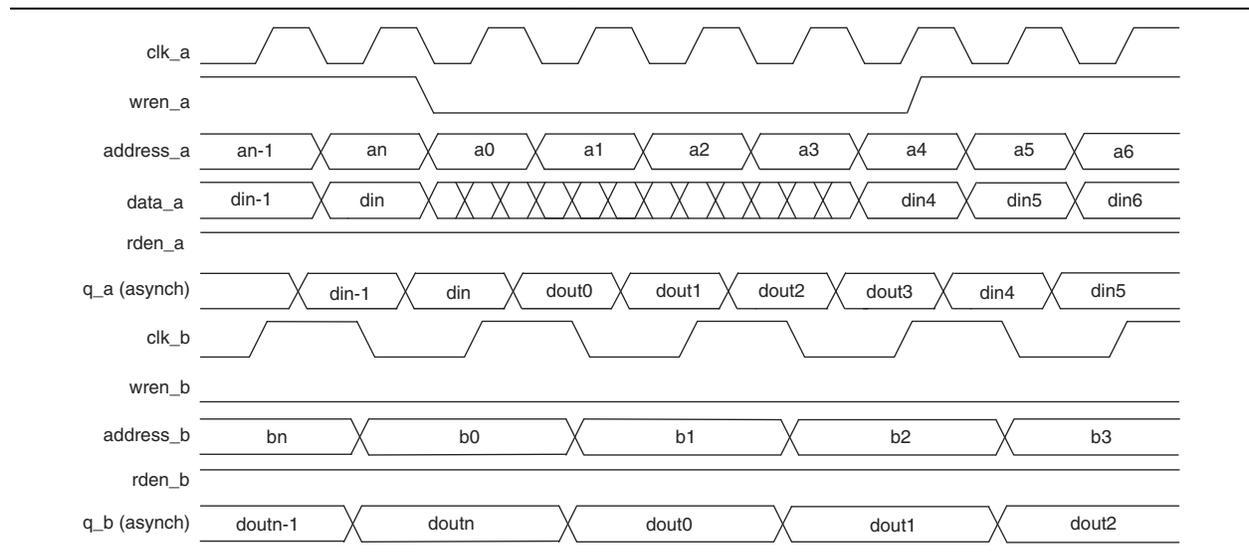
Cyclone IV GX devices incorporate a single hard IP block for $\times 1$, $\times 2$, or $\times 4$ PCIe (PIPE) in each device. This hard IP block is a complete PCIe (PIPE) protocol solution that implements the PHY-MAC layer, Data Link Layer, and Transaction Layer functionality. The hard IP for the PCIe (PIPE) block supports root-port and end-point configurations. This pre-verified hard IP block reduces risk, design time, timing closure, and verification. You can configure the block with the Quartus II software's PCI Express Compiler, which guides you through the process step by step.

For more information, refer to the *PCI Express Compiler User Guide*.

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

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

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



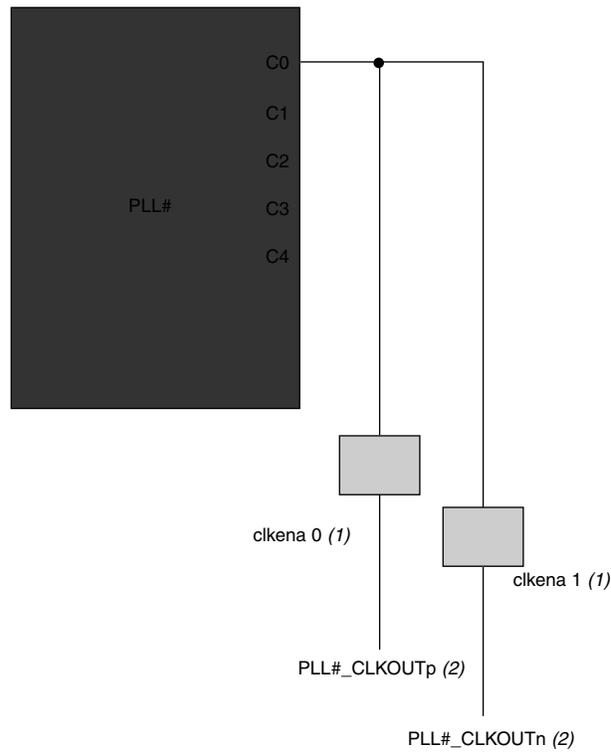
Shift Register Mode

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

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

Figure 5-11 shows the external clock outputs for PLLs.

Figure 5-11. External Clock Outputs for PLLs



Notes to Figure 5-11:

- (1) These external clock enable signals are available only when using the ALTCLKCTRL megafunction.
- (2) PLL#_CLKOUTp and PLL#_CLKOUTn pins are dual-purpose I/O pins that you can use as one single-ended clock output or one differential clock output. When using both pins as single-ended I/Os, one of them can be the clock output while the other pin is configured as a regular user I/O.

Each pin of a differential output pair is 180° out of phase. The Quartus II software places the NOT gate in your design into the I/O element to implement 180° phase with respect to the other pin in the pair. The clock output pin pairs support the same I/O standards as standard output pins.

 To determine which I/O standards are supported by the PLL clock input and output pins, refer to the *Cyclone IV Device I/O Features* chapter.

Cyclone IV PLLs can drive out to any regular I/O pin through the GCLK. You can also use the external clock output pins as GPIO pins if external PLL clocking is not required.

LVPECL I/O Support in Cyclone IV Devices

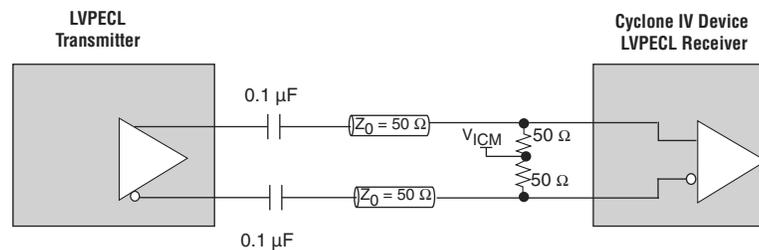
The LVPECL I/O standard is a differential interface standard that requires a 2.5-V V_{CCIO} . This standard is used in applications involving video graphics, telecommunications, data communications, and clock distribution. Cyclone IV devices support the LVPECL input standard at the dedicated clock input 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.

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

Document Revision History

Table 7-3 lists the revision history for this chapter.

Table 7-3. Document Revision History

Date	Version	Changes
March 2016	2.6	<ul style="list-style-type: none"> ■ Updated Table 7-1 to remove support for the N148 package. ■ Updated note (1) in Figure 7-2 to remove support for the N148 package. ■ Updated Figure 7-4 to remove support for the N148 package.
May 2013	2.5	Updated Table 7-2 to add new device options and packages.
February 2013	2.4	Updated Table 7-2 to add new device options and packages.
October 2012	2.3	Updated Table 7-1 and Table 7-2.
December 2010	2.2	<ul style="list-style-type: none"> ■ Updated for the Quartus II software version 10.1 release. ■ Added Cyclone IV E new device package information. ■ Updated Table 7-2. ■ Minor text edits.
November 2010	2.1	Updated “Data and Data Clock/Strobe Pins” section.
February 2010	2.0	<ul style="list-style-type: none"> ■ Added Cyclone IV E devices information for the Quartus II software version 9.1 SP1 release. ■ Updated Table 7-1. ■ Added Table 7-2. ■ Added Figure 7-5 and Figure 7-6.
November 2009	1.0	Initial release.

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

devices. The internal oscillator is designed to ensure that its maximum frequency is guaranteed to meet EPCS device specifications. Cyclone IV devices offer the option to select `CLKUSR` as the external clock source for `DCLK`. You can change the clock source option in the Quartus II software in the **Configuration** tab of the **Device and Pin Options** dialog box.

 EPCS1 does not support Cyclone IV devices because of its insufficient memory capacity.

Table 8-6. AS DCLK Output Frequency

Oscillator	Minimum	Typical	Maximum	Unit
40 MHz	20	30	40	MHz

In configuration mode, the Cyclone IV device enables the serial configuration device by driving the `nCS0` output pin low, which connects to the `nCS` pin of the configuration device. The Cyclone IV device uses the `DCLK` and `DATA [1]` pins to send operation commands and read address signals to the serial configuration device. The configuration device provides data on its `DATA` pin, which connects to the `DATA [0]` input of the Cyclone IV device.

All AS configuration pins (`DATA [0]`, `DCLK`, `nCS0`, and `DATA [1]`) have weak internal pull-up resistors that are always active. After configuration, these pins are set as input tri-stated and are driven high by the weak internal pull-up resistors.

The timing parameters for AS mode are not listed here because the t_{CF2CD} , t_{CF2ST0} , t_{CFG} , t_{STATUS} , t_{CF2ST1} , and t_{CD2UM} timing parameters are identical to the timing parameters for PS mode shown in Table 8-12 on page 8-36.

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

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

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

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

If the configuration bitstream size exceeds the capacity of a serial configuration device, you must select a larger configuration device, enable the compression feature, or both. When configuring multiple devices, the size of the bitstream is the sum of the individual device's configuration bitstream.

Configuring Multiple Cyclone IV Devices with the Same Design

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

- Multiple `.sof`
- Single `.sof`

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

Multiple SRAM Object Files

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

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

Table 8-13. FPP Timing Parameters for Cyclone IV Devices (Part 2 of 2)

Symbol	Parameter	Minimum		Maximum		Unit
		Cyclone IV ⁽¹⁾	Cyclone IV E ⁽²⁾	Cyclone IV ⁽¹⁾	Cyclone IV E ⁽²⁾	
t_{ST2CK}	$\overline{nSTATUS}$ high to first rising edge of DCLK	2		—		μs
t_{DH}	Data hold time after rising edge on DCLK	0		—		ns
t_{CD2UM}	CONF_DONE high to user mode ⁽⁵⁾	300		650		μs
t_{CD2CU}	CONF_DONE high to CLKUSR enabled	4 × maximum DCLK period		—		—
t_{CD2UMC}	CONF_DONE high to user mode with CLKUSR option on	$t_{CD2CU} + (3,192 \times \text{CLKUSR period})$		—		—
t_{DSU}	Data setup time before rising edge on DCLK	5	8	—	—	ns
t_{CH}	DCLK high time	3.2	6.4	—	—	ns
t_{CL}	DCLK low time	3.2	6.4	—	—	ns
t_{CLK}	DCLK period	7.5	15	—	—	ns
f_{MAX}	DCLK frequency ⁽⁶⁾	—	—	133	66	MHz

Notes to Table 8-13:

- (1) Applicable for Cyclone IV GX and Cyclone IV E with 1.2-V core voltage.
- (2) Applicable for Cyclone IV E with 1.0-V core voltage.
- (3) This value is applicable if you do not delay configuration by extending the $\overline{nCONFIG}$ or $\overline{nSTATUS}$ low pulse width.
- (4) This value is applicable if you do not delay configuration by externally holding the $\overline{nSTATUS}$ low.
- (5) The minimum and maximum numbers apply only if you choose the internal oscillator as the clock source for starting the device.
- (6) Cyclone IV E devices with 1.0-V core voltage have slower F_{MAX} when compared with Cyclone IV GX devices with 1.2-V core voltage.

JTAG Configuration

JTAG has developed a specification for boundary-scan testing (BST). The BST architecture offers the capability to efficiently test components on PCBs with tight lead spacing. The BST architecture can test pin connections without using physical test probes and capture functional data while a device is normally operating. You can also use the JTAG circuitry to shift configuration data into the device. The Quartus II software automatically generates .sof for JTAG configuration with a download cable in the Quartus II software Programmer.



For more information about the JTAG boundary-scan testing, refer to the *JTAG Boundary-Scan Testing for Cyclone IV Devices* chapter.

Table 8-20. Dedicated Configuration Pins on the Cyclone IV Device (Part 3 of 4)

Pin Name	User Mode	Configuration Scheme	Pin Type	Description
DCLK ⁽¹⁾	N/A	PS, FPP, AS, AP ⁽²⁾	Input (PS, FPP) ⁽²⁾	In PS and FPP configuration, DCLK is the clock input used to clock data from an external source into the target Cyclone IV device. Data is latched into the device on the rising edge of DCLK.
	I/O		Output (AS, AP)	In AS mode, DCLK is an output from the Cyclone IV device that provides timing for the configuration interface. It has an internal pull-up resistor (typically 25 kΩ) that is always active. In AP mode, DCLK is an output from the Cyclone IV E device that provides timing for the configuration interface. ⁽²⁾ In AS or AP configuration schemes, this pin is driven into an inactive state after configuration completes. Alternatively, in active schemes, you can use this pin as a user I/O during user mode. In PS or FPP schemes that use a control host, you must drive DCLK either high or low, whichever is more convenient. In passive schemes, you cannot use DCLK as a user I/O in user mode. Toggling this pin after configuration does not affect the configured device.
DATA[0] ⁽¹⁾	I/O	PS, FPP, AS, AP ⁽²⁾	Input (PS, FPP, AS). Bidirectional (AP) ⁽²⁾	Data input. In serial configuration modes, bit-wide configuration data is presented to the target Cyclone IV device on the DATA[0] pin. In AS mode, DATA[0] has an internal pull-up resistor that is always active. After AS configuration, DATA[0] is a dedicated input pin with optional user control. After PS or FPP configuration, DATA[0] is available as a user I/O pin. The state of this pin depends on the Dual-Purpose Pin settings. After AP configuration, DATA[0] is a dedicated bidirectional pin with optional user control. ⁽²⁾
DATA[1]/ASDO ⁽¹⁾	I/O	FPP, AS, AP ⁽²⁾	Input (FPP). Output (AS). Bidirectional (AP) ⁽²⁾	The DATA[1] pin functions as the ASDO pin in AS mode. Data input in non-AS mode. Control signal from the Cyclone IV device to the serial configuration device in AS mode used to read out configuration data. In AS mode, DATA[1] has an internal pull-up resistor that is always active. After AS configuration, DATA[1] is a dedicated output pin with optional user control. In a PS configuration scheme, DATA[1] functions as a user I/O pin during configuration, which means it is tri-stated. After FPP configuration, DATA[1] is available as a user I/O pin and the state of this 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[1] is a dedicated bidirectional pin with optional user control. ⁽²⁾

In user mode, Cyclone IV devices support the `CHANGE_EDREG` JTAG instruction, that allows you to write to the 32-bit storage register. You can use Jam™ STAPL files (`.jam`) to automate the testing and verification process. You can only execute this instruction when the device is in user mode, and it is a powerful design feature that enables you to dynamically verify the CRC functionality in-system without having to reconfigure the device. You can then use the CRC circuit to check for real errors induced by an SEU.

Table 9–1 describes the `CHANGE_EDREG` JTAG instructions.

Table 9–1. CHANGE_EDREG JTAG Instruction

JTAG Instruction	Instruction Code	Description
<code>CHANGE_EDREG</code>	00 0001 0101	This instruction connects the 32-bit CRC storage register between TDI and TDO. Any precomputed CRC is loaded into the CRC storage register to test the operation of the error detection CRC circuitry at the <code>CRC_ERROR</code> pin.

 After the test completes, Altera recommends that you power cycle the device.

Automated SEU Detection

Cyclone IV devices offer on-chip circuitry for automated checking of SEU detection. Applications that require the device to operate error-free at high elevations or in close proximity to earth’s north or south pole require periodic checks to ensure continued data integrity. The error detection cyclic redundancy code feature controlled by the **Device and Pin Options** dialog box in the Quartus II software uses a 32-bit CRC circuit to ensure data reliability and is one of the best options for mitigating SEU.

You can implement the error detection CRC feature with existing circuitry in Cyclone IV devices, eliminating the need for external logic. The CRC is computed by the device during configuration and checked against an automatically computed CRC during normal operation. The `CRC_ERROR` pin reports a soft error when configuration CRAM data is corrupted. You must decide whether to reconfigure the FPGA by strobing the `nCONFIG` pin low or ignore the error.

CRC_ERROR Pin

A specific `CRC_ERROR` error detection pin is required to monitor the results of the error detection circuitry during user mode. Table 9–2 describes the `CRC_ERROR` pin.

Table 9–2. Cyclone IV Device CRC_ERROR Pin Description

CRC_ERROR Pin Type	Description
I/O, Output (open-drain)	Active high signal indicates that the error detection circuit has detected errors in the configuration SRAM bits. This pin is optional and is used when the CRC error detection circuit is enabled in the Quartus II software from the Error Detection CRC tab of the Device and Pin Options dialog box. When using this pin, connect it to an external 10-kΩ pull-up resistor to an acceptable voltage that satisfies the input voltage of the receiving device.

 The `CRC_ERROR` pin information for Cyclone IV devices is reported in the Cyclone IV Devices Pin-Outs on the Altera® website.

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: <ul style="list-style-type: none"> ■ Updated “Configuration Error Detection” section. ■ Updated Table 9-6. ■ Added Cyclone IV E devices in Table 9-6.
November 2009	1.0	Initial release.

Actual lock time depends on the transition density of the incoming data and the ppm difference between the receiver input reference clock and the upstream transmitter reference clock.

Transition from the LTD state to the LTR state occurs when either of the following conditions is met:

- Signal detection circuitry indicates the absence of valid signal levels at the receiver input buffer. This condition is valid for PCI Express (PIPE) mode only. CDR transitions are not dependent on signal detection circuitry in other modes.
- The recovered clock is not within the configured ppm frequency threshold setting with respect to CDR clocks from multipurpose PLLs.

In automatic lock mode, the switch from LTR to LTD states is indicated by the assertion of the `rx_freqlocked` signal and the switch from LTD to LTR states indicated by the de-assertion of the `rx_freqlocked` signal.

Manual Lock Mode

State transitions are controlled manually by using `rx_locktorefclk` and `rx_locktodata` ports. The LTR/LTD controller sets the CDR state depending on the logic level on the `rx_locktorefclk` and `rx_locktodata` ports. This mode provides the flexibility to control the CDR for a reduced lock time compared to the automatic lock mode. In automatic lock mode, the LTR/LTD controller relies on the ppm detector and the phase relationship detector to set the CDR in LTR or LTD mode. The ppm detector and phase relationship detector reaction times can be too long for some applications that require faster CDR lock time.

In manual lock mode, the `rx_freqlocked` signal is asserted when the CDR is in LTD state and de-asserted when CDR is in LTR state. For descriptions of `rx_locktorefclk` and `rx_locktodata` port controls, refer to Table 1-27 on page 1-87.

-  If you do not enable the optional `rx_locktorefclk` and `rx_locktodata` ports, the Quartus II software automatically configures the LTR/LTD controller in automatic lock mode.
-  The recommended transceiver reset sequence varies depending on the CDR lock mode. For more information about the reset sequence recommendations, refer to the *Reset Control and Power Down for Cyclone IV GX Devices* chapter.

Deserializer

The deserializer converts received serial data from the receiver input buffer to parallel 8- or 10-bit data. Serial data is assumed to be received from the LSB to the MSB. The deserializer operates with the high-speed recovered clock from the CDR with the frequency at half of the serial data rate.

 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)	<ul style="list-style-type: none"> ■ Low-speed clock in each channel is sourced independently ■ Phase compensation FIFO in each channel has its own pointers and control logic 	<ul style="list-style-type: none"> ■ Transmitter Only ■ Receiver Only ■ Transmitter and Receiver
Bonded (×2 and ×4)	<ul style="list-style-type: none"> ■ 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 	<ul style="list-style-type: none"> ■ Transmitter Only ■ Transmitter and Receiver

Non-Bonded Channel Configuration

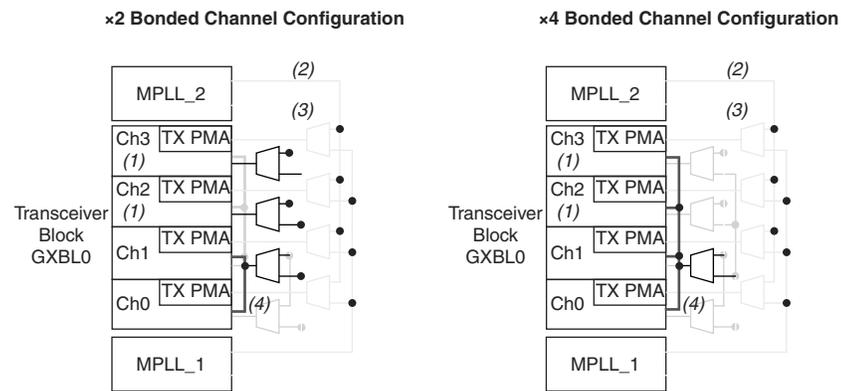
In non-bonded channel configuration, the high- and low-speed clocks for each channel are sourced independently. The phase compensation FIFOs in each channel has its own pointers and control logic. When implementing multi-channel serial interface in non-bonded channel configuration, the clock skew and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

Figure 1-36 and Figure 1-37 show the independent high-speed clock and bonded low-speed clock distributions for transceivers in F324 and smaller packages, and in F484 and larger packages in bonded ($\times 2$ and $\times 4$) channel configuration.

Figure 1-36. Clock Distribution in Bonded ($\times 2$ and $\times 4$) Channel Configuration for Transceivers in F324 and Smaller Packages.

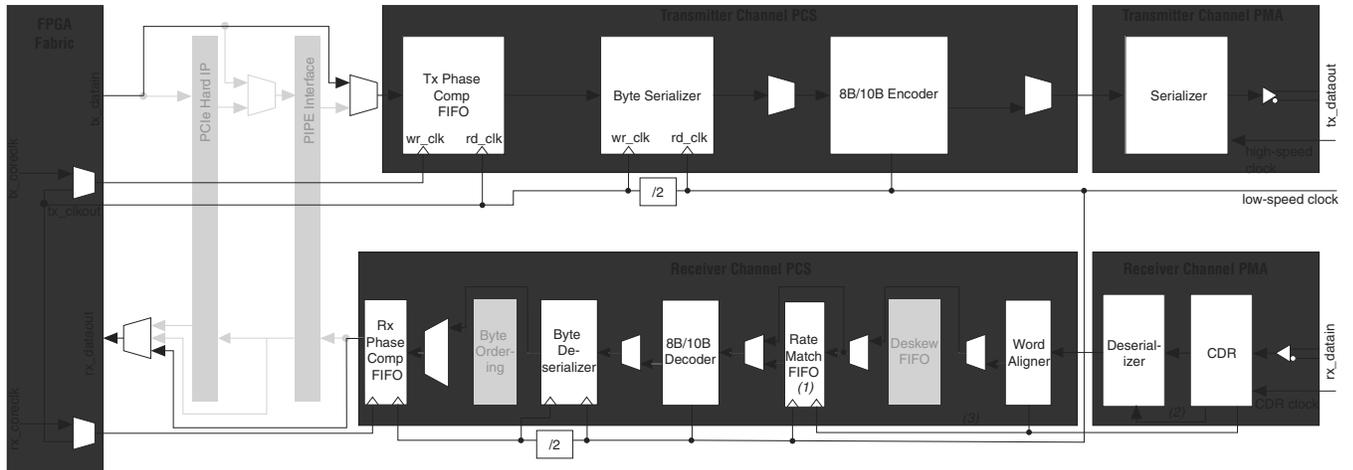


Notes to Figure 1-36:

- (1) Transceiver channels 2 and 3 are not available for devices in F169 and smaller packages.
- (2) High-speed clock.
- (3) Low-speed clock.
- (4) Bonded common low-speed clock path.

Figure 1-60 shows the transceiver channel datapath and clocking when configured in Serial RapidIO mode.

Figure 1-60. Transceiver Channel Datapath and Clocking when Configured in Serial RapidIO Mode



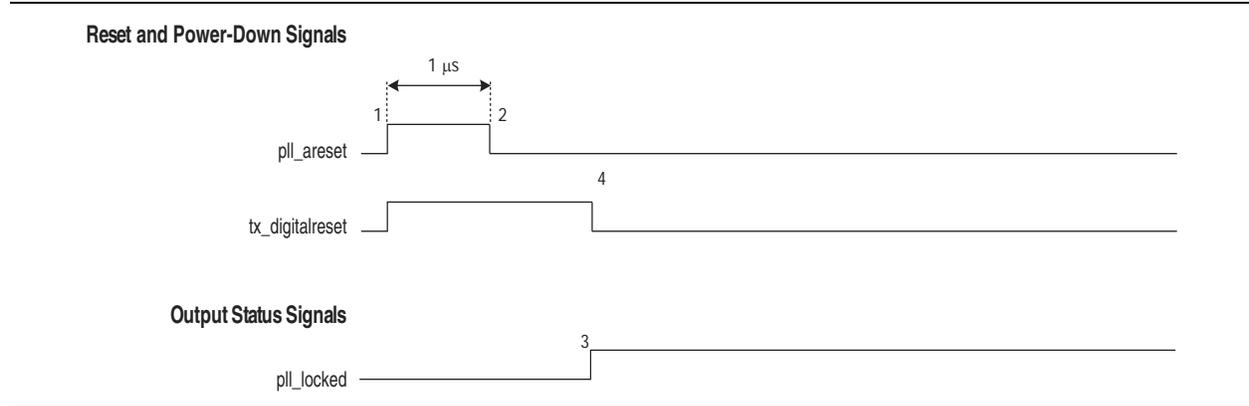
Notes to Figure 1-60:

- (1) Optional rate match FIFO.
- (2) High-speed recovered clock.
- (3) Low-speed recovered clock.

Transmitter Only Channel

This configuration contains only a transmitter channel. If you create a **Transmitter Only** instance in the ALTGX MegaWizard Plug-In Manager in Basic $\times 4$ functional mode, use the reset sequence shown in Figure 2-3.

Figure 2-3. Sample Reset Sequence for Bonded and Non-Bonded Configuration Transmitter Only Channels



As shown in Figure 2-3, perform the following reset procedure for the **Transmitter Only** channel configuration:

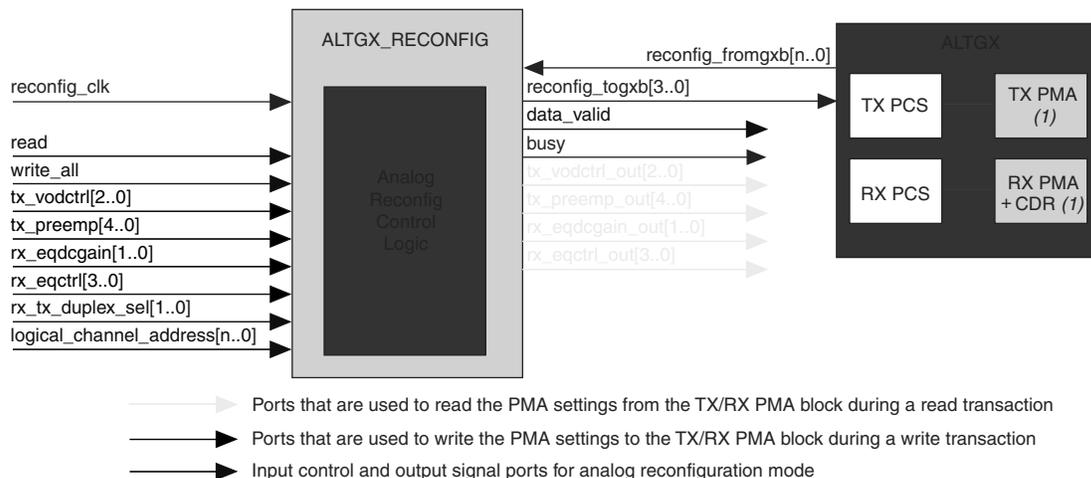
1. After power up, assert `pll_areset` for a minimum period of 1 μs (the time between markers 1 and 2).
2. Keep the `tx_digitalreset` signal asserted during this time period. After you de-assert the `pll_areset` signal, the multipurpose PLL starts locking to the transmitter input reference clock.
3. When the multipurpose PLL locks, as indicated by the `pll_locked` signal going high (marker 3), de-assert the `tx_digitalreset` signal (marker 4). At this point, the transmitter is ready for transmitting data.

Table 3-2. Dynamic Reconfiguration Controller Port List (ALTGX_RECONFIG Instance) (Part 5 of 7)

Port Name	Input/ Output	Description																																				
tx_preemp[4..0] ⁽¹⁾	Input	<p>This is an optional pre-emphasis write control for the transmit buffer. Depending on what value you set at this input, the controller dynamically writes the value to the pre-emphasis control register of the transmit buffer.</p> <p>The width of this signal is fixed to 5 bits if you enable either the Use 'logical_channel_address' port for Analog controls reconfiguration option or the Use same control signal for all the channels option in the Analog controls screen. Otherwise, the width of this signal is 5 bits per channel.</p> <table border="1"> <thead> <tr> <th>tx_preemp[4..0]</th> <th>Corresponding ALTGX instance settings</th> <th>Corresponding pre-emphasis setting (mA)</th> </tr> </thead> <tbody> <tr> <td>00000</td> <td>0</td> <td>Disabled</td> </tr> <tr> <td>00001</td> <td>1</td> <td>0.5</td> </tr> <tr> <td>00101</td> <td>5</td> <td>1.0</td> </tr> <tr> <td>01001</td> <td>9</td> <td>1.5</td> </tr> <tr> <td>01101</td> <td>13</td> <td>2.0</td> </tr> <tr> <td>10000</td> <td>16</td> <td>2.375</td> </tr> <tr> <td>10001</td> <td>17</td> <td>2.5</td> </tr> <tr> <td>10010</td> <td>18</td> <td>2.625</td> </tr> <tr> <td>10011</td> <td>19</td> <td>2.75</td> </tr> <tr> <td>10100</td> <td>20</td> <td>2.875</td> </tr> <tr> <td>10101</td> <td>21</td> <td>3.0</td> </tr> </tbody> </table> <p>All other values => N/A</p>	tx_preemp[4..0]	Corresponding ALTGX instance settings	Corresponding pre-emphasis setting (mA)	00000	0	Disabled	00001	1	0.5	00101	5	1.0	01001	9	1.5	01101	13	2.0	10000	16	2.375	10001	17	2.5	10010	18	2.625	10011	19	2.75	10100	20	2.875	10101	21	3.0
		tx_preemp[4..0]	Corresponding ALTGX instance settings	Corresponding pre-emphasis setting (mA)																																		
		00000	0	Disabled																																		
		00001	1	0.5																																		
		00101	5	1.0																																		
		01001	9	1.5																																		
		01101	13	2.0																																		
		10000	16	2.375																																		
		10001	17	2.5																																		
		10010	18	2.625																																		
		10011	19	2.75																																		
		10100	20	2.875																																		
		10101	21	3.0																																		
rx_eqctrl[3..0] ⁽¹⁾	Input	<p>This is an optional write control to write an equalization control value for the receive side of the PMA.</p> <p>The width of this signal is fixed to 4 bits if you enable either the Use 'logical_channel_address' port for Analog controls reconfiguration option or the Use same control signal for all the channels option in the Analog controls screen. Otherwise, the width of this signal is 4 bits per channel.</p> <table border="1"> <thead> <tr> <th>rx_eqctrl[3..0]</th> <th>Corresponding ALTGX instance settings</th> </tr> </thead> <tbody> <tr> <td>0001</td> <td>Low</td> </tr> <tr> <td>0101</td> <td>Medium Low</td> </tr> <tr> <td>0100</td> <td>Medium High</td> </tr> <tr> <td>0111</td> <td>High</td> </tr> </tbody> </table> <p>All other values => N/A</p>	rx_eqctrl[3..0]	Corresponding ALTGX instance settings	0001	Low	0101	Medium Low	0100	Medium High	0111	High																										
		rx_eqctrl[3..0]	Corresponding ALTGX instance settings																																			
		0001	Low																																			
		0101	Medium Low																																			
		0100	Medium High																																			
		0111	High																																			

Figure 3–9 shows the connection for PMA reconfiguration mode.

Figure 3–9. ALTGX and ALTGX_RECONFIG Connection for PMA Reconfiguration Mode



Note to Figure 3–9:

(1) This block can be reconfigured in PMA reconfiguration mode.

Transceiver Channel Reconfiguration Mode

You can dynamically reconfigure the transceiver channel from an existing functional mode to a different functional mode by selecting the **Channel Reconfiguration** option in ALTGX and ALTGX_RECONFIG MegaWizards. The blocks that are reconfigured by channel reconfiguration mode are the PCS and RX PMA blocks of a transceiver channel.

 For more information about reconfiguring the RX PMA blocks of the transceiver channel using channel reconfiguration mode, you can refer to “Data Rate Reconfiguration Mode Using RX Local Divider” on page 3–26.

In channel reconfiguration, only a write transaction can occur; no read transactions are allowed. You can optionally choose to trigger `write_all` once by selecting the continuous write operation in the ALTGX_RECONFIG MegaWizard Plug-In Manager. The Quartus II software then continuously writes all the words required for reconfiguration.

For channel reconfiguration, `.mif` files are required to dynamically reconfigure the transceivers channels in channel reconfiguration modes. The `.mif` carries the reconfiguration information that will be used to reconfigure the transceivers channel dynamically on-the-fly. The `.mif` contents is generated automatically when you select the **Generate GXB Reconfig MIF** option in the Quartus II software setting. For different `.mif` settings, you need to later reconfigure and recompile the ALTGX MegaWizard to generate the `.mif` based on the required reconfiguration settings.

The dynamic reconfiguration controller can optionally perform a continuous write operation or a regular write operation of the `.mif` contents in terms of word size (16-bit data) to the transceivers channel that is selected for reconfiguration.