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	3491
Number of Logic Elements/Cells	55856
Total RAM Bits	2396160
Number of I/O	324
Number of Gates	-
Voltage - Supply	0.97V ~ 1.03V
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/ep4ce55f23c9ln

Table 1–4 lists Cyclone IV GX device package offerings, including I/O and transceiver counts.

Table 1–4. Package Offerings for the Cyclone IV GX Device Family (1)

Package	F169			F324			F484			F672			F896		
Size (mm)	14 × 14			19 × 19			23 × 23			27 × 27			31 × 31		
Pitch (mm)	1.0			1.0			1.0			1.0			1.0		
Device	User I/O	LVDS (2)	XCVRs	User I/O	LVDS (2)	XCVRs	User I/O	LVDS (2)	XCVRs	User I/O	LVDS (2)	XCVRs	User I/O	LVDS (2)	XCVRs
EP4CGX15	72	25	2	—	—	—	—	—	—	—	—	—	—	—	—
EP4CGX22	72	25	2	150	64	4	—	—	—	—	—	—	—	—	—
EP4CGX30	72	25	2	150	64	4	290	130	4	—	—	—	—	—	—
EP4CGX50	—	—	—	—	—	—	290	130	4	310	140	8	—	—	—
EP4CGX75	—	—	—	—	—	—	290	130	4	310	140	8	—	—	—
EP4CGX110	—	—	—	—	—	—	270	120	4	393	181	8	475	220	8
EP4CGX150	—	—	—	—	—	—	270	120	4	393	181	8	475	220	8

Note to Table 1–4:

(1) Use the Pin Migration View window in Pin Planner of the Quartus II software to verify the pin migration compatibility when you perform device migration. For more information, refer to the *I/O Management* chapter in volume 2 of the *Quartus II Handbook*.

(2) This includes both dedicated and emulated LVDS pairs. For more information, refer to the *I/O Features in Cyclone IV Devices* chapter.

2. Logic Elements and Logic Array Blocks in Cyclone IV Devices

CYIV-51002-1.0

This chapter contains feature definitions for logic elements (LEs) and logic array blocks (LABs). Details are provided on how LEs work, how LABs contain groups of LEs, and how LABs interface with the other blocks in Cyclone® IV devices.

Logic Elements

Logic elements (LEs) are the smallest units of logic in the Cyclone IV device architecture. LEs are compact and provide advanced features with efficient logic usage. Each LE has the following features:

- A four-input look-up table (LUT), which can implement any function of four variables
- A programmable register
- A carry chain connection
- A register chain connection
- The ability to drive the following interconnects:
  - Local
  - Row
  - Column
  - Register chain
  - Direct link
- Register packing support
- Register feedback support

© 2009 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at www.altera.com/common/legal.html. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO 9001:2008 Registered

Read or Write Clock Mode

Cyclone IV devices M9K memory blocks can implement read or write clock mode for FIFO and simple dual-port memories. In this mode, a write clock controls the data inputs, write address, and `wren` registers. Similarly, a read clock controls the data outputs, read address, and `rden` registers. M9K memory blocks support independent clock enables for both the read and write clocks.

When using read or write mode, if you perform a simultaneous read or write to the same address location, the output read data is unknown. If you require the output data to be a known value, use either single-clock mode, input clock mode, or output clock mode and choose the appropriate read-during-write behavior in the MegaWizard Plug-In Manager.

Single-Clock Mode

Cyclone IV devices M9K memory blocks can implement single-clock mode for FIFO, ROM, true dual-port, simple dual-port, and single-port memories. In this mode, you can control all registers of the M9K memory block with a single clock together with clock enable.

Design Considerations

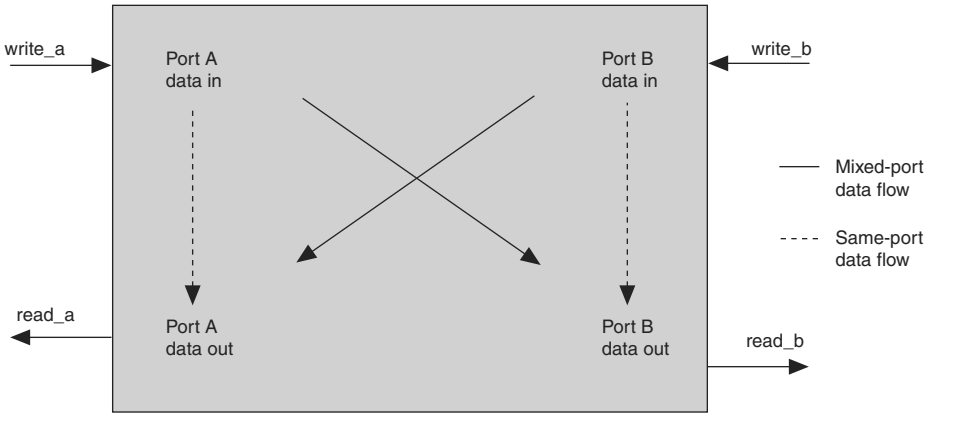
This section describes designing with M9K memory blocks.

Read-During-Write Operations

"Same-Port Read-During-Write Mode" on page 3–16 and "Mixed-Port Read-During-Write Mode" on page 3–16 describe the functionality of the various RAM configurations when reading from an address during a write operation at that same address.

There are two read-during-write data flows: same-port and mixed-port. Figure 3–13 shows the difference between these flows.

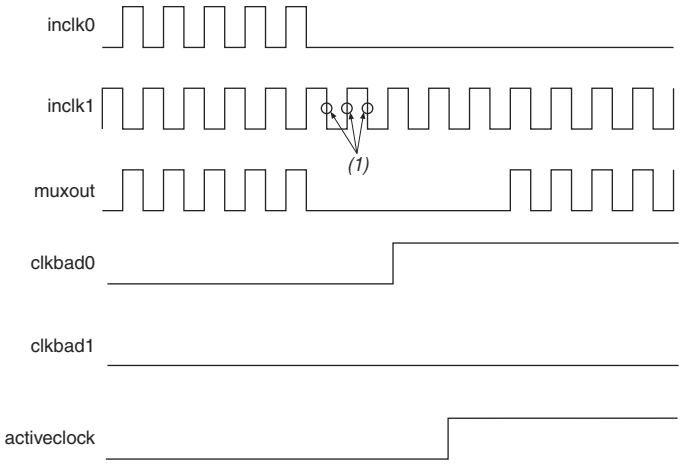
Figure 3–13. Cyclone IV Devices Read-During-Write Data Flow



20%. This feature is useful when clock sources can originate from multiple cards on the backplane, requiring a system-controlled switchover between frequencies of operation. Choose the secondary clock frequency so the VCO operates in the recommended frequency range. Also, set the M, N, and C counters accordingly to keep the VCO operating frequency in the recommended range.

Figure 5–18 shows a waveform example of the switchover feature when using automatic loss of clock detection. Here, the `inclk0` signal remains low. After the `inclk0` signal remains low for approximately two clock cycles, the clock-sense circuitry drives the `clkbad0` signal high. Also, because the reference clock signal is not toggling, the switchover state machine controls the multiplexer through the `clksw` signal to switch to `inclk1`.

Figure 5–18. Automatic Switchover Upon Clock Loss Detection *(1)*



Note to Figure 5–18:

(1) Switchover is enabled on the falling edge of `inclk0` or `inclk1`, depending on which clock is available. In this figure, switchover is enabled on the falling edge of `inclk1`.

Manual Override

If you are using the automatic switchover, you must switch input clocks with the manual override feature with the `clkswitch` input.

Figure 5–19 shows an example of a waveform illustrating the switchover feature when controlled by `clkswitch`. In this case, both clock sources are functional and `inclk0` is selected as the reference clock. A low-to-high transition of the `clkswitch` signal starts the switchover sequence. The `clkswitch` signal must be high for at least three clock cycles (at least three of the longer clock period if `inclk0` and `inclk1` have different frequencies). On the falling edge of `inclk0`, the reference clock of the counter, `muxout`, is gated off to prevent any clock glitching. On the falling edge of `inclk1`, the reference clock multiplexer switches from `inclk0` to `inclk1` as the PLL reference, and the `activeclock` signal changes to indicate which clock is currently feeding the PLL.

Table 8–8 provides the configuration time for AS configuration.

Table 8–8. AS Configuration Time for Cyclone IV Devices (1)

Symbol	Parameter	Cyclone IV E	Cyclone IV GX	Unit
t_{SU}	Setup time	10	8	ns
t_H	Hold time	0	0	ns
t_{CO}	Clock-to-output time	4	4	ns

Note to Table 8–8:

(1) For the AS configuration timing diagram, refer to the *Serial Configuration (EPCS) Devices Datasheet*.

Enabling compression reduces the amount of configuration data that is sent to the Cyclone IV device, which also reduces configuration time. On average, compression reduces configuration time by 50%.

Programming Serial Configuration Devices

Serial configuration devices are non-volatile, flash memory-based devices. You can program these devices in-system with the USB-Blaster™ or ByteBlaster™ II download cables. Alternatively, you can program them with the Altera Programming Unit (APU), supported third-party programmers, or a microprocessor with the SRunner software driver.

You can perform in-system programming of serial configuration devices through the AS programming interface. During in-system programming, the download cable disables device access to the AS interface by driving the nCE pin high. Cyclone IV devices are also held in reset by a low level on nCONFIG. After programming is complete, the download cable releases nCE and nCONFIG, allowing the pull-down and pull-up resistors to drive V_{CC} and GND, respectively.

To perform in-system programming of a serial configuration device through the AS programming interface, you must place the diodes and capacitors as close as possible to the Cyclone IV device. You must ensure that the diodes and capacitors maintain a maximum AC voltage of 4.1 V (Figure 8–6).

If you want to use the setup shown in Figure 8–6 to perform in-system programming of a serial configuration device and single- or multi-device AS configuration, you do not require a series resistor on the DATA line at the near end of the serial configuration device. The existing diodes and capacitors are sufficient.

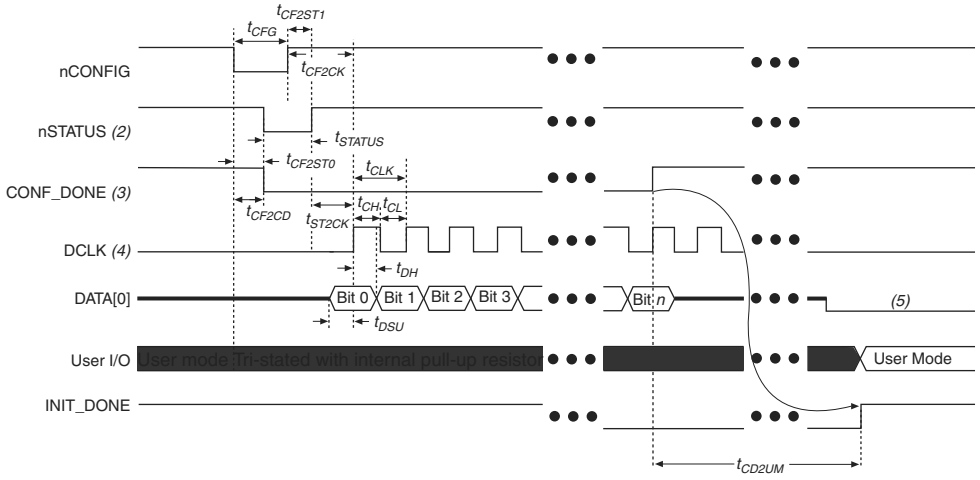
Altera has developed the Serial FlashLoader (SFL), a JTAG-based in-system programming solution for Altera serial configuration devices. The SFL is a bridge design for the Cyclone IV device that uses its JTAG interface to access the EPCS JIC (JTAG Indirect Configuration Device Programming) file and then uses the AS interface to program the EPCS device. Both the JTAG interface and AS interface are bridged together inside the SFL design.

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

PS Configuration Timing

A PS configuration must meet the setup and hold timing parameters and the maximum clock frequency. When using a microprocessor or another intelligent host to control the PS interface, ensure that you meet these timing requirements. Figure 8–16 shows the timing waveform for PS configuration when using an external host device.

Figure 8–16. PS Configuration Timing Waveform *(1)*



Notes to Figure 8–16:

(1) The beginning of this waveform shows the device in user mode. In user mode, nCONFIG, nSTATUS, and CONF_DONE are at logic-high levels. When nCONFIG is pulled low, a reconfiguration cycle begins.

(2) After power up, the Cyclone IV device holds nSTATUS low during POR delay.

(3) After power up, before and during configuration, CONF_DONE is low.

(4) In user mode, drive DCLK either high or low when using the PS configuration scheme, whichever is more convenient. When using the AS configuration scheme, DCLK is a Cyclone IV device output pin and must not be driven externally.

(5) Do not leave the DATA[0] pin floating after configuration. Drive the DATA[0] pin high or low, whichever is more convenient.

Table 8–12 lists the PS configuration timing parameters for Cyclone IV devices.

Table 8–12. PS Configuration Timing Parameters For Cyclone IV Devices (Part 1 of 2)

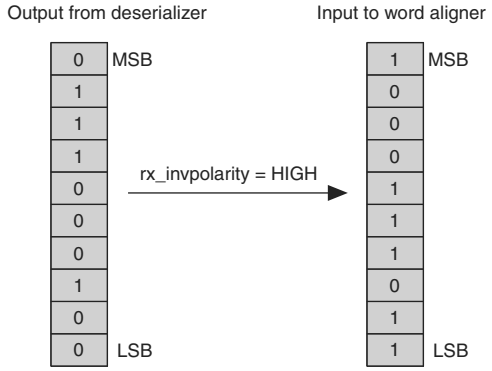
Symbol	Parameter	Minimum		Maximum		Unit
		Cyclone IV *(1)*	Cyclone IV E *(2)*	Cyclone IV *(1)*	Cyclone IV E *(2)*	
t_{CF2CD}	nCONFIG low to CONF_DONE low	—		500		ns
t_{CF2ST0}	nCONFIG low to nSTATUS low	—		500		ns
t_{CFG}	nCONFIG low pulse width	500		—		ns
t_{STATUS}	nSTATUS low pulse width	45		230 *(3)*		µs

Table 8–28. Document Revision History (Part 2 of 2)

Date	Version	Changes
July 2010	1.2	Updated for the Quartus II software 10.0 release: ■ Updated "Power-On Reset (POR) Circuit", "Configuration and JTAG Pin I/O Requirements", and "Reset" sections. ■ Updated Figure 8–10. ■ Updated Table 8–16 and Table 8–17.
February 2010	1.1	Updated for the Quartus II software 9.1 SP1 release: ■ Added "Overriding the Internal Oscillator" and "AP Configuration (Supported Flash Memories)" sections. ■ Updated "JTAG Instructions" section. ■ Added Table 8–6. ■ Updated Table 8–2, Table 8–3, Table 8–4, Table 8–6, Table 8–11, Table 8–13, Table 8–14, Table 8–15, and Table 8–18. ■ Updated Figure 8–4, Figure 8–5, Figure 8–6, Figure 8–13, Figure 8–14, Figure 8–15, Figure 8–17, Figure 8–18, Figure 8–23, Figure 8–24, Figure 8–25, Figure 8–26, Figure 8–27, Figure 8–28, and Figure 8–29.
November 2009	1.0	Initial release.

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

Figure 1–19. Receiver Polarity Inversion



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

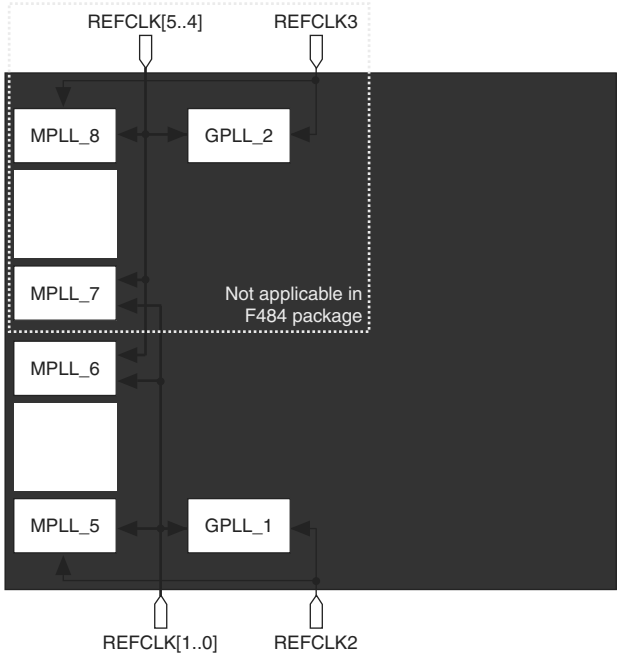
The `rx_invpolarity` signal is dynamic and might cause initial disparity errors in an 8B/10B encoded link. The downstream system must be able to tolerate these disparity errors.

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

Transceiver Clocking Architecture

The multipurpose PLLs and general-purpose PLLs located on the left side of the device generate the clocks required for the transceiver operation. The following sections describe the Cyclone IV GX transceiver clocking architecture:

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

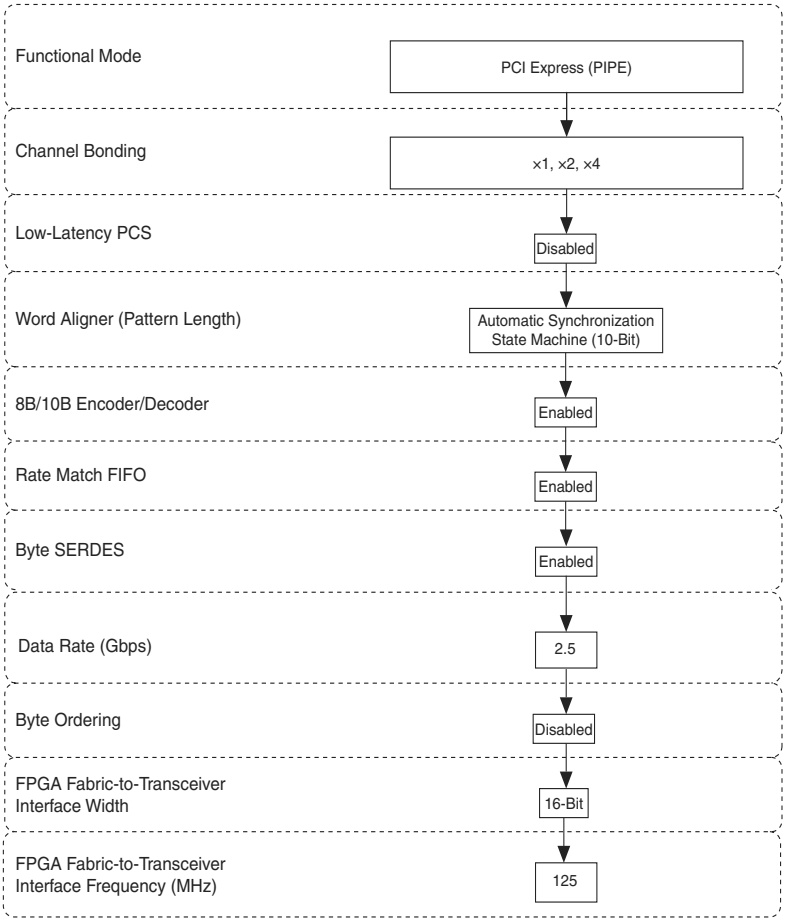
I/O Standard	HSSI Protocol	Coupling	Termination	VCC_CLKIN Level		I/O Pin Type		
				Input	Output	Column I/O	Row I/O	Supported Banks
LVDS	ALL	Differential AC (Needs off-chip resistor to restore V_{CM})	Off-chip	2.5 V	Not Supported	Yes	No	3A, 3B, 8A, 8B
LVPECL	ALL		Off-chip	2.5 V	Not Supported	Yes	No	3A,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

Configuring the hard IP module requires using the PCI Express Compiler. When configuring the transceiver for PCIe implementation with hard IP module, the byte serializer and deserializer are not enabled, providing an 8-bit transceiver-PIPE-hard IP data interface width running at 250 MHz clock frequency.

For more information about PCIe implementation with hard IP module, refer to the *PCI Express Compiler User Guide*.

Figure 1–49 shows the transceiver configuration in PIPE mode.

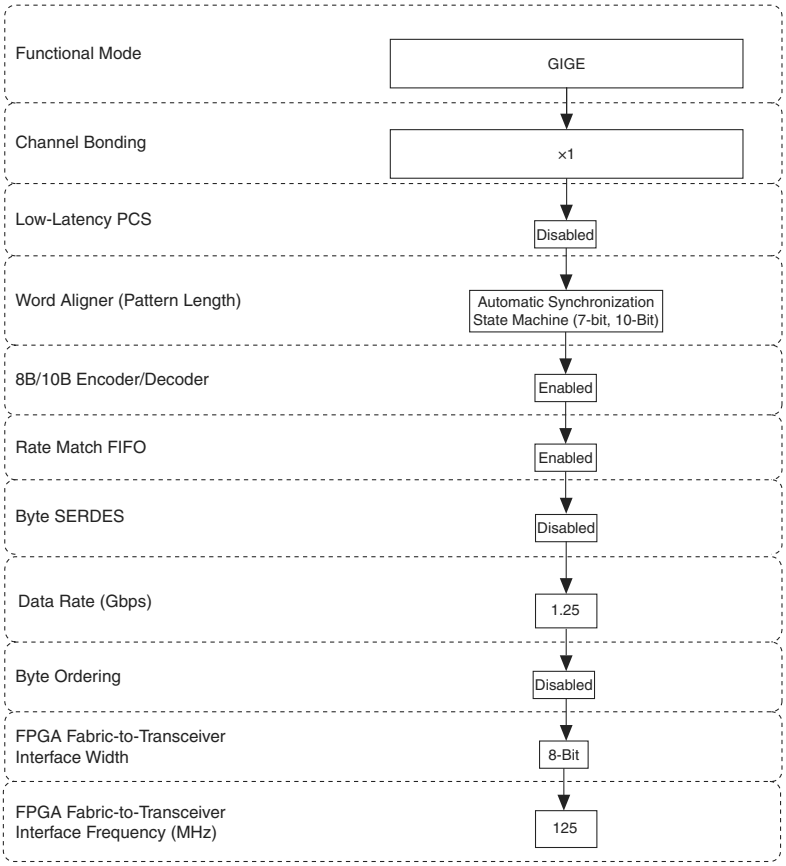
Figure 1–49. Transceiver Configuration in PIPE Mode



When configuring the transceiver into PIPE mode using ALTGX megafunction for PCIe implementation, the PHY-MAC, data link and transaction layers must be implemented in user logics. The PCIe hard IP block is bypassed in this configuration.

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

Figure 1–56. Transceiver Configuration in GIGE Mode



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

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

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.

All Supported Functional Modes Except the PCIe Functional Mode

This section describes reset sequences for transceiver channels in bonded and non-bonded configurations. Timing diagrams of some typical configurations are shown to facilitate proper reset sequence implementation. In these functional modes, you can set the receiver CDR either in automatic lock or manual lock mode.

In manual lock mode, the receiver CDR locks to the reference clock (lock-to-reference) or the incoming serial data (lock-to-data), depending on the logic levels on the `rx_locktorefclk` and `rx_locktodata` signals. With the receiver CDR in manual lock mode, you can either configure the transceiver channels in the Cyclone IV GX device in a non-bonded configuration or a bonded configuration. In a bonded configuration, for example in XAUI mode, four channels are bonded together.

Table 2–4 lists the lock-to-reference (LTR) and lock-to-data (LTD) controller lock modes for the `rx_locktorefclk` and `rx_locktodata` signals.

Table 2–4. Lock-To-Reference and Lock-To-Data Modes

rx_locktorefclk	rx_locktodata	LTR/LTD Controller Lock Mode
1	0	Manual, LTR Mode
—	1	Manual, LTD Mode
0	0	Automatic Lock Mode

Bonded Channel Configuration

In a bonded channel configuration, you can reset all the bonded channels simultaneously. Examples of bonded channel configurations are the XAUI, PCIe Gen1 ×2 and ×4, and Basic ×2 and ×4 functional modes. In Basic ×2 and ×4 functional mode, you can bond **Transmitter Only** channels together.

In XAUI mode, the receiver and transmitter channels are bonded. Each of the receiver channels in this mode has its own `rx_freqlocked` output status signals. You must consider the timing of these signals in the reset sequence.

Table 2–5 lists the reset and power-down sequences for bonded configurations under the stated functional modes.

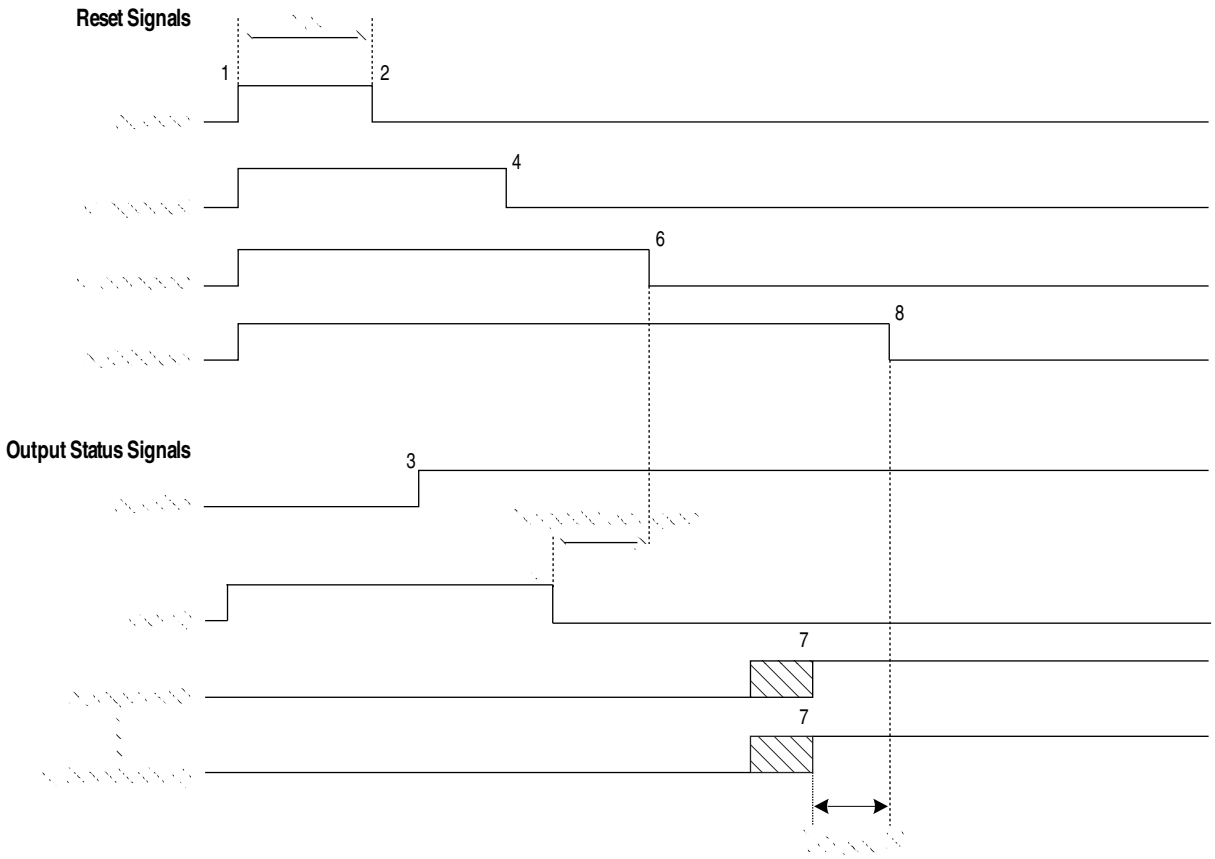
Table 2–5. Reset and Power-Down Sequences for Bonded Channel Configurations

Channel Set Up	Receiver CDR Mode	Refer to
Transmitter Only	Basic ×2 and ×4	"Transmitter Only Channel" on page 2–7
Receiver and Transmitter	Automatic lock mode for XAUI functional mode	"Receiver and Transmitter Channel—Receiver CDR in Automatic Lock Mode" on page 2–8
Receiver and Transmitter	Manual lock mode for XAUI functional mode	"Receiver and Transmitter Channel—Receiver CDR in Manual Lock Mode" on page 2–9

Receiver and Transmitter Channel—Receiver CDR in Automatic Lock Mode

This configuration contains both a transmitter and receiver channel. When the receiver CDR is in automatic lock mode, use the reset sequence shown in Figure 2–4.

Figure 2–4. Sample Reset Sequence for Bonded Configuration Receiver and Transmitter Channels—Receiver CDR in Automatic Lock Mode



Notes to Figure 2–4:

(1) The number of `rx_freqlocked[n]` signals depend on the number of channels configured. `n`=number of channels.

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

(3) The `busy` signal is asserted and deasserted only during initial power up when offset cancellation occurs. In subsequent reset sequences, the `busy` signal is asserted and deasserted only if there is a read or write operation to the ALTGX_RECONFIG megafunction.

As shown in Figure 2–4, perform the following reset procedure for the receiver CDR in automatic lock mode configuration:

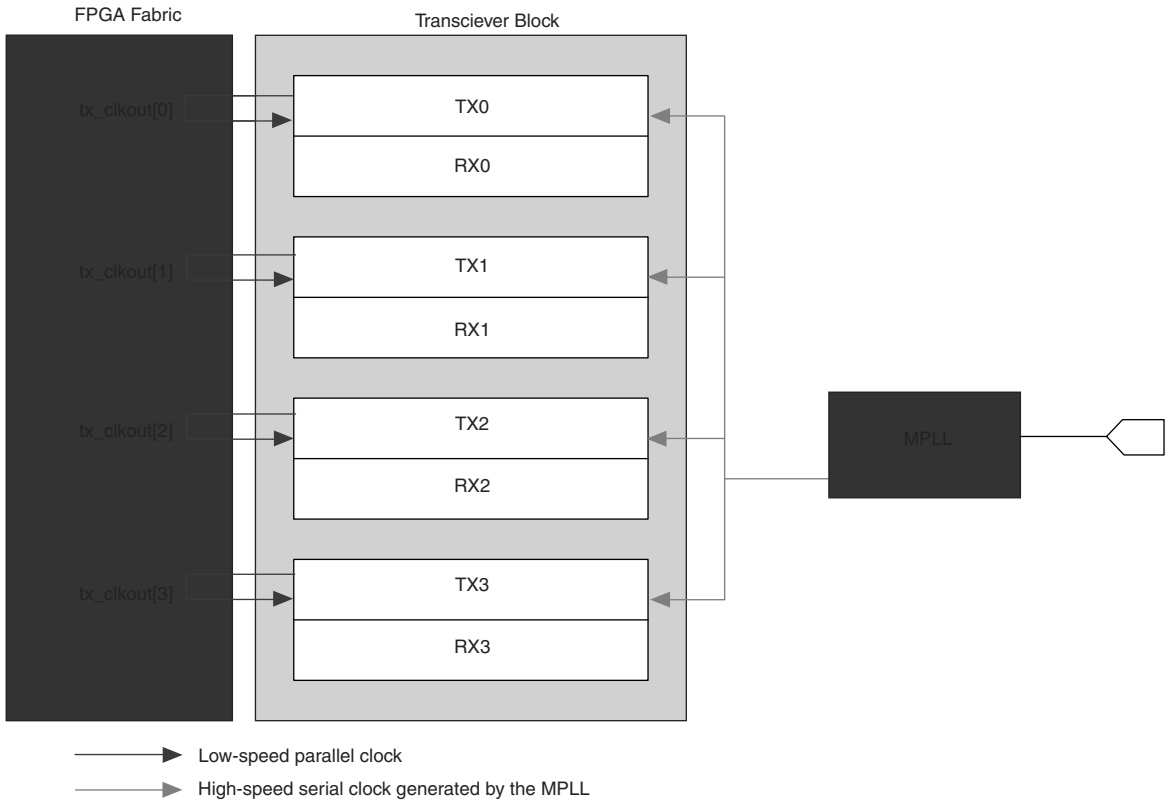
1. After power up, assert `pll_areset` for a minimum period of 1 µs (the time between markers 1 and 2).
2. Keep the `tx_digitalreset`, `rx_analogreset`, and `rx_digitalreset` signals asserted during this time period. After you deassert the `pll_areset` signal, the multipurpose PLL starts locking to the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

Option 2: Use the Respective Channel Transmitter Core Clocks

- Enable this option if you want the individual transmitter channel `tx_clkout` signals to provide the write clock to their respective Transmit Phase Compensation FIFOs.
- This option is typically enabled when each transceiver channel is reconfigured to a different functional mode using channel reconfiguration.

Figure 3–12 shows how each transmitter channel's `tx_clkout` signal provides a clock to the Transmit Phase Compensation FIFOs of the respective transceiver channels.

Figure 3–12. Option 2 for Transmitter Core Clocking (Channel Reconfiguration Mode)



Receiver core clocking refers to the clock that is used to read the parallel data from the Receiver Phase Compensation FIFO into the FPGA fabric. You can use one of the following clocks to read from the Receive Phase Compensation FIFO:

- `rx_coreclk`—you can use a clock of the same frequency as `rx_clkout` from the FPGA fabric to provide the read clock to the Receive Phase Compensation FIFO. If you use `rx_coreclk`, it overrides the `rx_clkout` options in the ALTGX MegaWizard Plug-In Manager.
- `rx_clkout`—the Quartus II software automatically routes `rx_clkout` to the FPGA fabric and back into the Receive Phase Compensation FIFO.

If you are reconfiguring the multipurpose PLL with a different M counter value, follow these steps:

1. During transceiver PLL reconfiguration, assert `tx_digitalreset`, `rx_digitalreset`, and `rx_analogreset` signals.
2. Perform PLL reconfiguration to update the multipurpose PLL with the PLL **.mif** files.
3. Perform channel reconfiguration and update the transceiver with the GXB reconfiguration **.mif** files. If you have multiple channel instantiations connected to the same multipurpose PLL, reconfigure each channel.
4. Deassert `tx_digitalreset` and `rx_analogreset` signals.
5. After the `rx_freqlocked` signal goes high, wait for at least 4 µs, and then deassert the `rx_digitalreset` signal.

Error Indication During Dynamic Reconfiguration

The ALTGX_RECONFIG MegaWizard Plug-In Manager provides an error status signal when you select the **Enable illegal mode checking** option or the **Enable self recovery** option in the **Error checks/data rate switch** screen. The conditions under which the error signal is asserted are:

- **Enable illegal mode checking option**—when you select this option, the dynamic reconfiguration controller checks whether an attempted operation falls under one of the conditions listed below. The dynamic reconfiguration controller detects these conditions within two `reconfig_clk` cycles, deasserts the `busy` signal, and asserts the error signal for two `reconfig_clk` cycles.
  - PMA controls, read operation—none of the output ports (`rx_eqctrl_out`, `rx_eqdcgain_out`, `tx_vodctrl_out`, and `tx_preemp_out`) are selected in the ALTGX_RECONFIG instance and the `read` signal is asserted.
  - PMA controls, write operation—none of the input ports (`rx_eqctrl`, `rx_eqdcgain`, `tx_vodctrl`, and `tx_preemp`) are selected in the ALTGX_RECONFIG instance and the `write_all` signal is asserted.
- **Channel reconfiguration** and **PMA reconfiguration mode** select - read operation option:
  - The `reconfig_mode_sel` input port is set to **3'b001** (Channel reconfiguration mode)
  - The `read` signal is asserted
- **Enable self recovery** option—when you select this option, the ALTGX_RECONFIG MegaWizard Plug-In Manager provides the error output port. The dynamic reconfiguration controller quits an operation if it did not complete within the expected number of clock cycles. After recovering from the illegal operation, the dynamic reconfiguration controller deasserts the `busy` signal and asserts the error output port for two `reconfig_clk` cycles.

☞ The error signal is not asserted when an illegal value is written to any of the PMA controls.

Table 1–3. Recommended Operating Conditions for Cyclone IV E Devices (1), (2) (Part 2 of 2)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{Diode}	Magnitude of DC current across PCI-clamp diode when enable	—	—	—	10	mA

Notes to Table 1–3:

(1) Cyclone IV E 1.0 V core voltage devices only support C8L, C9L, and I8L speed grades. Cyclone IV E 1.2 V core voltage devices only support C6, C7, C8, I7, and A7 speed grades.

(2) V_{CCIO} for all I/O banks must be powered up during device operation. All VCCA pins must be powered to 2.5 V (even when PLLs are not used) and must be powered up and powered down at the same time.

(3) V_{CC} must rise monotonically.

(4) V_{CCIO} powers all input buffers.

(5) The POR time for Standard POR ranges between 50 and 200 ms. Each individual power supply must reach the recommended operating range within 50 ms.

(6) The POR time for Fast POR ranges between 3 and 9 ms. Each individual power supply must reach the recommended operating range within 3 ms.

Table 1–4. Recommended Operating Conditions for Cyclone IV GX Devices (Part 1 of 2)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{CCINT} (3)	Core voltage, PCIe hard IP block, and transceiver PCS power supply	—	1.16	1.2	1.24	V
V_{CCA} (1), (3)	PLL analog power supply	—	2.375	2.5	2.625	V
V_{CCD_PLL} (2)	PLL digital power supply	—	1.16	1.2	1.24	V
V_{CCIO} (3), (4)	I/O banks power supply for 3.3-V operation	—	3.135	3.3	3.465	V
	I/O banks power supply for 3.0-V operation	—	2.85	3	3.15	V
	I/O banks power supply for 2.5-V operation	—	2.375	2.5	2.625	V
	I/O banks power supply for 1.8-V operation	—	1.71	1.8	1.89	V
	I/O banks power supply for 1.5-V operation	—	1.425	1.5	1.575	V
	I/O banks power supply for 1.2-V operation	—	1.14	1.2	1.26	V
V_{CC_CLKIN} (3), (5), (6)	Differential clock input pins power supply for 3.3-V operation	—	3.135	3.3	3.465	V
	Differential clock input pins power supply for 3.0-V operation	—	2.85	3	3.15	V
	Differential clock input pins power supply for 2.5-V operation	—	2.375	2.5	2.625	V
	Differential clock input pins power supply for 1.8-V operation	—	1.71	1.8	1.89	V
	Differential clock input pins power supply for 1.5-V operation	—	1.425	1.5	1.575	V
	Differential clock input pins power supply for 1.2-V operation	—	1.14	1.2	1.26	V
V_{CCH_GXB}	Transceiver output buffer power supply	—	2.375	2.5	2.625	V

Internal Weak Pull-Up and Weak Pull-Down Resistor

Table 1–12 lists the weak pull-up and pull-down resistor values for Cyclone IV devices.

Table 1–12. Internal Weak Pull-Up and Weak Pull-Down Resistor Values for Cyclone IV Devices *(1)*

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$R_{\_PU}$	Value of the I/O pin pull-up resistor before and during configuration, as well as user mode if you enable the programmable pull-up resistor option	V_{CCIO} = 3.3 V ± 5% *(2), (3)*	7	25	41	kΩ
		V_{CCIO} = 3.0 V ± 5% *(2), (3)*	7	28	47	kΩ
		V_{CCIO} = 2.5 V ± 5% *(2), (3)*	8	35	61	kΩ
		V_{CCIO} = 1.8 V ± 5% *(2), (3)*	10	57	108	kΩ
		V_{CCIO} = 1.5 V ± 5% *(2), (3)*	13	82	163	kΩ
		V_{CCIO} = 1.2 V ± 5% *(2), (3)*	19	143	351	kΩ
$R_{\_PD}$	Value of the I/O pin pull-down resistor before and during configuration	V_{CCIO} = 3.3 V ± 5% *(4)*	6	19	30	kΩ
		V_{CCIO} = 3.0 V ± 5% *(4)*	6	22	36	kΩ
		V_{CCIO} = 2.5 V ± 5% *(4)*	6	25	43	kΩ
		V_{CCIO} = 1.8 V ± 5% *(4)*	7	35	71	kΩ
		V_{CCIO} = 1.5 V ± 5% *(4)*	8	50	112	kΩ

Notes to Table 1–12:

(1) All I/O pins have an option to enable weak pull-up except the configuration, test, and JTAG pins. The weak pull-down feature is only available for JTAG TCK.

(2) Pin pull-up resistance values may be lower if an external source drives the pin higher than V_{CCIO}.

(3) $R_{\_PU} = (V_{CCIO} – V_I)/I_{R\_PU}$
Minimum condition: –40°C; $V_{CCIO} = V_{CC}$ + 5%, $V_I = V_{CC}$ + 5% – 50 mV;
Typical condition: 25°C; $V_{CCIO} = V_{CC}$, V_I = 0 V;
Maximum condition: 100°C; $V_{CCIO} = V_{CC}$ – 5%, V_I = 0 V; in which V_I refers to the input voltage at the I/O pin.

(4) $R_{\_PD} = V_I/I_{R\_PD}$
Minimum condition: –40°C; $V_{CCIO} = V_{CC}$ + 5%, V_I = 50 mV;
Typical condition: 25°C; $V_{CCIO} = V_{CC}$, $V_I = V_{CC}$ – 5%;
Maximum condition: 100°C; $V_{CCIO} = V_{CC}$ – 5%, $V_I = V_{CC}$ – 5%; in which V_I refers to the input voltage at the I/O pin.

Hot-Socketing

Table 1–13 lists the hot-socketing specifications for Cyclone IV devices.

Table 1–13. Hot-Socketing Specifications for Cyclone IV Devices

Symbol	Parameter	Maximum
$I_{IOPIN(DC)}$	DC current per I/O pin	300 µA
$I_{IOPIN(AC)}$	AC current per I/O pin	8 mA *(1)*
$I_{XCVRTX(DC)}$	DC current per transceiver TX pin	100 mA
$I_{XCVRRX(DC)}$	DC current per transceiver RX pin	50 mA

Note to Table 1–13:

(1) The I/O ramp rate is 10 ns or more. For ramp rates faster than 10 ns, |IIOPIN| = C dv/dt, in which C is the I/O pin capacitance and dv/dt is the slew rate.

☞ During hot-socketing, the I/O pin capacitance is less than 15 pF and the clock pin capacitance is less than 20 pF.

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

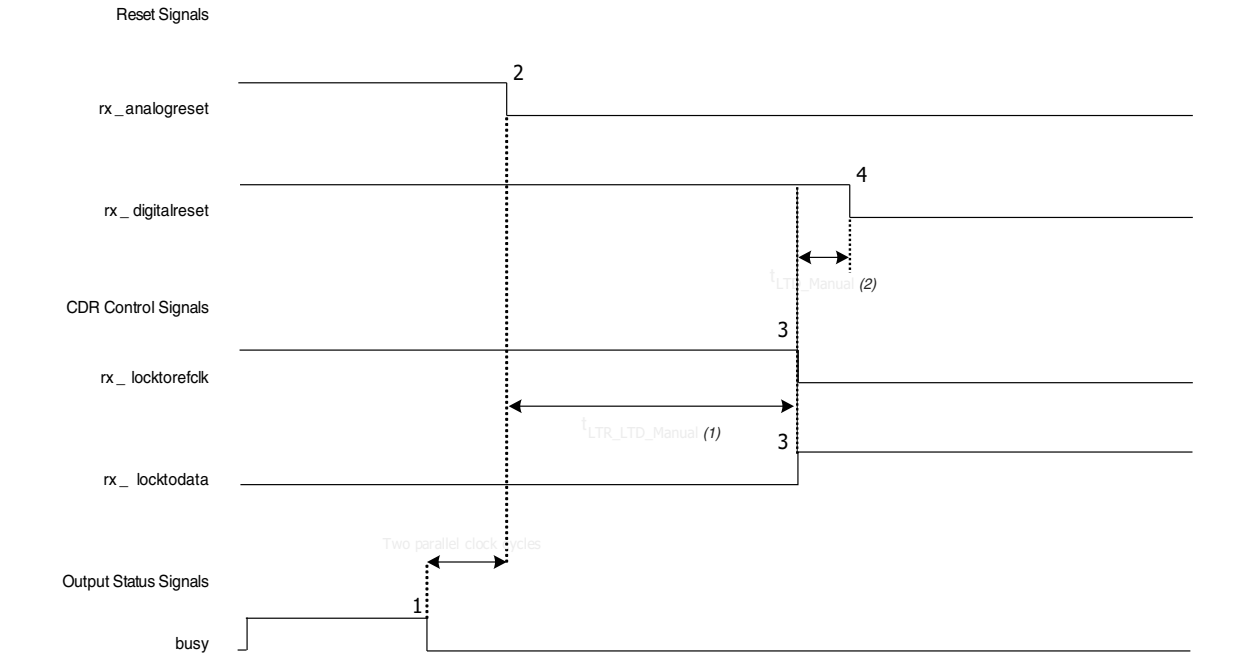


Figure 1–3 shows the lock time parameters in automatic mode.

Figure 1–3. Lock Time Parameters for Automatic Mode