



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	7155
Number of Logic Elements/Cells	114480
Total RAM Bits	3981312
Number of I/O	280
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/ep4ce115f23c8

Manual Clock Switchover	5-30
Guidelines	5-30
Programmable Bandwidth	5-32
Phase Shift Implementation	5-32
PLL Cascading	5-33
PLL Reconfiguration	5-34
PLL Reconfiguration Hardware Implementation	5-34
Post-Scale Counters (C0 to C4)	5-36
Scan Chain Description	5-37
Charge Pump and Loop Filter	5-38
Bypassing a PLL Counter	5-39
Dynamic Phase Shifting	5-39
Spread-Spectrum Clocking	5-41
PLL Specifications	5-41
Document Revision History	5-42

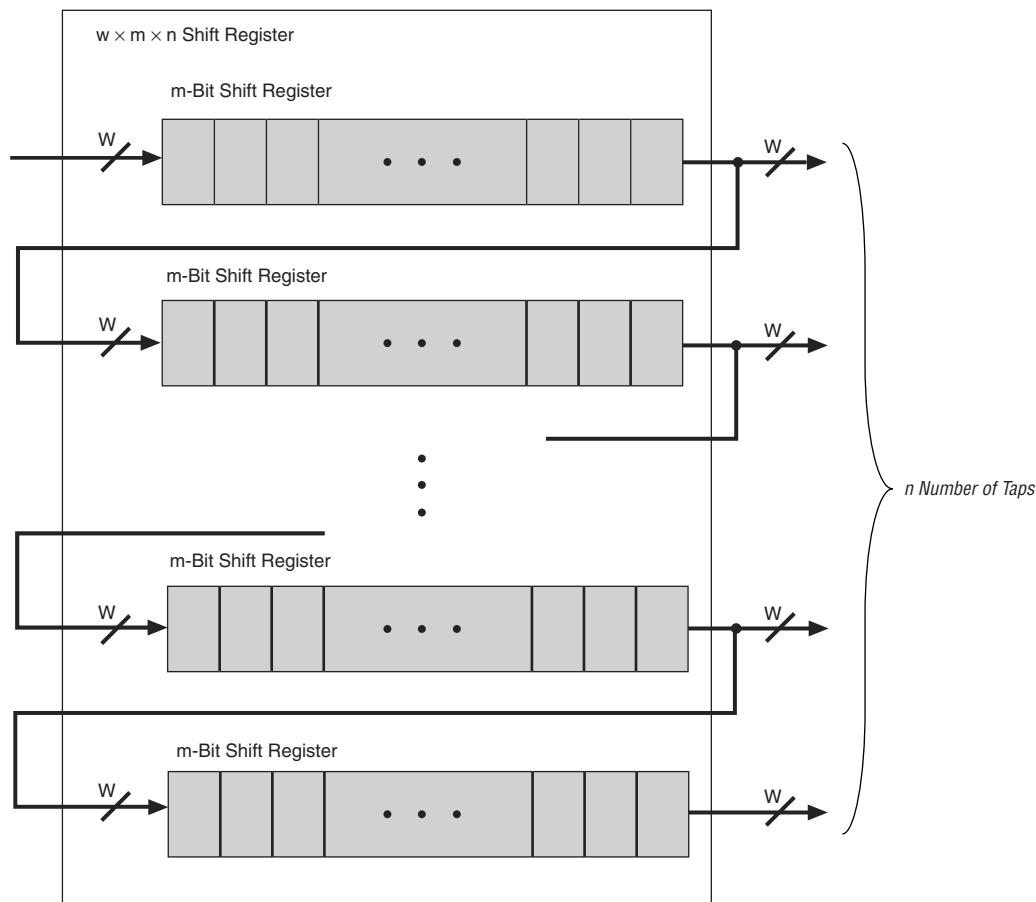
Section II. I/O Interfaces

Chapter 6. I/O Features in Cyclone IV Devices

Cyclone IV I/O Elements	6-2
I/O Element Features	6-3
Programmable Current Strength	6-3
Slew Rate Control	6-4
Open-Drain Output	6-4
Bus Hold	6-4
Programmable Pull-Up Resistor	6-5
Programmable Delay	6-5
PCI-Clamp Diode	6-6
OCT Support	6-6
On-Chip Series Termination with Calibration	6-8
On-Chip Series Termination Without Calibration	6-10
I/O Standards	6-11
Termination Scheme for I/O Standards	6-13
Voltage-Referenced I/O Standard Termination	6-14
Differential I/O Standard Termination	6-15
I/O Banks	6-16
High-Speed Differential Interfaces	6-22
External Memory Interfacing	6-23
Pad Placement and DC Guidelines	6-23
Pad Placement	6-23
DC Guidelines	6-23
Clock Pins Functionality	6-23
High-Speed I/O Interface	6-24
High-Speed I/O Standards Support	6-28
High Speed Serial Interface (HSSI) Input Reference Clock Support	6-28
LVDS I/O Standard Support in Cyclone IV Devices	6-29
Designing with LVDS	6-30
BLVDS I/O Standard Support in Cyclone IV Devices	6-30
Designing with BLVDS	6-31
RSDS, Mini-LVDS, and PPDS I/O Standard Support in Cyclone IV Devices	6-32
Designing with RSDS, Mini-LVDS, and PPDS	6-32
LVPECL I/O Support in Cyclone IV Devices	6-34
Differential SSTL I/O Standard Support in Cyclone IV Devices	6-35

Figure 3-12 shows the Cyclone IV devices M9K memory block in shift register mode.

Figure 3-12. Cyclone IV Devices Shift Register Mode Configuration



ROM Mode

Cyclone IV devices M9K memory blocks support ROM mode. A `.mif` initializes the ROM contents of these blocks. The address lines of the ROM are registered. The outputs can be registered or unregistered. The ROM read operation is identical to the read operation in the single-port RAM configuration.

FIFO Buffer Mode

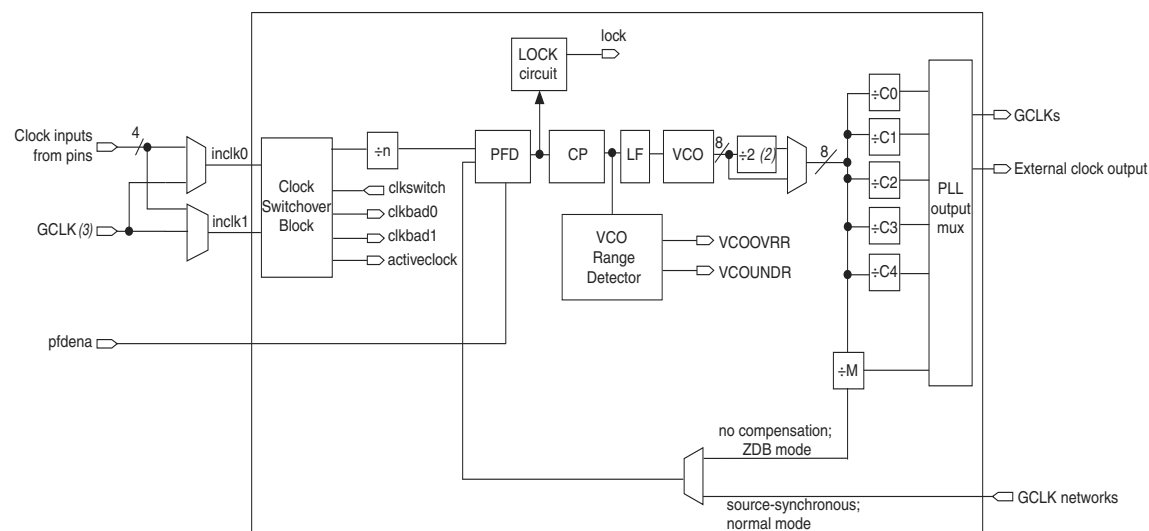
Cyclone IV devices M9K memory blocks support single-clock or dual-clock FIFO buffers. Dual clock FIFO buffers are useful when transferring data from one clock domain to another clock domain. Cyclone IV devices M9K memory blocks do not support simultaneous read and write from an empty FIFO buffer.



For more information about FIFO buffers, refer to the *Single- and Dual-Clock FIFO Megafunction User Guide*.


Figure 5-10 shows a simplified block diagram of the major components of the PLL of Cyclone IV E devices.

Figure 5-10. Cyclone IV E PLL Block Diagram ⁽¹⁾



Notes to Figure 5-10:

- (1) Each clock source can come from any of the four clock pins located on the same side of the device as the PLL.
- (2) This is the VCO post-scale counter K.
- (3) This input port is fed by a pin-driven dedicated GCLK, or through a clock control block if the clock control block is fed by an output from another PLL or a pin-driven dedicated GCLK. An internally generated global signal cannot drive the PLL.

 The VCO post-scale counter K is used to divide the supported VCO range by two. The VCO frequency reported by the Quartus II software in the PLL summary section of the compilation report takes into consideration the VCO post-scale counter value. Therefore, if the VCO post-scale counter has a value of 2, the frequency reported is lower than the f_{VCO} specification specified in the *Cyclone IV Device Datasheet* chapter.

External Clock Outputs

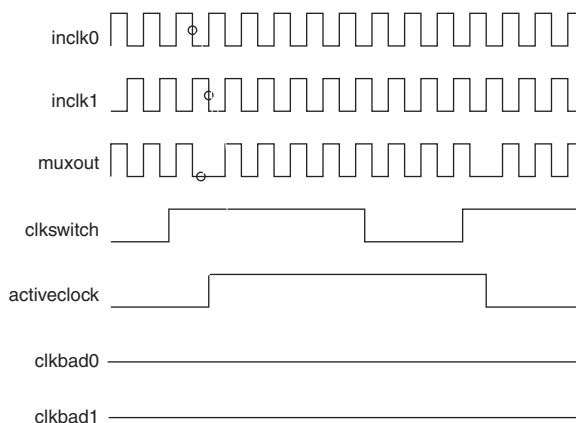
Each PLL of Cyclone IV devices supports one single-ended clock output or one differential clock output. Only the C0 output counter can feed the dedicated external clock outputs, as shown in Figure 5-11, without going through the GCLK. Other output counters can feed other I/O pins through the GCLK.

In this mode, the `activeclock` signal mirrors the `clkswitch` signal. As both blocks are still functional during the manual switch, neither `clkbad` signals go high. Because the switchover circuit is positive edge-sensitive, the falling edge of the `clkswitch` signal does not cause the circuit to switch back from `inclk1` to `inclk0`. When the `clkswitch` signal goes high again, the process repeats. The `clkswitch` signal and the automatic switch only works depending on the availability of the clock that is switched to. If the clock is unavailable, the state machine waits until the clock is available.



When `CLKSWITCH = 1`, it overrides the automatic switch-over function. As long as `clkswitch` signal is high, further switch-over action is blocked.

Figure 5–19. Clock Switchover Using the `clkswitch` Control ⁽¹⁾



Note to Figure 5–19:

(1) Both `inclk0` and `inclk1` must be running when the `clkswitch` signal goes high to start a manual clock switchover event.

Manual Clock Switchover

PLLs of Cyclone IV devices support manual switchover, in which the `clkswitch` signal controls whether `inclk0` or `inclk1` is the input clock to the PLL. The characteristics of a manual switchover are similar to the manual override feature in an automatic clock switchover, in which the switchover circuit is edge-sensitive. When the `clkswitch` signal goes high, the switchover sequence starts. The falling edge of the `clkswitch` signal does not cause the circuit to switch back to the previous input clock.



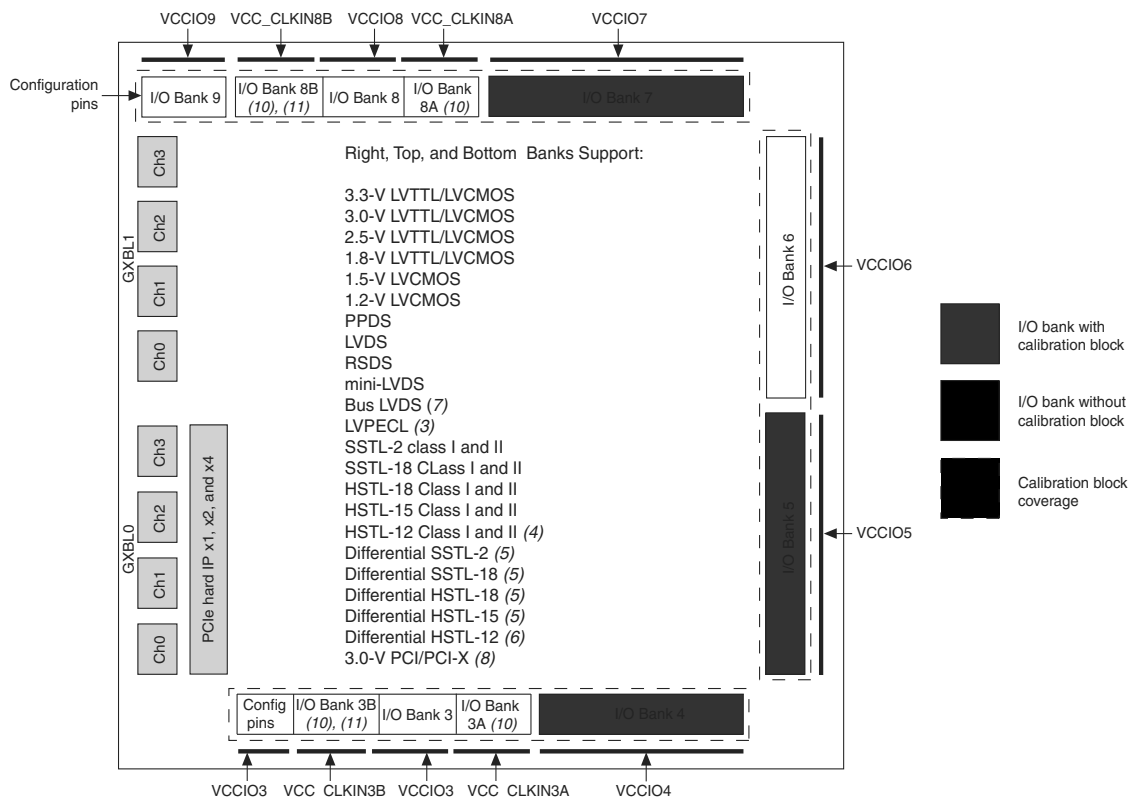
For more information about PLL software support in the Quartus II software, refer to the *ALTPLL Megafunction User Guide*.

Guidelines

Use the following guidelines to design with clock switchover in PLLs:

- Clock loss detection and automatic clock switchover require the `inclk0` and `inclk1` frequencies be within 20% of each other. Failing to meet this requirement causes the `clkbad0` and `clkbad1` signals to function improperly.

Figure 6-11. Cyclone IV GX I/O Banks for EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 (1), (2), (9)



Notes to Figure 6-11:

- (1) This is a top view of the silicon die. For exact pin locations, refer to the pin list and the Quartus II software.
- (2) True differential (PPDS, LVDS, mini-LVDS, and RSQS I/O standards) outputs are supported in row I/O banks 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 4, 7, and 8.
- (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. PLL output clock pins do not support Class II interface type of differential SSTL-18, HSTL-18, HSTL-15, and HSTL-12 I/O standards.
- (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 4, 7, and 8.
- (7) BLVDS output uses two single-ended outputs with the second output programmed as inverted. BLVDS input uses the LVDS input buffer.
- (8) The PCI-X I/O standard does not meet the IV curve requirement at the linear region.
- (9) The OCT block is located in the shaded banks 4, 5, and 7.
- (10) The dedicated clock input I/O banks 3A, 3B, 8A, and 8B can be used either for HSSI input reference clock pins or clock input pins.
- (11) Single-ended clock input support is available for dedicated clock input I/O banks 3B and 8B.

The simpler method for multi-device AP configuration is the byte-wide multi-device AP configuration. In the byte-wide multi-device AP configuration, the LSB of the DATA[7..0] pin from the flash and master device (set to the AP configuration scheme) is connected to the slave devices set to the FPP configuration scheme, as shown in Figure 8–8.

[illegible]

- (1) Connect the pull-up resistors to the V_{CCIO} supply of the bank in which the pin resides.
- (2) Connect the pull-up resistor to the V_{CCIO} supply voltage of the I/O bank in which the nCE pin resides.
- (3) The $nCEO$ pin is left unconnected or used as a user I/O pin when it does not feed the nCE pin of another device.
- (4) The $MSEL$ pin settings vary for different configuration voltage standards and POR time. You must set the master device in AP mode and the slave devices in FPP mode. To connect $MSEL[3..0]$ for the master device in AP mode and the slave devices in FPP mode, refer to Table 8–5 on page 8–9. Connect the $MSEL$ pins directly to V_{CCA} or GND.
- (5) The AP configuration ignores the $WAIT$ signal during configuration mode. However, if you are accessing flash during user mode with user logic, you can optionally use the normal I/O to monitor the $WAIT$ signal from the Micron P30 or P33 flash.
- (6) Connect the repeater buffers between the Cyclone IV E master device and slave devices for $DATA[15..0]$ and $DCLK$. All I/O inputs must maintain a maximum AC voltage of 4.1 V. The output resistance of the repeater buffers must fit the maximum overshoot equation outlined in “Configuration and JTAG Pin I/O Requirements” on page 8–5.

The more efficient setup is one in which some of the slave devices are connected to the LSB of the DATA[7..0] and the remaining slave devices are connected to the MSB of the DATA[15..8]. In the word-wide multi-device AP configuration, the nCEO pin of the master device enables two separate daisy chains of slave devices, allowing both chains to be programmed concurrently, as shown in Figure 8–9.

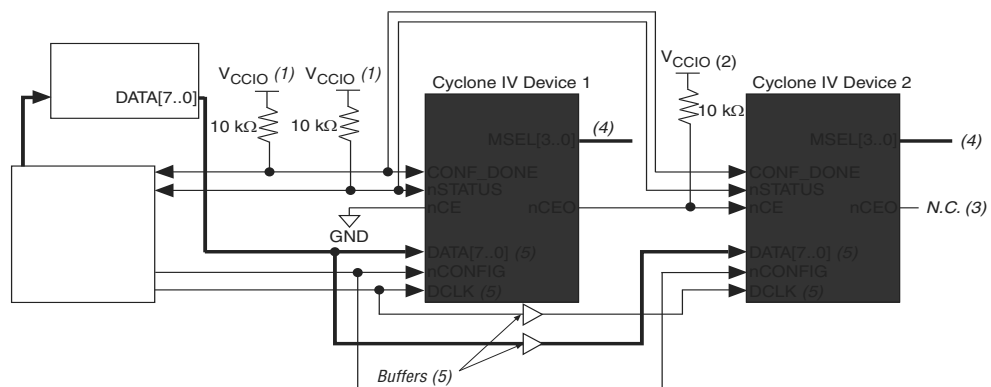
To ensure that DCLK and DATA[0] are not left floating at the end of the configuration, the MAX II device must drive them either high or low, whichever is convenient on your board. The DATA[0] pin is available as a user I/O pin after configuration. When you choose the FPP scheme in the Quartus II software, the DATA[0] pin is tri-stated by default in user mode and must be driven by the external host device. To change this default option in the Quartus II software, select the **Dual-Purpose Pins** tab of the **Device and Pin Options** dialog box.

The DCLK speed must be below the specified system frequency to ensure correct configuration. No maximum DCLK period exists, which means you can pause configuration by halting DCLK for an indefinite amount of time.

The external host device can also monitor the CONF_DONE and INIT_DONE pins to ensure successful configuration. The CONF_DONE pin must be monitored by the external device to detect errors and to determine when programming is complete. If all configuration data is sent, but CONF_DONE or INIT_DONE has not gone high, the external device must reconfigure the target device.

Figure 8-20 shows how to configure multiple devices with a MAX II device. This circuit is similar to the FPP configuration circuit for a single device, except the Cyclone IV devices are cascaded for multi-device configuration.

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



Notes to Figure 8-20:

- (1) The pull-up resistor must be connected to a supply that provides an acceptable input signal for all devices in the chain. V_{CC} must be high enough to meet the V_{IH} specification of the I/O on the device and the external host.
- (2) Connect the pull-up resistor to the V_{CCIO} supply voltage of the I/O bank in which the nCE pin resides.
- (3) The nCEO pin is left unconnected or used as a user I/O pin when it does not feed the nCE pin of another device.
- (4) The MSEL pin settings vary for different configuration voltage standards and POR time. To connect the MSEL pins, refer to Table 8-4 on page 8-8 and Table 8-5 on page 8-9. Connect the MSEL pins directly to V_{CCA} or GND.
- (5) All I/O inputs must maintain a maximum AC voltage of 4.1 V. DATA[7..0] and DCLK must fit the maximum overshoot outlined in Equation 8-1 on page 8-5.

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

ACTIVE_DISENGAGE

The ACTIVE_DISENGAGE instruction places the active configuration controller (AS and AP) into an idle state prior to JTAG programming. The two purposes of placing the active controller in an idle state are:

- To ensure that it is not trying to configure the device during JTAG programming
- To allow the controllers to properly recognize a successful JTAG programming that results in the device reaching user mode

The ACTIVE_DISENGAGE instruction is required before JTAG programming regardless of the current state of the Cyclone IV device if the MSEL pins are set to an AS or AP configuration scheme. If the ACTIVE_DISENGAGE instruction is issued during a passive configuration scheme (PS or FPP), it has no effect on the Cyclone IV device. Similarly, the CONFIG_IO instruction is issued after an ACTIVE_DISENGAGE instruction, but is no longer required to properly halt configuration. Table 8-17 lists the required, recommended, and optional instructions for each configuration mode. The ordering of the required instructions is a hard requirement and must be met to ensure functionality.

Table 8-17. JTAG Programming Instruction Flows ⁽¹⁾

JTAG Instruction	Configuration Scheme and Current State of the Cyclone IV Device											
	Prior to User Mode (Interrupting Configuration)				User Mode				Power Up			
	PS	FPP	AS	AP	PS	FPP	AS	AP	PS	FPP	AS	AP
ACTIVE_DISENGAGE	O	O	R	R	O	O	O	R	O	O	R	R
CONFIG_IO	Rc	Rc	O	O	O	O	O	O	NA	NA	NA	NA
Other JTAG instructions	O	O	O	O	O	O	O	O	O	O	O	O
JTAG_PROGRAM	R	R	R	R	R	R	R	R	R	R	R	R
CHECK_STATUS	Rc	Rc	Rc	Rc	Rc	Rc	Rc	Rc	Rc	Rc	Rc	Rc
JTAG_STARTUP	R	R	R	R	R	R	R	R	R	R	R	R
JTAG TAP Reset/other instruction	R	R	R	R	R	R	R	R	R	R	R	R

Note to Table 8-17:

- (1) "R" indicates that the instruction must be executed before the next instruction, "O" indicates the optional instruction, "Rc" indicates the recommended instruction, and "NA" indicates that the instruction is not allowed in this mode.

In the AS or AP configuration scheme, the ACTIVE_DISENGAGE instruction puts the active configuration controller into idle state. If a successful JTAG programming is executed, the active controller is automatically re-engaged after user mode is reached through JTAG programming. This causes the active controller to transition to their respective user mode states.

If JTAG programming fails to get the Cyclone IV device to enter user mode and re-engage active programming, there are available methods to achieve this:


- In AS configuration scheme, you can re-engage the AS controller by moving the JTAG TAP controller to the reset state or by issuing the ACTIVE_ENGAGE instruction.

- In AP configuration scheme, the only way to re-engage the AP controller is to issue the `ACTIVE_ENGAGE` instruction. In this case, asserting the `nCONFIG` pin does not re-engage either active controller.

ACTIVE_ENGAGE

The `ACTIVE_ENGAGE` instruction allows you to re-engage a disengaged active controller. You can issue this instruction any time during configuration or user mode to re-engage an already disengaged active controller, as well as trigger reconfiguration of the Cyclone IV device in the active configuration scheme.

The `ACTIVE_ENGAGE` instruction functions as the `PULSE_NCONFIG` instruction when the device is in the PS or FPP configuration schemes. The `nCONFIG` pin is disabled when the `ACTIVE_ENGAGE` instruction is issued.

 Altera does not recommend using the `ACTIVE_ENGAGE` instruction, but it is provided as a fail-safe instruction for re-engaging the active configuration controller (AS and AP).


Overriding the Internal Oscillator

This feature allows you to override the internal oscillator during the active configuration scheme. The AS and AP configuration controllers use the internal oscillator as the clock source. You can change the clock source to `CLKUSR` through the JTAG instruction.

The `EN_ACTIVE_CLK` and `DIS_ACTIVE_CLK` JTAG instructions toggle on or off whether or not the active clock is sourced from the `CLKUSR` pin or the internal configuration oscillator. To source the active clock from the `CLKUSR` pin, issue the `EN_ACTIVE_CLK` instruction. This causes the `CLKUSR` pin to become the active clock source. When using the `EN_ACTIVE_CLK` instruction, you must enable the internal oscillator for the clock change to occur. By default, the configuration oscillator is disabled after configuration and initialization is complete as well as the device has entered user mode.

However, the internal oscillator is enabled in user mode by any of the following conditions:

- A reconfiguration event (for example, driving the `nCONFIG` pin to go low)
- Remote update is enabled
- Error detection is enabled

 When using the `EN_ACTIVE_CLK` and `DIS_ACTIVE_CLK` JTAG instructions to override the internal oscillator, you must clock the `CLKUSR` pin at two times the expected `DCLK` frequency. The `CLKUSR` pin allows a maximum frequency of 40 MHz (40 MHz `DCLK`).

Normally, a test instrument uses the `CLKUSR` pin when it wants to drive its own clock to control the AS state machine.

To revert the clock source back to the configuration oscillator, issue the `DIS_ACTIVE_CLK` instruction. After you issue the `DIS_ACTIVE_CLK` instruction, you must continue to clock the `CLKUSR` pin for 10 clock cycles. Otherwise, even toggling the `nCONFIG` pin does not revert the clock source and reconfiguration does not occur. A POR reverts the clock source back to the configuration oscillator. Toggling the `nCONFIG` pin or driving the JTAG state machine to reset state does not revert the clock source.

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

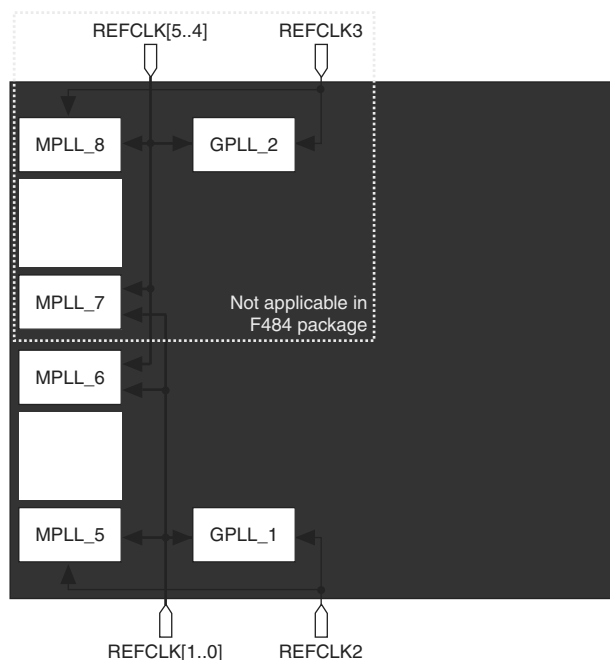
CRC_ERROR Pin Type	Description
I/O, Output (open-drain)	Active high signal indicates that the error detection circuit has detected errors in the configuration SRAM bits. This pin is optional and is used when the CRC error detection circuit is enabled in the Quartus II software from the Error Detection 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.

© 2015 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.



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.

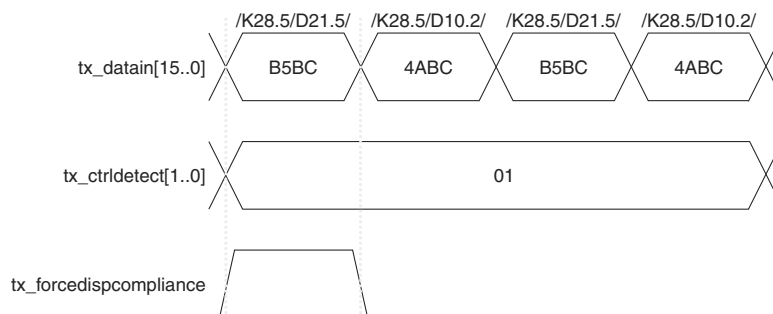
The input reference clocks reside in banks 3A, 3B, 8A, and 8B have dedicated $V_{CC_CLKIN3A}$, $V_{CC_CLKIN3B}$, $V_{CC_CLKIN8A}$, and $V_{CC_CLKIN8B}$ power supplies separately in their respective I/O banks to avoid the different power level requirements in the same bank for general purpose I/Os (GPIOs). 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

The compliance pattern is a repeating sequence of the four code groups: /K28.5/; /D21.5/; /K28.5/; /D10.2/. Figure 1-53 shows the compliance pattern transmission where the tx_forcedispcompliance port must be asserted in the same parallel clock cycle as /K28.5/D21.5/ of the compliance pattern on tx_datain[15..0] port.

Figure 1-53. Compliance Pattern Transmission Support in PCI Express (PIPE) Mode



Reset Requirement

Cyclone IV GX devices meet the PCIe reset time requirement from device power up to the link active state with the configuration schemes listed in Table 1-17.

Table 1-18. Electrical Idle Inference Conditions

Device	Configuration Scheme	Configuration Time (ms)
EP4CGX15	Passive serial (PS)	51
EP4CGX22	PS	92
EP4CGX30 ⁽¹⁾	PS	92
EP4CGX50	Fast passive parallel (FPP)	41
EP4CGX75	FPP	41
EP4CGX110	FPP	70
EP4CGX150	FPP	70

Note to Table 1-18:


(1) EP4CGX30 device in F484 package fulfills the PCIe reset time requirement using FPP configuration scheme with configuration time of 41 ms.

GIGE Mode

GIGE mode provides the transceiver channel datapath configuration for GbE (specifically the 1000 Base-X physical layer device (PHY) standard) protocol implementation. The Cyclone IV GX transceiver provides the PMA and the following PCS functions as defined in the IEEE 802.3 specification for 1000 Base-X PHY:

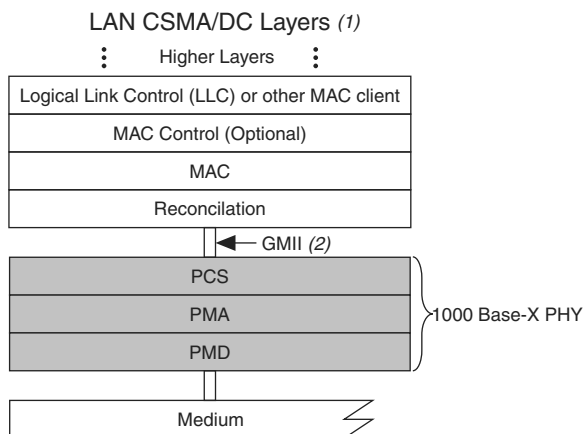
- 8B/10B encoding and decoding
- synchronization

If you enabled the auto-negotiation state machine in the FPGA core with the rate match FIFO, refer to “Clock Frequency Compensation” on page 1-63.

 Cyclone IV GX transceivers do not have built-in support for some PCS functions such as auto-negotiation state machine, collision-detect, and carrier-sense. If required, you must implement these functions in a user logic or external circuits.

The 1000 Base-X PHY is defined by IEEE 802.3 standard as an intermediate or transition layer that interfaces various physical media with the media access control (MAC) in a GbE system. The 1000 Base-X PHY, which has a physical interface data rate of 1.25 Gbps consists of the PCS, PMA, and physical media dependent (PMD) layers. Figure 1-54 shows the 1000 Base-X PHY in LAN layers.

Figure 1-54. 1000 Base-X PHY in a GbE OSI Reference Model

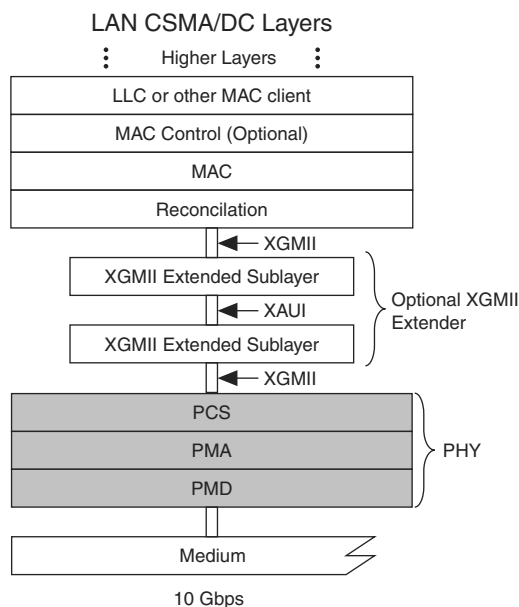


Notes to Figure 1-54:

- (1) CSMA/CD = Carrier-Sense Multiple Access with Collision Detection
- (2) GMII = gigabit medium independent interface

converted within the XGMII extender sublayer into an 8B/10B encoded data stream. Each data stream is then transmitted across a single differential pair running at 3.125 Gbps. At the XAUI receiver, the incoming data is decoded and mapped back to the 32-bit XGMII format. This provides a transparent extension of the physical reach of the XGMII and also reduces the interface pin count.

Figure 1-62. XAUI in 10 Gbps LAN Layers



XAUI functions as a self-managed interface because code group synchronization, channel deskew, and clock domain decoupling is handled with no upper layer support requirements. This functionality is based on the PCS code groups that are used during the inter-packet gap time and idle periods.

Method 3: Writing Different Control Signals for all the Transceiver Channels at the Same Time

If you disable the **Use the same control signal for all the channels** option, the PMA control ports for a write transaction are separate for each channel. If you disable this option, the width of the PMA control ports are fixed as follows:

PMA Control Ports Used in a Write Transaction

- `tx_vodctrl` is 3 bits per channel
- `tx_preemp` are 5 bits per channel
- `rx_eqdcgain` is 2 bits per channel
- `rx_eqctrl` is 4 bits per channel

For example, if you have two channels, the `tx_vodctrl` is 6 bits wide (`tx_vodctrl [2:0]` corresponds to channel 1 and `tx_vodctrl [5:3]` corresponds to channel 2).

PMA Control Ports Used in a Read Transaction

The width of the PMA control ports for a read transaction are always separate for each channel as explained in “Method 2: Writing the Same Control Signals to Control All the Transceiver Channels” on page 3-16.

Write Transaction

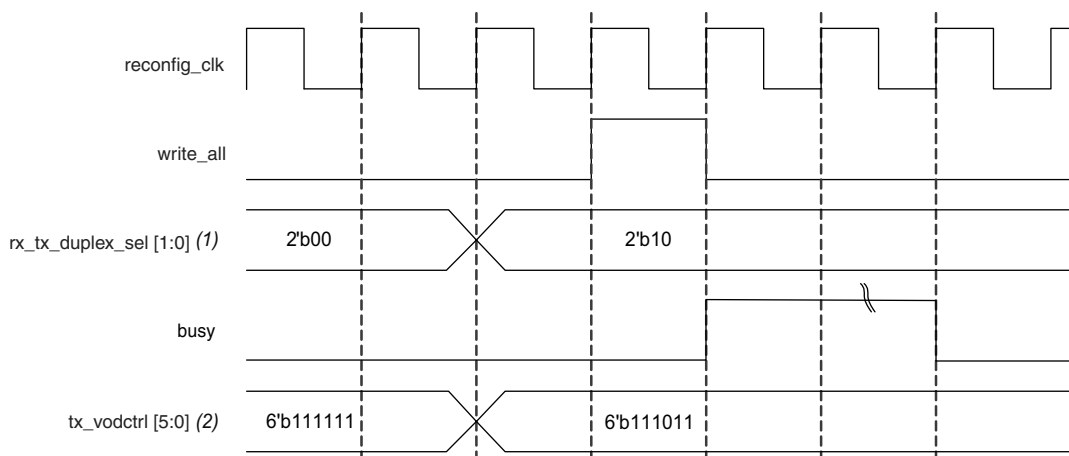
Because the PMA controls of all the channels are written, if you want to reconfigure a specific channel connected to the `ALTGX_RECONFIG` instance, set the new value at the corresponding PMA control port of the channel under consideration and retain the previously stored values in the other active channels with a read transaction prior to this write transaction.

For example, if the number of channels controlled by the `ALTGX_RECONFIG` instance is two, the `tx_vodctrl` signal in this case would be 6 bits wide. The `tx_vodctrl [2:0]` signal corresponds to channel 1 and the `tx_vodctrl [5:3]` signal corresponds to channel 2.

- To dynamically reconfigure the PMA controls of only channel 2 with a new value, first perform a read transaction to retrieve the existing PMA control values from `tx_vodctrl_out [5:0]`. Use the `tx_vodctrl_out [2:0]` value for `tx_vodctrl [2:0]` to write in channel 1. By doing so, channel 1 is overwritten with the same value.
- Perform a write transaction. This ensures that the new values are written only to channel 2 while channel 1 remains unchanged.

Figure 3-8 shows a write transaction waveform with the **Use the same control signal for all the channels** option disabled.

Figure 3-8. Write Transaction Waveform—Use the same control signal for all the channels Option Disabled



Notes to Figure 3-8:

- (1) In this waveform example, you want to write to only the transmitter portion of the channel.
- (2) In this waveform example, the number of channels controlled by the dynamic reconfiguration controller (the ALTGX_RECONFIG instance) is two and that the `tx_vodctrl` control port is enabled.



Simultaneous write and read transactions are not allowed.

Read Transaction

The read transaction in Method 3 is identical to that in Method 2. Refer to “Read Transaction” on page 3-18.



This is the slowest method. You have to write all the PMA settings for all channels even if you may only be changing one parameter on the channel. Altera recommends using the `logical_channel_address` method for time-critical applications.

For each method, you can additionally reconfigure the PMA setting of both transmitter and receiver portion, transmitter portion only, or receiver portion only of the transceiver channel. For more information, refer to “Dynamic Reconfiguration Controller Port List” on page 3-4. You can enable the `rx_tx_duplex_sel` port by selecting the **Use 'rx_tx_duplex_sel' port to enable RX only, TX only or duplex reconfiguration** option on the **Error checks** tab of the ALTGX_RECONFIG MegaWizard Plug-In Manager.

Figure 3-9 shows the ALTGX_RECONFIG connection to the ALTGX instances when set in analog reconfiguration mode. For the port information, refer to the “Dynamic Reconfiguration Controller Port List” on page 3-4.

Table 3-5. rx_dataoutfull[31..0] FPGA Fabric-Transceiver Channel Interface Signal Descriptions (Part 3 of 3)

FPGA Fabric-Transceiver Channel Interface Description	Receive Signal Description (Based on Cyclone IV GX Supported FPGA Fabric-Transceiver Channel Interface Widths)
20-bit FPGA fabric-Transceiver Channel Interface with PCS-PMA set to 10 bits	Two 10-bit Data (rx_dataout) rx_dataoutfull[9:0] - rx_dataout (LSByte) and rx_dataoutfull[25:16] - rx_dataout (MSByte)
	Two Receiver Sync Status Bits rx_dataoutfull[10] - rx_syncstatus (LSB) and rx_dataoutfull[26] - rx_syncstatus (MSB)
	rx_dataoutfull[11] and rx_dataoutfull[27]: 8B/10B disparity error indicator (rx_disperr)
	Two Receiver Pattern Detect Bits rx_dataoutfull[12] - rx_patterndetect (LSB) and rx_dataoutfull[28] - rx_patterndetect (MSB)
	rx_dataoutfull[13] and rx_dataoutfull[29]: Rate Match FIFO deletion status indicator (rx_rmfifodatadeleted) in non-PCI Express (PIPE) functional modes
	rx_dataoutfull[14] and rx_dataoutfull[30]: Rate Match FIFO insertion status indicator (rx_rmfifodatainserted) in non-PCI Express (PIPE) functional modes
	rx_dataoutfull[15] and rx_dataoutfull[31]: 8B/10B running disparity indicator (rx_runningdisp)

Data Rate Reconfiguration Mode Using RX Local Divider

The RX local divider resides in the RX PMA block for every channels. This is a hardware feature where a /2 divider is available in each of the receiver channel for the supported device. You can use this RX local divider to reconfigure the data rate at the receiver channel. This can be used for protocols such as SDI that has data rates in divisions of 2.

By using this RX local divider, you can support two different data rates without using additional transceiver PLLs. This dynamic reconfiguration mode is available only for the receiver and not applicable to the transmitter. This reconfiguration mode using the RX local divider (/2) is only supported and available in EP4CGX30 (F484 package), EP4CGX50, and EP4CGX75 devices.



For more information about this RX local divider, refer to the *Cyclone IV GX Transceiver Architecture* chapter.

The OCT resistance may vary with the variation of temperature and voltage after calibration at device power-up. Use Table 1-10 and Equation 1-1 to determine the final OCT resistance considering the variations after calibration at device power-up. Table 1-10 lists the change percentage of the OCT resistance with voltage and temperature.

Table 1-10. OCT Variation After Calibration at Device Power-Up for Cyclone IV Devices ⁽¹⁾

Nominal Voltage	dR/dT (%/°C)	dR/dV (%/mV)
3.0	0.262	-0.026
2.5	0.234	-0.039
1.8	0.219	-0.086
1.5	0.199	-0.136
1.2	0.161	-0.288

Note to Table 1-10:

(1) This specification is not applicable to EP4CGX15, EP4CGX22, and EP4CGX30 devices.

Equation 1-1. Final OCT Resistance ^{(1), (2), (3), (4), (5), (6)}

$$\Delta R_V = (V_2 - V_1) \times 1000 \times dR/dV \text{ — (7)}$$

$$\Delta R_T = (T_2 - T_1) \times dR/dT \text{ — (8)}$$

$$\text{For } \Delta R_x < 0; MF_x = 1 / (|\Delta R_x|/100 + 1) \text{ — (9)}$$

$$\text{For } \Delta R_x > 0; MF_x = \Delta R_x/100 + 1 \text{ — (10)}$$

$$MF = MF_V \times MF_T \text{ — (11)}$$

$$R_{\text{final}} = R_{\text{initial}} \times MF \text{ — (12)}$$

Notes to Equation 1-1:

- (1) T_2 is the final temperature.
- (2) T_1 is the initial temperature.
- (3) MF is multiplication factor.
- (4) R_{final} is final resistance.
- (5) R_{initial} is initial resistance.
- (6) Subscript x refers to both V and T .
- (7) ΔR_V is a variation of resistance with voltage.
- (8) ΔR_T is a variation of resistance with temperature.
- (9) dR/dT is the change percentage of resistance with temperature after calibration at device power-up.
- (10) dR/dV is the change percentage of resistance with voltage after calibration at device power-up.
- (11) V_2 is final voltage.
- (12) V_1 is the initial voltage.

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

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

Notes to Table 1–25:

- (1) This table is applicable for general purpose PLLs and multipurpose PLLs.
- (2) You must connect V_{CCD_PLL} to V_{CCINT} through the decoupling capacitor and ferrite bead.
- (3) This parameter is limited in the Quartus II software by the I/O maximum frequency. The maximum I/O frequency is different for each I/O standard.
- (4) The V_{CO} frequency reported by the Quartus II software in the PLL Summary section of the compilation report takes into consideration the V_{CO} post-scale counter K value. Therefore, if the counter K has a value of 2, the frequency reported can be lower than the f_{VCO} specification.
- (5) A high input jitter directly affects the PLL output jitter. To have low PLL output clock jitter, you must provide a clean clock source that is less than 200 ps.
- (6) Peak-to-peak jitter with a probability level of 10^{-12} (14 sigma, 99.9999999974404% confidence level). The output jitter specification applies to the intrinsic jitter of the PLL when an input jitter of 30 ps is applied.
- (7) With 100-MHz scanclk frequency.
- (8) The cascaded PLLs specification is applicable only with the following conditions:
 - Upstream PLL— $0.59 \text{ MHz} \leq \text{Upstream PLL bandwidth} < 1 \text{ MHz}$
 - Downstream PLL—Downstream PLL bandwidth $> 2 \text{ MHz}$
- (9) PLL cascading is not supported for transceiver applications.