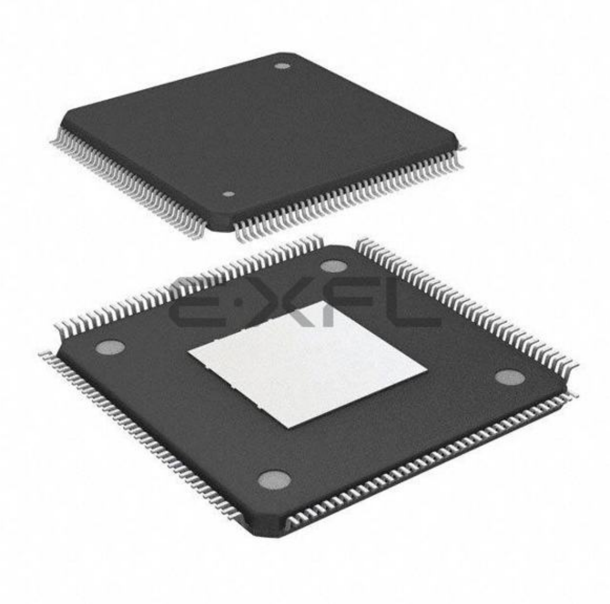
Intel - EP4CE6E22C7 Datasheet



Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

Applications of Embedded - FPGAs

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

Details	
Product Status	Active
Number of LABs/CLBs	392
Number of Logic Elements/Cells	6272
Total RAM Bits	276480
Number of I/O	91
Number of Gates	-
Voltage - Supply	1.15V ~ 1.25V
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/ep4ce6e22c7

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

Chapter Revision Dates

The chapters in this document, Cyclone IV Device Handbook,, were revised on the following dates. Where chapters or groups of chapters are available separately, part numbers are listed.

Chapter 1. Cyclone IV FPGA Device Family Overview
Revised: *March 2016*
Part Number: *CYIV-51001-2.0*

Chapter 2. Logic Elements and Logic Array Blocks in Cyclone IV Devices
Revised: *November 2009*
Part Number: *CYIV-51002-1.0*

Chapter 3. Memory Blocks in Cyclone IV Devices
Revised: *November 2011*
Part Number: *CYIV-51003-1.1*

Chapter 4. Embedded Multipliers in Cyclone IV Devices
Revised: *February 2010*
Part Number: *CYIV-51004-1.1*

Chapter 5. Clock Networks and PLLs in Cyclone IV Devices
Revised: *October 2012*
Part Number: *CYIV-51005-2.4*

Chapter 6. I/O Features in Cyclone IV Devices
Revised: *March 2016*
Part Number: *CYIV-51006-2.7*

Chapter 7. External Memory Interfaces in Cyclone IV Devices
Revised: *March 2016*
Part Number: *CYIV-51007-2.6*

Chapter 8. Configuration and Remote System Upgrades in Cyclone IV Devices
Revised: *May 2013*
Part Number: *CYIV-51008-1.7*

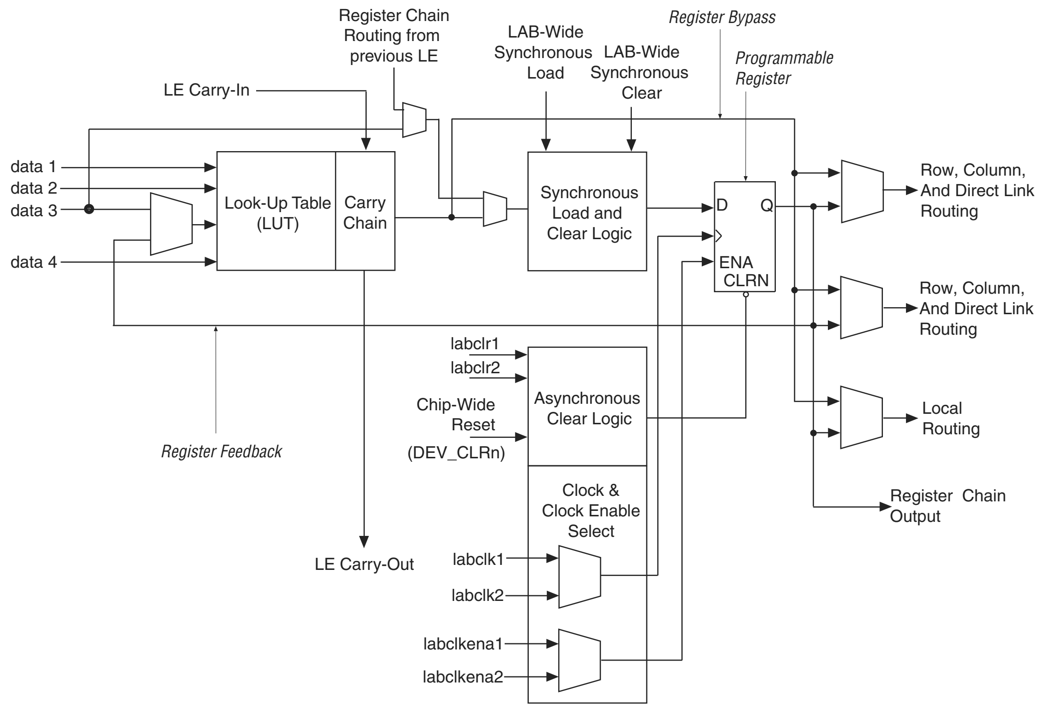
Chapter 9. SEU Mitigation in Cyclone IV Devices
Revised: *May 2013*
Part Number: *CYIV-51009-1.3*

Chapter 10. JTAG Boundary-Scan Testing for Cyclone IV Devices
Revised: *December 2013*
Part Number: *CYIV-51010-1.3*

Chapter 11. Power Requirements for Cyclone IV Devices
Revised: *May 2013*
Part Number: *CYIV-51011-1.3*

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

Figure 2–1. Cyclone IV Device LEs



LE Features

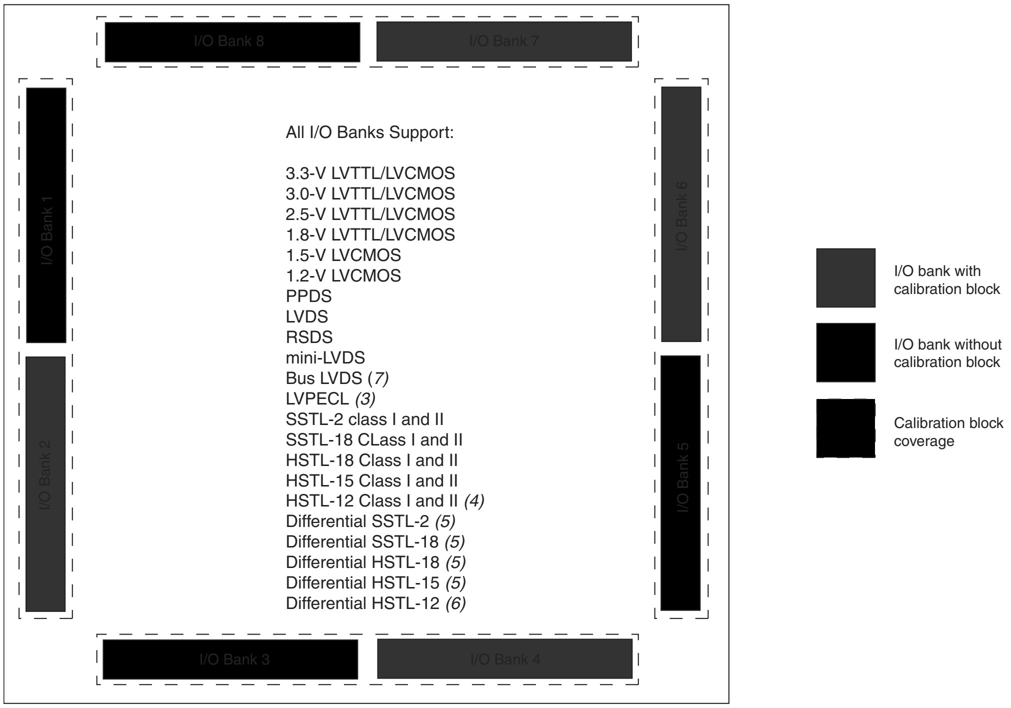
You can configure the programmable register of each LE for D, T, JK, or SR flipflop operation. Each register has data, clock, clock enable, and clear inputs. Signals that use the global clock network, general-purpose I/O pins, or any internal logic can drive the clock and clear control signals of the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

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

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

Figure 6–9 shows the overview of Cyclone IV E I/O banks.

Figure 6–9. Cyclone IV E I/O Banks *(1), (2)*



Notes to Figure 6–9:

(1) This is a top view of the silicon die. This is only a graphical representation. For exact pin locations, refer to the pin list and the Quartus II software.

(2) True differential (PPDS, LVDS, mini-LVDS, and RSDS I/O standards) outputs are supported in row I/O banks 1, 2, 5, and 6 only. External resistors are needed for the differential outputs in column I/O banks.

(3) The LVPECL I/O standard is only supported on clock input pins. This I/O standard is not supported on output pins.

(4) The HSTL-12 Class II is supported in column I/O banks 3, 4, 7, and 8 only.

(5) The differential SSTL-18 and SSTL-2, differential HSTL-18, and HSTL-15 I/O standards are supported only on clock input pins and phase-locked loops (PLLs) output clock pins. Differential SSTL-18, differential HSTL-18, and HSTL-15 I/O standards do not support Class II output.

(6) The differential HSTL-12 I/O standard is only supported on clock input pins and PLL output clock pins. Differential HSTL-12 Class II is supported only in column I/O banks 3, 4, 7, and 8.

(7) BLVDS output uses two single-ended outputs with the second output programmed as inverted. BLVDS input uses true LVDS input buffer.

Differential SSTL I/O Standard Support in Cyclone IV Devices

The differential SSTL I/O standard is a memory-bus standard used for applications such as high-speed DDR SDRAM interfaces. Cyclone IV devices support differential SSTL-2 and SSTL-18 I/O standards. The differential SSTL output standard is only supported at `PLL#_CLKOUT` pins using two single-ended SSTL output buffers (`PLL#_CLKOUTp` and `PLL#_CLKOUTn`), with the second output programmed to have opposite polarity. The differential SSTL input standard is supported on the GCLK pins only, treating differential inputs as two single-ended SSTL and only decoding one of them.

The differential SSTL I/O standard requires two differential inputs with an external reference voltage (`VREF`) as well as an external termination voltage (`VTT`) of 0.5 × V_{CCIO} to which termination resistors are connected.

For differential SSTL electrical specifications, refer to "Differential I/O Standard Termination" on page 6–15 and the *Cyclone IV Device Datasheet* chapter.

Figure 6–8 on page 6–15 shows the differential SSTL Class I and Class II interface.

Differential HSTL I/O Standard Support in Cyclone IV Devices

The differential HSTL I/O standard is used for the applications designed to operate in 0 V to 1.2 V, 0 V to 1.5 V, or 0 V to 1.8 V HSTL logic switching range. Cyclone IV devices support differential HSTL-18, HSTL-15, and HSTL-12 I/O standards. The differential HSTL input standard is available on GCLK pins only, treating the differential inputs as two single-ended HSTL and only decoding one of them. The differential HSTL output standard is only supported at the `PLL#_CLKOUT` pins using two single-ended HSTL output buffers (`PLL#_CLKOUT`*p* and `PLL#_CLKOUT`*n*), with the second output programmed to have opposite polarity.

The differential HSTL I/O standard requires two differential inputs with an external reference voltage (`VREF`), as well as an external termination voltage (`VTT`) of 0.5 × V_{CCIO} to which termination resistors are connected.

For differential HSTL signaling characteristics, refer to "Differential I/O Standard Termination" on page 6–15 and the *Cyclone IV Device Datasheet* chapter.

Figure 6–7 on page 6–15 shows the differential HSTL Class I and Class II interface.

True Differential Output Buffer Feature

Cyclone IV devices true differential transmitters offer programmable pre-emphasis—you can turn it on or off. The default setting is on.

Programmable Pre-Emphasis

The programmable pre-emphasis boosts the high frequencies of the output signal to compensate the frequency-dependant attenuation of the transmission line to maximize the data eye opening at the far-end receiver. Without pre-emphasis, the output current is limited by the V_{OD} specification and the output impedance of the transmitter. At high frequency, the slew rate may not be fast enough to reach full V_{OD}

Board Design Considerations

This section explains how to achieve the optimal performance from a Cyclone IV I/O interface and ensure first-time success in implementing a functional design with optimal signal quality. You must consider the critical issues of controlled impedance of traces and connectors, differential routing, and termination techniques to get the best performance from Cyclone IV devices.

Use the following general guidelines to improve signal quality:

- Base board designs on controlled differential impedance. Calculate and compare all parameters, such as trace width, trace thickness, and the distance between two differential traces.
- Maintain equal distance between traces in differential I/O standard pairs as much as possible. Routing the pair of traces close to each other maximizes the common-mode rejection ratio (CMRR).
- Longer traces have more inductance and capacitance. These traces must be as short as possible to limit signal integrity issues.
- Place termination resistors as close to receiver input pins as possible.
- Use surface mount components.
- Avoid 90° corners on board traces.
- Use high-performance connectors.
- Design backplane and card traces so that trace impedance matches the impedance of the connector and termination.
- Keep an equal number of vias for both signal traces.
- Create equal trace lengths to avoid skew between signals. Unequal trace lengths result in misplaced crossing points and decrease system margins as the TCCS value increases.
- Limit vias because they cause discontinuities.
- Keep switching transistor-to-transistor logic (TTL) signals away from differential signals to avoid possible noise coupling.
- Do not route TTL clock signals to areas under or above the differential signals.
- Analyze system-level signals.

For PCB layout guidelines, refer to *AN 224: High-Speed Board Layout Guidelines* and *AN 315: Guidelines for Designing High-Speed FPGA PCBs.*

Software Overview

Cyclone IV devices high-speed I/O system interfaces are created in core logic by a Quartus II software megafunction because they do not have a dedicated circuit for the SERDES. Cyclone IV devices use the I/O registers and LE registers to improve the timing performance and support the SERDES. The Quartus II software allows you to design your high-speed interfaces using ALTLVDS megafunction. This megafunction

Table 6–12. Document Revision History (Part 2 of 2)

Date	Version	Changes
February 2010	2.0	■ Added Cyclone IV E devices information for the Quartus II software version 9.1 SP1 release. ■ Updated Table 6–2, Table 6–3, and Table 6–10. ■ Updated “I/O Banks” section. ■ Added Figure 6–9. ■ Updated Figure 6–10 and Figure 6–11. ■ Added Table 6–4, Table 6–6, and Table 6–8.
November 2009	1.0	Initial release.

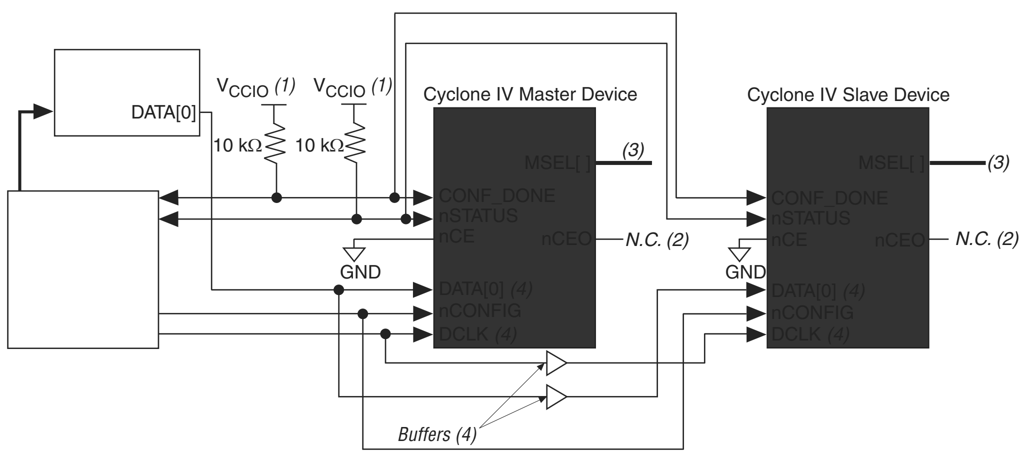
After the first device completes configuration in a multi-device configuration chain, its nCEO pin drives low to activate the nCE pin of the second device, which prompts the second device to begin configuration. The second device in the chain begins configuration in one clock cycle. Therefore, the transfer of data destinations is transparent to the external host device. nCONFIG, nSTATUS, DCLK, DATA[0], and CONF_DONE configuration pins are connected to every device in the chain. To ensure signal integrity and prevent clock skew problems, configuration signals may require buffering. Ensure that DCLK and DATA lines are buffered. All devices initialize and enter user mode at the same time because all CONF_DONE pins are tied together.

If any device detects an error, configuration stops for the entire chain and you must reconfigure the entire chain because all nSTATUS and CONF_DONE pins are tied together. For example, if the first device flags an error on nSTATUS, it resets the chain by pulling its nSTATUS pin low. This behavior is similar to a single device detecting an error.

You can have multiple devices that contain the same configuration data in your system. To support this configuration scheme, all device nCE inputs are tied to GND, while the nCEO pins are left floating. nCONFIG, nSTATUS, DCLK, DATA[0], and CONF_DONE configuration pins are connected to every device in the chain. To ensure signal integrity and prevent clock skew problems, configuration signals may require buffering. Ensure that the DCLK and DATA lines are buffered. Devices must be of the same density and package. All devices start and complete configuration at the same time.

Figure 8–15 shows a multi-device PS configuration when both Cyclone IV devices are receiving the same configuration data.

Figure 8–15. Multi-Device PS Configuration When Both Devices Receive the Same Data



Notes to Figure 8–15:

(1) You must connect the pull-up resistor 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) The nCEO pins of both devices are left unconnected or used as user I/O pins when configuring the same configuration data into multiple devices.

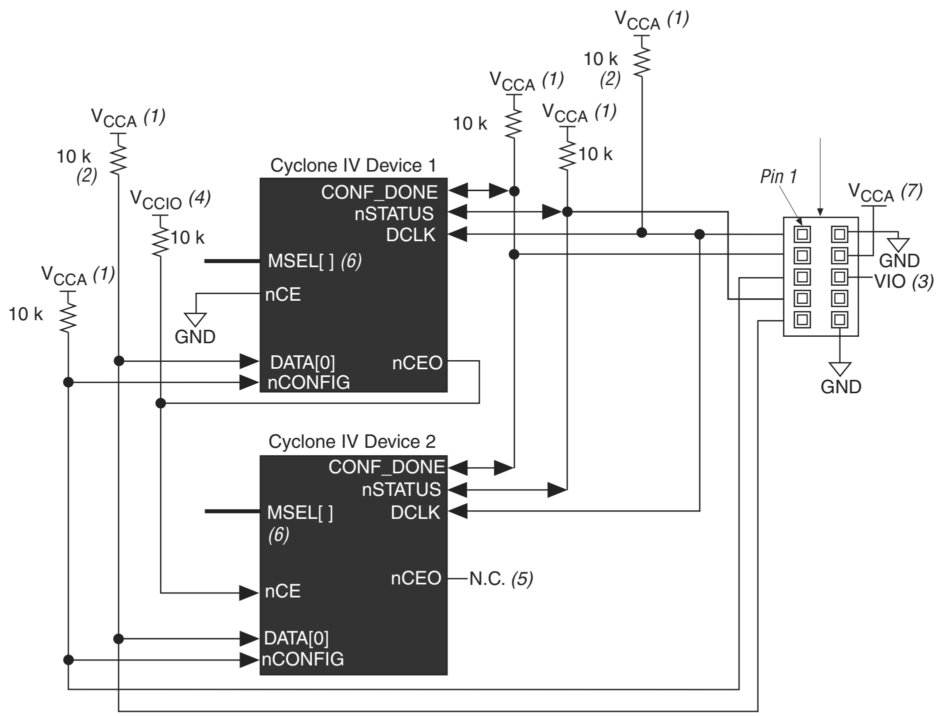
(3) The MSEL pin settings vary for different configuration voltage standards and POR time. To connect the MSEL pins, refer to Table 8–3 on page 8–8, Table 8–4 on page 8–8, and Table 8–5 on page 8–9. Connect the MSEL pins directly to V_{CCA} or GND.

(4) All I/O inputs must maintain a maximum AC voltage of 4.1 V. DATA[0] and DCLK must fit the maximum overshoot outlined in Equation 8–1 on page 8–5.

You can use a download cable to configure multiple Cyclone IV device configuration pins. `nCONFIG`, `nSTATUS`, `DCLK`, `DATA[0]`, and `CONF_DONE` are connected to every device in the chain. All devices in the chain utilize and enter user mode at the same time because all `CONF_DONE` pins are tied together.

In addition, the entire chain halts configuration if any device detects an error because the `nSTATUS` pins are tied together. Figure 8–18 shows the PS configuration for multiple Cyclone IV devices using a MasterBlaster, USB-Blaster, ByteBlaster II, or ByteBlasterMV cable.

Figure 8–18. Multi-Device PS Configuration Using a Download Cable



Notes to Figure 8–18:

(1) You must connect the pull-up resistor to the same supply voltage as the V_{CCA} supply.

(2) The pull-up resistors on `DATA[0]` and `DCLK` are only required if the download cable is the only configuration scheme used on your board. This ensures that `DATA[0]` and `DCLK` are not left floating after configuration. For example, if you also use a configuration device, the pull-up resistors on `DATA[0]` and `DCLK` are not required.

(3) Pin 6 of the header is a V_{IO} reference voltage for the MasterBlaster output driver. V_{IO} must match the V_{CCA} of the device. For this value, refer to the *MasterBlaster Serial/USB Communications Cable User Guide*. When using the ByteBlasterMV download cable, this pin is a no connect. When using USB-Blaster, ByteBlaster II, and EthernetBlaster cables, this pin is connected to `nCE` when it is used for AS programming. Otherwise, it is a no connect.

(4) Connect the pull-up resistor to the V_{CCIO} supply voltage of the I/O bank in which the `nCE` pin resides.

(5) The `nCEO` pin of the last device in the chain is left unconnected or used as a user I/O pin.

(6) The `MSEL` pin settings vary for different configuration voltage standards and POR time. To connect `MSEL` for PS configuration schemes, refer to Table 8–3 on page 8–8, 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.

(7) Power up the V_{CC} of the ByteBlaster II, USB-Blaster, or ByteBlasterMV cable with a 2.5 V supply from V_{CCA}. Third-party programmers must switch to 2.5 V. Pin 4 of the header is a V_{CC} power supply for the MasterBlaster cable. The MasterBlaster cable can receive power from either 5.0- or 3.3-V circuit boards, DC power supply, or 5.0 V from the USB cable. For this value, refer to the *MasterBlaster Serial/USB Communications Cable User Guide*.

Table 8–25 lists the contents of previous state register 1 and previous state register 2 in the status register. The status register bit in Table 8–25 shows the bit positions in a 3-bit register. The previous state register 1 and previous state register 2 have the same bit definitions. The previous state register 1 reflects the current application configuration and the previous state register 2 reflects the previous application configuration.

Table 8–25. Remote System Upgrade Previous State Register 1 and Previous State Register 2 Contents in Status Register

Status Register Bit	Definition	Description
30	nCONFIG source	One-hot, active-high field that describes the reconfiguration source that caused the Cyclone IV device to leave the previous application configuration. If there is a tie, the higher bit order indicates precedence. For example, if nCONFIG and remote system upgrade nCONFIG reach the reconfiguration state machine at the same time, the nCONFIG precedes the remote system upgrade nCONFIG.
29	CRC error source	
28	nSTATUS source	
27	User watchdog timer source	
26	Remote system upgrade nCONFIG source	
25:24	Master state machine current state	The state of the master state machine during reconfiguration causes the Cyclone IV device to leave the previous application configuration.
23:0	Boot address	The address used by the configuration scheme to load the previous application configuration.

If a capture is inappropriately done while capturing a previous state before the system has entered remote update application configuration for the first time, a value outputs from the shift register to indicate that the capture is incorrectly called.

Remote System Upgrade State Machine

The remote system upgrade control and update registers have identical bit definitions, but serve different roles (Table 8–22 on page 8–75). While both registers can only be updated when the device is loaded with a factory configuration image, the update register writes are controlled by the user logic, and the control register writes are controlled by the remote system upgrade state machine.

In factory configurations, the user logic should send the option bits (Cd_early and Osc_int), the configuration address, and watchdog timer settings for the next application configuration bit to the update register. When the logic array configuration reset (RU_nCONFIG) goes high, the remote system upgrade state machine updates the control register with the contents of the update register and starts system reconfiguration from the new application page.

☞ To ensure the successful reconfiguration between the pages, assert the RU_nCONFIG signal for a minimum of 250 ns. This is equivalent to strobing the reconfig input of the ALTREMOTE_UPDATE megafunction high for a minimum of 250 ns.

If there is an error or reconfiguration trigger condition, the remote system upgrade state machine directs the system to load a factory or application configuration (based on mode and error condition) by setting the control register accordingly.

Table 8–26 lists the contents of the control register after such an event occurs for all possible error or trigger conditions.

The remote system upgrade status register is updated by the dedicated error monitoring circuitry after an error condition, but before the factory configuration is loaded.

Table 8–26. Control Register Contents After an Error or Reconfiguration Trigger Condition

Reconfiguration Error/Trigger	Control Register Setting In Remote Update
`nCONFIG` reset	All bits are 0
`nSTATUS` error	All bits are 0
CORE triggered reconfiguration	Update register
CRC error	All bits are 0
Wd time out	All bits are 0

User Watchdog Timer

The user watchdog timer prevents a faulty application configuration from indefinitely stalling the device. The system uses the timer to detect functional errors after an application configuration is successfully loaded into the Cyclone IV device.

The user watchdog timer is a counter that counts down from the initial value loaded into the remote system upgrade control register by the factory configuration. The counter is 29 bits wide and has a maximum count value of 2^{29}. When specifying the user watchdog timer value, specify only the most significant 12 bits. The remote system upgrade circuitry appends 17'b1000 to form the 29-bits value for the watchdog timer. The granularity of the timer setting is 2^{17} cycles. The cycle time is based on the frequency of the 10-MHz internal oscillator or `CLKUSR` (maximum frequency of 40 MHz).

Table 8–27 lists the operating range of the 10-MHz internal oscillator.

Table 8–27. 10-MHz Internal Oscillator Specifications

Minimum	Typical	Maximum	Unit
5	6.5	10	MHz

The user watchdog timer begins counting after the application configuration enters device user mode. This timer must be periodically reloaded or reset by the application configuration before the timer expires by asserting `RU_nRSTIMER`. If the application configuration does not reload the user watchdog timer before the count expires, a time-out signal is generated by the remote system upgrade dedicated circuitry. The time-out signal tells the remote system upgrade circuitry to set the user watchdog timer status bit (`Wd`) in the remote system upgrade status register and reconfigures the device by loading the factory configuration.

To allow the remote system upgrade dedicated circuitry to reset the watchdog timer, you must assert the `RU_nRSTIMER` signal active for a minimum of 250 ns. This is equivalent to strobing the `reset_timer` input of the ALTREMOTE_UPDATE megafunction high for a minimum of 250 ns.

Errors during configuration are detected by the CRC engine. Functional errors must not exist in the factory configuration because it is stored and validated during production and is never updated remotely.

9. SEU Mitigation in Cyclone IV Devices

CYIV-51009-1.3

This chapter describes the cyclical redundancy check (CRC) error detection feature in user mode and how to recover from soft errors.

☞ Configuration error detection is supported in all Cyclone® IV devices including Cyclone IV GX devices, Cyclone IV E devices with 1.0-V core voltage, and Cyclone IV E devices with 1.2-V core voltage. However, user mode error detection is only supported in Cyclone IV GX devices and Cyclone IV E devices with 1.2-V core voltage.

Dedicated circuitry built into Cyclone IV devices consists of a CRC error detection feature that can optionally check for a single-event upset (SEU) continuously and automatically.

In critical applications used in the fields of avionics, telecommunications, system control, medical, and military applications, it is important to be able to:

- Confirm the accuracy of the configuration data stored in an FPGA device
- Alert the system to an occurrence of a configuration error

Using the CRC error detection feature for Cyclone IV devices does not impact fitting or performance.

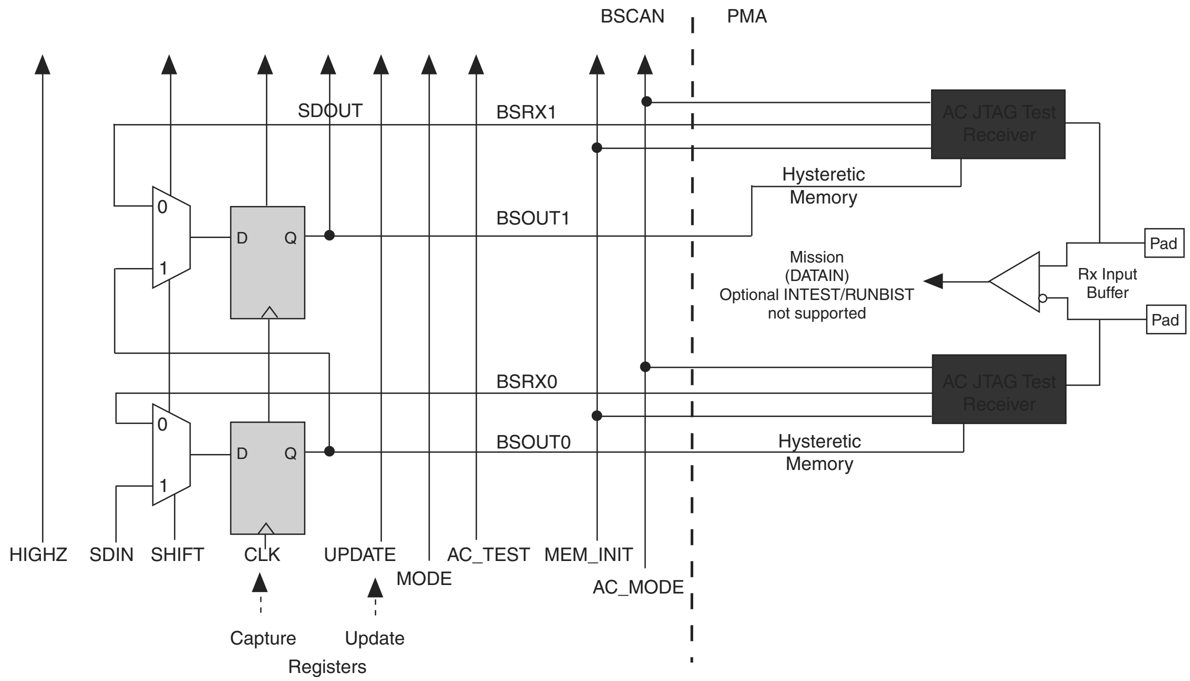
This chapter contains the following sections:

Configuration Error Detection

☞ Configuration error detection is available in all Cyclone IV devices including Cyclone IV GX devices, Cyclone IV E devices with 1.0-V core voltage, and Cyclone IV E devices with 1.2-V core voltage.

Figure 10–2 shows the Cyclone IV GX HSSI receiver BSC.

Figure 10–2. HSSI Receiver BSC with IEEE Std. 1149.6 BST Circuitry for the Cyclone IV GX Devices



For more information about Cyclone IV devices user I/O boundary-scan cells, refer to the *IEEE 1149.1 (JTAG) Boundary-Scan Testing for Cyclone III Devices* chapter.

BST Operation Control

Table 10–1 lists the boundary-scan register length for Cyclone IV devices.

Table 10–1. Boundary-Scan Register Length for Cyclone IV Devices (Part 1 of 2)

Device	Boundary-Scan Register Length
EP4CE6	603
EP4CE10	603
EP4CE15	1080
EP4CE22	732
EP4CE30	1632
EP4CE40	1632
EP4CE55	1164
EP4CE75	1314
EP4CE115	1620
EP4CGX15	260
EP4CGX22	494
EP4CGX30 (1)	494
EP4CGX50	1006

Figure 1–50 and Figure 1–51 show the detection mechanism example for a successful and unsuccessful receiver detection scenarios respectively. The `tx_forceelecidle` port must be asserted at least 10 parallel clock cycles prior to assertion of `tx_detectrxloop` port to ensure the transmitter buffer is properly tri-stated. Detection completion is indicated by `pipephydonestatus` assertion, with detection successful indicated by 3'b011 on `pipestatus[2..0]` port, or detection unsuccessful by 3'b000 on `pipestatus[2..0]` port.

Figure 1–50. Example of Successful Receiver Detect Operation

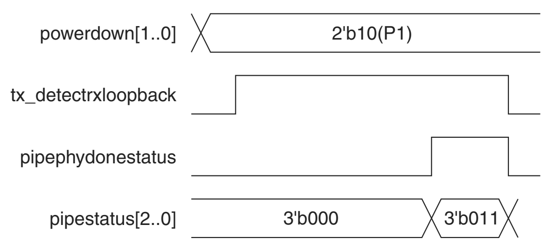
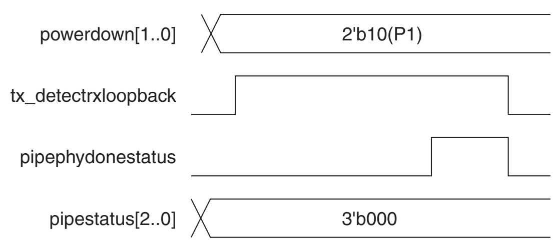


Figure 1–51. Example of Unsuccessful Receiver Detect Operation

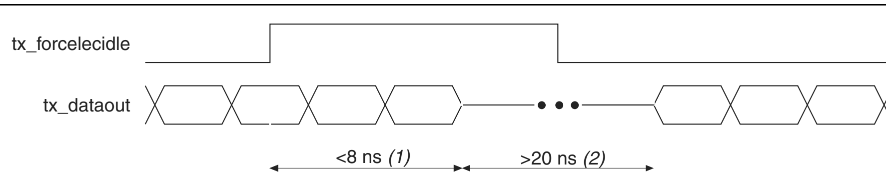


Electrical Idle Control

The Cyclone IV GX transceivers support transmitter buffer in electrical idle state using the `tx_forceelecidle` port. During electrical idle, the transmitter buffer differential and common mode output voltage levels are compliant to the PCIe Base Specification 2.0 for Gen1 signaling rate.

Figure 1–52 shows the relationship between assertion of the `tx_forceelecidle` port and the transmitter buffer output on the `tx_dataout` port.

Figure 1–52. Transmitter Buffer Electrical Idle State



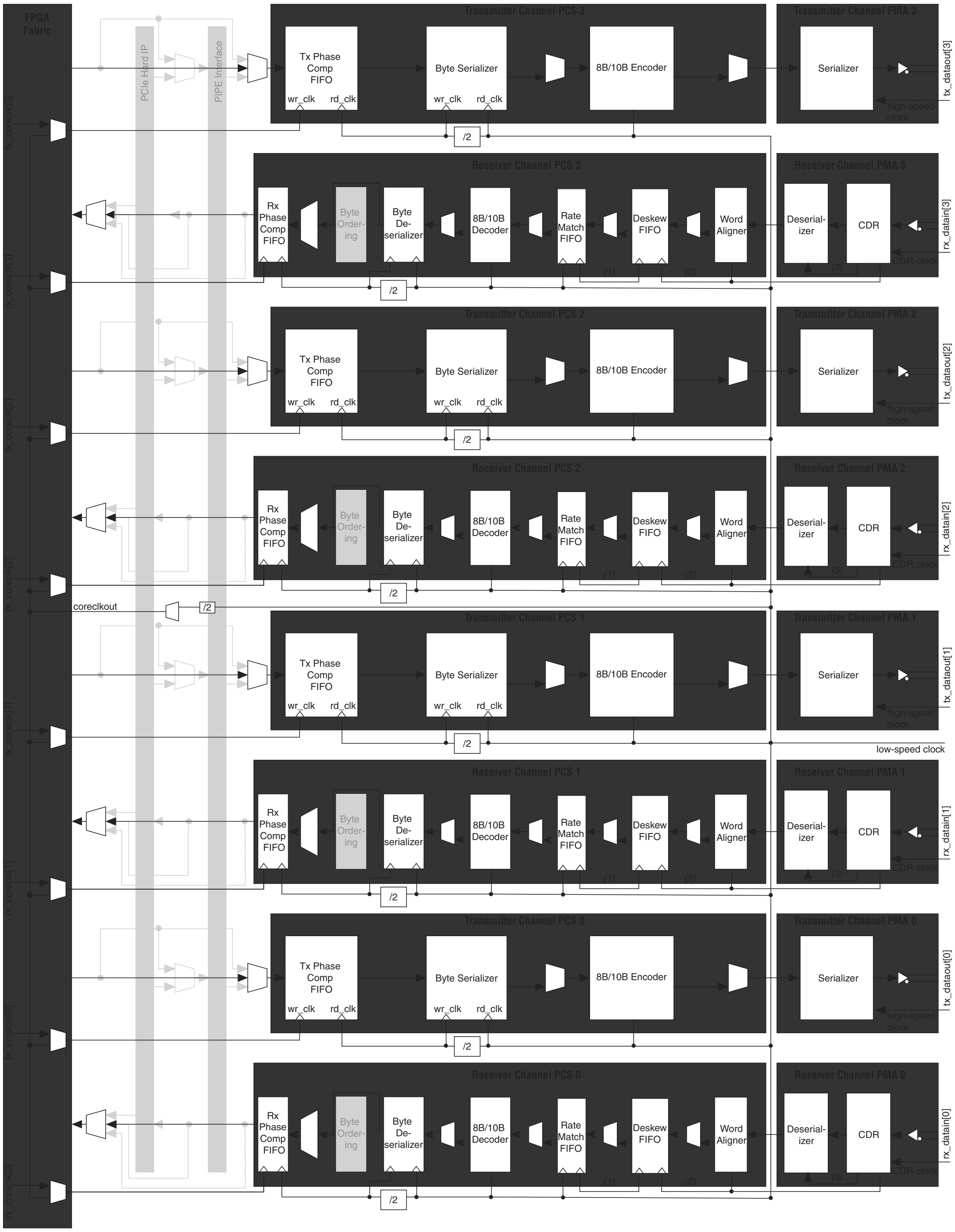
Notes to Figure 1–52:

(1) The protocol requires the transmitter buffer to transition to a valid electrical idle after sending an electrical idle ordered set within 8 ns.

(2) The protocol requires transmitter buffer to stay in electrical idle for a minimum of 20 ns for Gen1 signaling rate.

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.

Table 2–3. Blocks Affected by Reset and Power-Down Signals (Part 2 of 2)

Transceiver Block	rx_digitalreset	rx_analogreset	tx_digitalreset	pll_areset	gxb_powerdown
Serializer	—	—	✓	—	✓
Transmitter Buffer	—	—	—	—	✓
Transmitter XAUI State Machine	—	—	✓	—	✓
Receiver Buffer	—	—	—	—	✓
Receiver CDR	—	✓	—	—	✓
Receiver Deserializer	—	—	—	—	✓
Receiver Word Aligner	✓	—	—	—	✓
Receiver Deskew FIFO	✓	—	—	—	✓
Receiver Clock Rate Compensation FIFO	✓	—	—	—	✓
Receiver 8B/10B Decoder	✓	—	—	—	✓
Receiver Byte Deserializer	✓	—	—	—	✓
Receiver Byte Ordering	✓	—	—	—	✓
Receiver Phase Compensation FIFO	✓	—	—	—	✓
Receiver XAUI State Machine	✓	—	—	—	✓
BIST Verifiers	✓	—	—	—	✓

Transceiver Reset Sequences

You can configure transceiver channels in Cyclone IV GX devices in various configurations. In all functional modes except XAUI functional mode, transceiver channels can be either bonded or non-bonded. In XAUI functional mode, transceiver channels must be bonded. In PCI Express® (PCIe®) functional mode, transceiver channels can be either bonded or non-bonded and need to follow a specific reset sequence.

The two categories of reset sequences for Cyclone IV GX devices described in this chapter are:

- “All Supported Functional Modes Except the PCIe Functional Mode” on page 2–6—describes the reset sequences in bonded and non-bonded configurations.
- “PCIe Functional Mode” on page 2–17—describes the reset sequence for the initialization/compliance phase and the normal operation phase in PCIe functional modes.

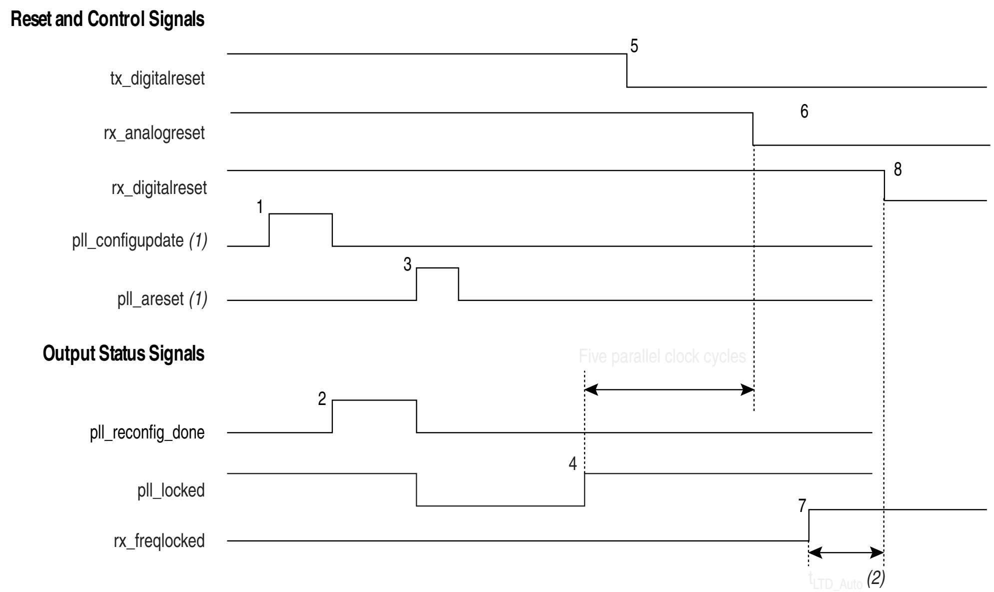
Dynamic Reconfiguration Reset Sequences

When using dynamic reconfiguration in data rate divisions in PLL reconfiguration or channel reconfiguration mode, use the following reset sequences.

Reset Sequence in PLL Reconfiguration Mode

Use the example reset sequence shown in Figure 2–11 when you use the PLL dynamic reconfiguration controller to change the data rate of the transceiver channel. In this example, PLL dynamic reconfiguration is used to dynamically reconfigure the data rate of the transceiver channel configured in Basic ×1 mode with the receiver CDR in automatic lock mode.

Figure 2–11. Reset Sequence When Using the PLL Dynamic Reconfiguration Controller to Change the Data Rate of the Transceiver Channel



Notes to Figure 2–11:

(1) The `pll_configupdate` and `pll_areset` signals are driven by the ALTPLL_RECONFIG megafunction. For more information, refer to *AN 609: Implementing Dynamic Reconfiguration in Cyclone IV GX Devices* and the *Cyclone IV Dynamic Reconfiguration* chapter.

(2) For t_{LTD_Auto} duration, refer to the *Cyclone IV Device Datasheet* chapter.

As shown in Figure 2–11, perform the following reset procedure when using the PLL dynamic reconfiguration controller to change the configuration of the PLLs in the transmitter channel:

1. Assert the `tx_digitalreset`, `rx_analogreset`, and `rx_digitalreset` signals. The `pll_configupdate` signal is asserted (marker 1) by the ALTPLL_RECONFIG megafunction after the final data bit is sent out. The `pll_reconfig_done` signal is asserted (marker 2) to inform the ALTPLL_RECONFIG megafunction that the scan chain process is completed. The ALTPLL_RECONFIG megafunction then asserts the `pll_areset` signal (marker 3) to reset the transceiver PLL.

The following are the channel reconfiguration mode options:

- Channel interface reconfiguration
- Data rate division at receiver channel

Channel Interface Reconfiguration Mode

Enable this option if the reconfiguration of the transceiver channel involves the following changes:

- The reconfigured channel has a changed FPGA fabric-Transceiver channel interface data width
- The reconfigured channel has changed input control signals and output status signals
- The reconfigured channel has enabled and disabled the static PCS blocks of the transceiver channel

The following are the new input signals available when you enable this option:

- `tx_datainfull`—the width of this input signal depends on the number of channels you set up in the ALTGX MegaWizard Plug-In Manager. It is 22 bits wide per channel. This signal is available only for **Transmitter only** and **Receiver and Transmitter** configurations. This port replaces the existing `tx_datain` port.
- `rx_dataoutfull`—the width of this output signal depends on the number of channels you set up in the ALTGX MegaWizard Plug-In Manager. It is 32 bits wide per channel. This signal is available only for **Receiver only** and **Receiver and Transmitter** configurations. This port replaces the existing `rx_dataout` port.

  The Quartus II software has legality checks for the connectivity of `tx_datainfull` and `rx_dataoutfull` and the various control and status signals you enable in the **Clocking/Interface** screen. For example, the Quartus II software allows you to select and connect the `pipestatus` and `powerdn` signals. It assumes that you are planning to switch to and from PCI Express (PIPE) functional mode.

Control and Status Signals for Channel Reconfiguration

The various control and status signals involved in the Channel Reconfiguration mode are as follows. Refer to “Dynamic Reconfiguration Controller Port List” on page 3–4 for the descriptions of the control and status signals.

The following are the input control signals:

- `logical_channel_address[n..0]`
- `reset_reconfig_address`
- `reconfig_reset`
- `reconfig_mode_sel[2..0]`
- `write_all`

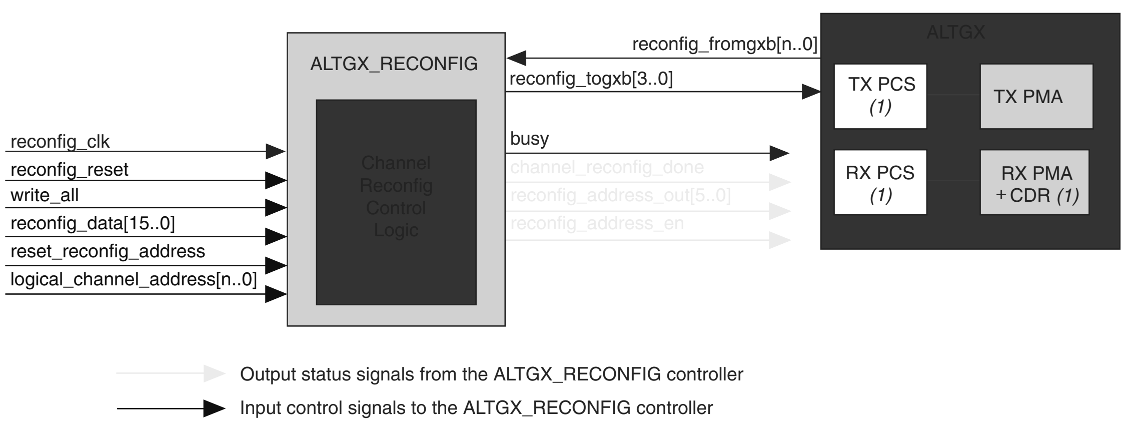
The following are output status signals:

- `reconfig_address_en`
- `reconfig_address_out[5..0]`
- `channel_reconfig_done`
- `busy`

The ALTGX_RECONFIG connection to the ALTGX instances when set in channel reconfiguration mode are as follows. For the port information, refer to “Dynamic Reconfiguration Controller Port List” on page 3–4.

Figure 3–10 shows the connection for channel reconfiguration mode.

Figure 3–10. ALTGX and ALTGX_RECONFIG Connection for Channel Reconfiguration Mode



Note to Figure 3–10:

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

Figure 1–2 shows the lock time parameters in manual mode.

☞ LTD = lock-to-data. LTR = lock-to-reference.

Figure 1–2. Lock Time Parameters for Manual Mode

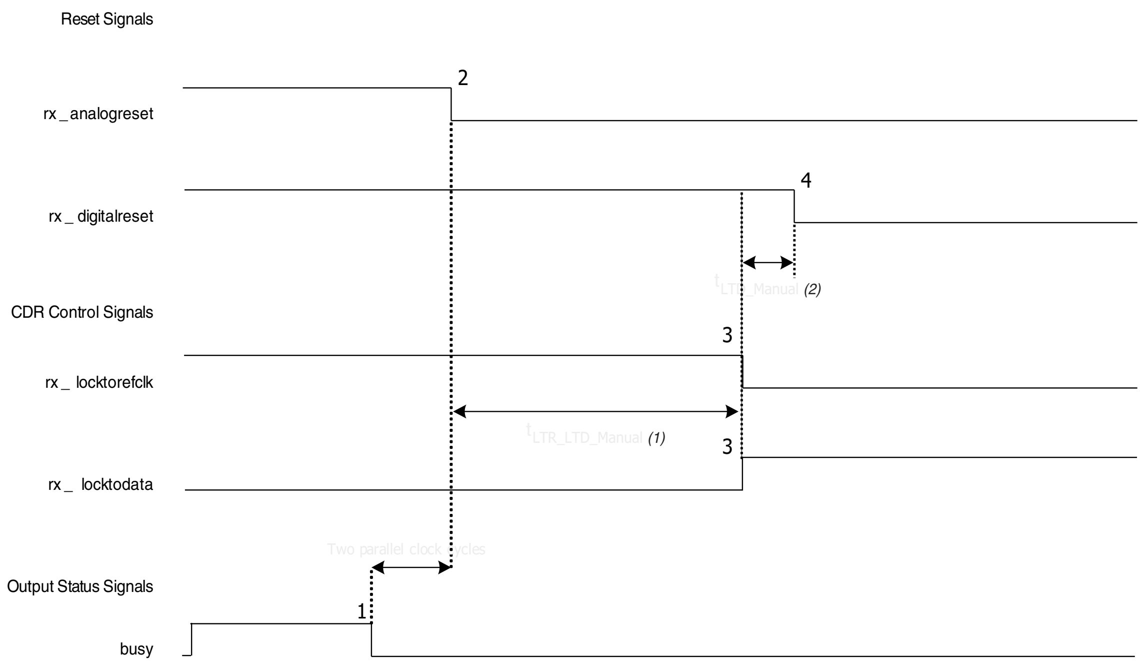


Figure 1–3 shows the lock time parameters in automatic mode.

Figure 1–3. Lock Time Parameters for Automatic Mode