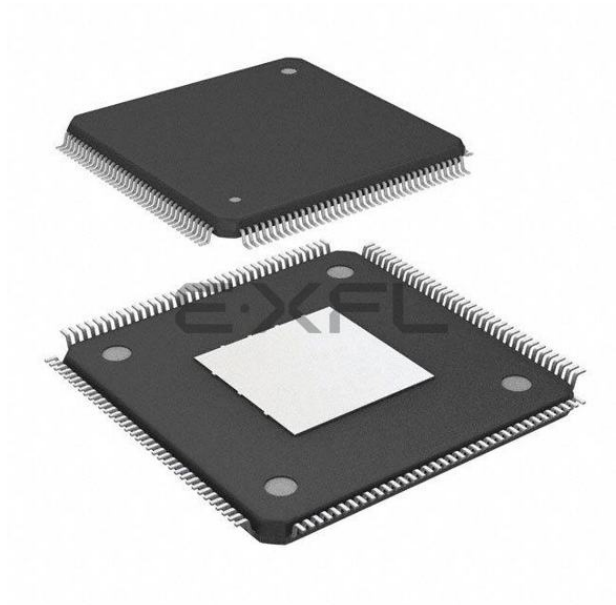


Welcome to E-XFL.COM

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	963
Number of Logic Elements/Cells	15408
Total RAM Bits	516096
Number of I/O	81
Number of Gates	-
Voltage - Supply	0.97V ~ 1.03V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	144-LQFP Exposed Pad
Supplier Device Package	144-EQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce15e22i8ln

Contents

Document Revision History

Table 1–10 lists the revision history for this chapter.

Table 1–10. Document Revision History

Date	Version	Changes
March 2016	2.0	■ Updated Table 1–4 and Table 1–5 to remove support for the N148 package. ■ Updated Figure 1–2 to remove support for the N148 package.
April 2014	1.9	Updated "Packaging Ordering Information for the Cyclone IV E Device".
May 2013	1.8	Updated Table 1–3, Table 1–6 and Figure 1–3 to add new device options and packages.
February 2013	1.7	Updated Table 1–3, Table 1–6 and Figure 1–3 to add new device options and packages.
October 2012	1.6	Updated Table 1–3 and Table 1–4.
November 2011	1.5	■ Updated "Cyclone IV Device Family Features" section. ■ Updated Figure 1–2 and Figure 1–3.
December 2010	1.4	■ Updated for the Quartus II software version 10.1 release. ■ Added Cyclone IV E new device package information. ■ Updated Table 1–1, Table 1–2, Table 1–3, Table 1–5, and Table 1–6. ■ Updated Figure 1–3. ■ Minor text edits.
July 2010	1.3	Updated Table 1–2 to include F484 package information.
March 2010	1.2	■ Updated Table 1–3 and Table 1–6. ■ Updated Figure 1–3. ■ Minor text edits.
February 2010	1.1	■ Added Cyclone IV E devices in Table 1–1, Table 1–3, and Table 1–6 for the Quartus II software version 9.1 SP1 release. ■ Added the "Cyclone IV Device Family Speed Grades" and "Configuration" sections. ■ Added Figure 1–3 to include Cyclone IV E Device Packaging Ordering Information. ■ Updated Table 1–2, Table 1–4, and Table 1–5 for Cyclone IV GX devices. ■ Minor text edits.
November 2009	1.0	Initial release.

In addition to the three general routing outputs, LEs in an LAB have register chain outputs, which allows registers in the same LAB to cascade together. The register chain output allows the LUTs to be used for combinational functions and the registers to be used for an unrelated shift register implementation. These resources speed up connections between LABs while saving local interconnect resources.

LE Operating Modes

Cyclone IV LEs operate in the following modes:

- Normal mode
- Arithmetic mode

The Quartus® II software automatically chooses the appropriate mode for common functions, such as counters, adders, subtractors, and arithmetic functions, in conjunction with parameterized functions such as the library of parameterized modules (LPM) functions. You can also create special-purpose functions that specify which LE operating mode to use for optimal performance, if required.

Normal Mode

Normal mode is suitable for general logic applications and combinational functions. In normal mode, four data inputs from the LAB local interconnect are inputs to a four-input LUT (Figure 2–2). The Quartus II Compiler automatically selects the carry-in (`cin`) or the `data3` signal as one of the inputs to the LUT. LEs in normal mode support packed registers and register feedback.

Figure 2–2 shows LEs in normal mode.

Figure 2–2. Cyclone IV Device LEs in Normal Mode

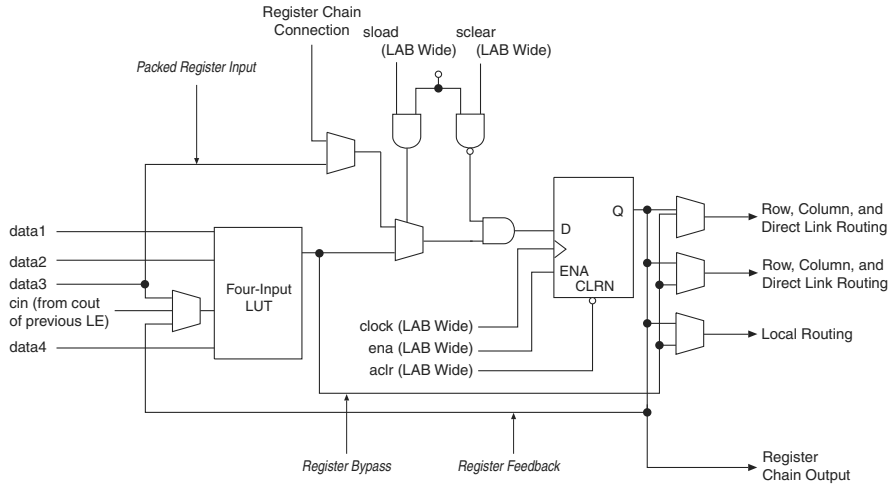


Table 3–1 lists the features supported by the M9K memory.

Table 3–1. Summary of M9K Memory Features

Feature	M9K Blocks
Configurations (depth × width)	8192 × 1 4096 × 2 2048 × 4 1024 × 8 1024 × 9 512 × 16 512 × 18 256 × 32 256 × 36
Parity bits	✓
Byte enable	✓
Packed mode	✓
Address clock enable	✓
Single-port mode	✓
Simple dual-port mode	✓
True dual-port mode	✓
Embedded shift register mode *(1)*	✓
ROM mode	✓
FIFO buffer *(1)*	✓
Simple dual-port mixed width support	✓
True dual-port mixed width support *(2)*	✓
Memory initialization file (**.mif**)	✓
Mixed-clock mode	✓
Power-up condition	Outputs cleared
Register asynchronous clears	Read address registers and output registers only
Latch asynchronous clears	Output latches only
Write or read operation triggering	Write and read: Rising clock edges
Same-port read-during-write	Outputs set to **Old Data** or **New Data**
Mixed-port read-during-write	Outputs set to **Old Data** or **Don't Care**

Notes to Table 3–1:

(1) FIFO buffers and embedded shift registers that require external logic elements (LEs) for implementing control logic.

(2) Width modes of ×32 and ×36 are not available.

For information about the number of M9K memory blocks for Cyclone IV devices, refer to the *Cyclone IV Device Family Overview* chapter in volume 1 of the *Cyclone IV Device Handbook*.

Table 5–2. GCLK Network Connections for EP4CGX30, EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 Devices (1), (2) (Part 1 of 4)

GCLK Network Clock Sources	GCLK Networks																													
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
CLKIO4/DIFFCLK_2n	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—
CLKIO5/DIFFCLK_2p	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—
CLKIO6/DIFFCLK_3n	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—
CLKIO7/DIFFCLK_3p	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	✓	—	✓	—	—	—	—	—	—	—	—	—	—	—	—
CLKIO8/DIFFCLK_5n	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	—	✓	—	—	—	—	—	—	—
CLKIO9/DIFFCLK_5p	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	✓	—	—	—	—	—	—
CLKIO10/DIFFCLK_4n/REFCLK3n	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	✓		—	—	—	—	—	—
CLKIO11/DIFFCLK_4p/REFCLK3p	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	✓	—	✓	—	—	—	—	—	—
CLKIO12/DIFFCLK_7p/REFCLK2p	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	—	✓	—
CLKIO13/DIFFCLK_7n/REFCLK2n	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	✓
CLKIO14/DIFFCLK_6p	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	✓	—
CLKIO15/DIFFCLK_6n	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	✓	—	✓
PLL_1_C0	✓	—	—	✓	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	✓	—	✓
PLL_1_C1	—	✓	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	✓	—
PLL_1_C2	✓	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	—	—	—
PLL_1_C3	—	✓	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	—	—
PLL_1_C4	—	—	✓	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	✓	✓
PLL_2_C0	—	—	—	—	—	—	✓	—	—	✓	—	✓	—	—	—	—	—	—	✓	—	—	✓	—	✓	—	—	—	—	—	—
PLL_2_C1	—	—	—	—	—	—	—	✓	—	—	✓	—	—	—	—	—	—	—	—	✓	—	—	✓	—	—	—	—	—	—	—
PLL_2_C2	—	—	—	—	—	—	✓	—	✓	—	—	—	—	—	—	—	—	—	✓	—	✓	—	—	—	—	—	—	—	—	—
PLL_2_C3	—	—	—	—	—	—	—	✓	—	✓	—	—	—	—	—	—	—	—	—	✓	—	✓	—	—	—	—	—	—	—	—
PLL_2_C4	—	—	—	—	—	—	—	—	✓	—	✓	✓	—	—	—	—	—	—	—	—	✓	—	✓	✓	—	—	—	—	—	—

Table 5–12. Dynamic Phase Shifting Control Signals (Part 2 of 2)

Signal Name	Description	Source	Destination
scanclk	Free running clock from core used in combination with phasestep to enable or disable dynamic phase shifting. Shared with scanclk for dynamic reconfiguration.	GCLK or I/O pins	PLL reconfiguration circuit
phasedone	When asserted, it indicates to core logic that the phase adjustment is complete and PLL is ready to act on a possible second adjustment pulse. Asserts based on internal PLL timing. De-asserts on the rising edge of scanclk.	PLL reconfiguration circuit	Logic array or I/O pins

Table 5–13 lists the PLL counter selection based on the corresponding PHASECOUNTERSELECT setting.

Table 5–13. Phase Counter Select Mapping

phasecounterselect			Selects
[2]	[1]	[0]	
0	0	0	All Output Counters
0	0	1	M Counter
0	1	0	C0 Counter
0	1	1	C1 Counter
1	0	0	C2 Counter
1	0	1	C3 Counter
1	1	0	C4 Counter

To perform one dynamic phase-shift, follow these steps:

1. Set PHASEUPDOWN and PHASECOUNTERSELECT as required.
2. Assert PHASESTEP for at least two SCANCLK cycles. Each PHASESTEP pulse allows one phase shift.
3. Deassert PHASESTEP after PHASEDONE goes low.
4. Wait for PHASEDONE to go high.
5. Repeat steps 1 through 4 as many times as required to perform multiple phase-shifts.

PHASEUPDOWN and PHASECOUNTERSELECT signals are synchronous to SCANCLK and must meet the t_{su} and t_h requirements with respect to the SCANCLK edges.

You can repeat dynamic phase-shifting indefinitely. For example, in a design where the VCO frequency is set to 1,000 MHz and the output clock frequency is set to 100 MHz, performing 40 dynamic phase shifts (each one yields 125 ps phase shift) results in shifting the output clock by 180°, in other words, a phase shift of 5 ns.

7. External Memory Interfaces in Cyclone IV Devices

CYIV-51007-2.6

This chapter describes the memory interface pin support and the external memory interface features of Cyclone® IV devices.

In addition to an abundant supply of on-chip memory, Cyclone IV devices can easily interface with a broad range of external memory devices, including DDR2 SDRAM, DDR SDRAM, and QDR II SRAM. External memory devices are an important system component of a wide range of image processing, storage, communications, and general embedded applications.

Altera recommends that you construct all DDR2 or DDR SDRAM external memory interfaces using the Altera® ALTMEMPHY megafunction. You can implement the controller function using the Altera DDR2 or DDR SDRAM memory controllers, third-party controllers, or a custom controller for unique application needs. Cyclone IV devices support QDR II interfaces electrically, but Altera does not supply controller or physical layer (PHY) megafunctions for QDR II interfaces.

This chapter includes the following sections:

For more information about supported maximum clock rate, device and pin planning, IP implementation, and device termination, refer to the *External Memory Interface Handbook*.

Table 7–2 lists the number of DQS or DQ groups supported on each side of the Cyclone IV E device.

Table 7–2. Cyclone IV E Device DQS and DQ Bus Mode Support for Each Side of the Device (Part 1 of 3)

Device	Package	Side	Number ×8 Groups	Number ×9 Groups	Number ×16 Groups	Number ×18 Groups	Number ×32 Groups	Number ×36 Groups
EP4CE6 EP4CE10	144-pin EQFP	Left	0	0	0	0	—	—
		Right	0	0	0	0	—	—
		Bottom *(1), (3)*	1	0	0	0	—	—
		Top *(1), (4)*	1	0	0	0	—	—
	256-pin UBGA	Left *(1)*	1	1	0	0	—	—
		Right *(2)*	1	1	0	0	—	—
		Bottom	2	2	1	1	—	—
		Top	2	2	1	1	—	—
	256-pin FBGA	Left *(1)*	1	1	0	0	—	—
		Right *(2)*	1	1	0	0	—	—
		Bottom	2	2	1	1	—	—
		Top	2	2	1	1	—	—
EP4CE15	144-pin EQFP	Left	0	0	0	0	—	—
		Right	0	0	0	0	—	—
		Bottom *(1), (3)*	1	0	0	0	—	—
		Top *(1), (4)*	1	0	0	0	—	—
	164-pin MBGA	Left	0	0	0	0	—	—
		Right	0	0	0	0	—	—
		Bottom *(1), (3)*	1	0	0	0	—	—
		Top *(1), (4)*	1	0	0	0	—	—
	256-pin MBGA	Left	1	1	0	0	—	—
		Right	1	1	0	0	—	—
		Bottom *(1), (3)*	2	2	1	1	—	—
		Top *(1), (4)*	2	2	1	1	—	—
	256-pin UBGA	Left *(1)*	1	1	0	0	—	—
		Right *(2)*	1	1	0	0	—	—
		Bottom	2	2	1	1	—	—
		Top	2	2	1	1	—	—
	256-pin FBGA	Left *(1)*	1	1	0	0	—	—
		Right *(2)*	1	1	0	0	—	—
		Bottom	2	2	1	1	—	—
		Top	2	2	1	1	—	—
	484-pin FBGA	Left	4	4	2	2	1	1
		Right	4	4	2	2	1	1
		Bottom	4	4	2	2	1	1
		Top	4	4	2	2	1	1

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

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

Table 8–6. AS DCLK Output Frequency

Oscillator	Minimum	Typical	Maximum	Unit
40 MHz	20	30	40	MHz

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

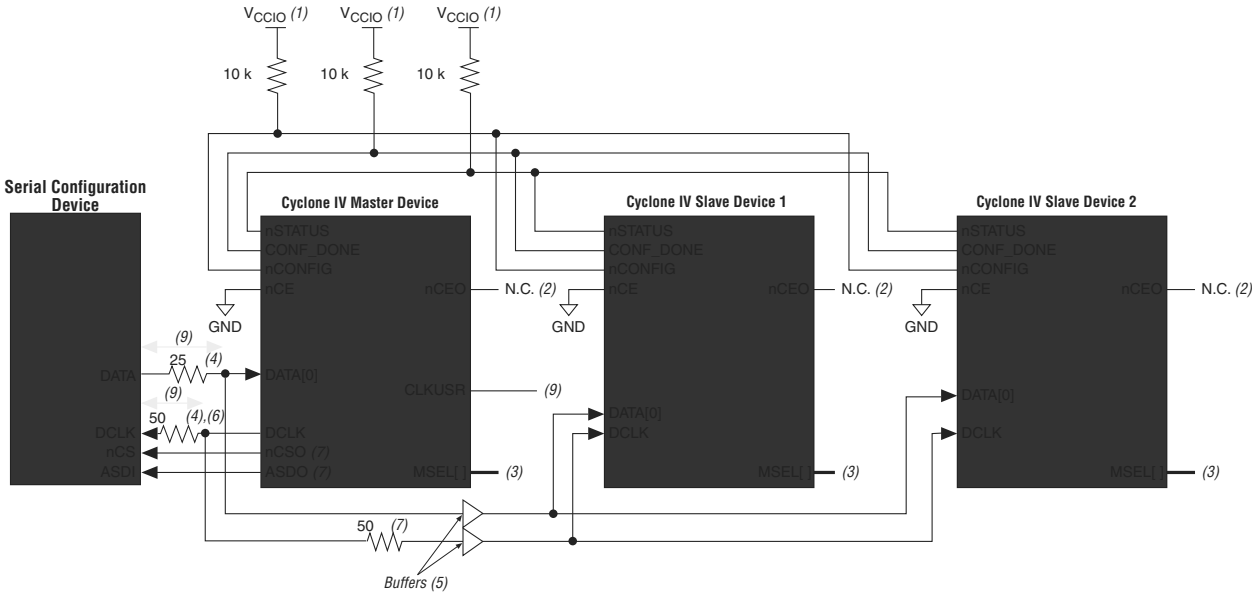
All AS configuration pins (DATA[0], DCLK, nCSO, and DATA[1]) have weak internal pull-up resistors that are always active. After configuration, these pins are set as input tri-stated and are driven high by the weak internal pull-up resistors.

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

Single SRAM Object File

The second method configures both the master device and slave devices with the same **.sof**. The serial configuration device stores one copy of the **.sof**. You must set up one or more slave devices in the chain. All the slave devices must be set up in the same way (Figure 8–5).

Figure 8–5. Multi-Device AS Configuration in Which Devices Receive the Same Data with a Single .sof



Notes to Figure 8–5:

(1) Connect the pull-up resistors to the V_{CCIO} supply of the bank in which the pin resides.

(2) 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.

(3) The MSEL pin settings vary for different configuration voltage standards and POR time. You must set the master device of the Cyclone IV device in AS mode and the slave devices in PS mode. To connect the MSEL pins for the master device in AS mode and slave devices in PS mode, 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) Connect the series resistor at the near end of the serial configuration device.

(5) Connect the repeater buffers between the master and slave devices for DATA[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.

(6) The 50-Ω series resistors are optional if the 3.3-V configuration voltage standard is applied. For optimal signal integrity, connect these 50-Ω series resistors if the 2.5- or 3.0-V configuration voltage standard is applied.

(7) These pins are dual-purpose I/O pins. The nCSO pin functions as FLASH_nCE pin in AP mode. The ASDO pin functions as DATA[1] pin in AP and FPP modes.

(8) Only Cyclone IV GX devices have an option to select CLKUSR (40 MHz maximum) as the external clock source for DCLK.

(9) For multi-devices AS configuration using Cyclone IV E with 1,0 V core voltage, the maximum board trace-length from the serial configuration device to the junction-split on both DCLK and Data0 line is 3.5 inches.

In this setup, all the Cyclone IV devices in the chain are connected for concurrent configuration. This reduces the AS configuration time because all the Cyclone IV devices are configured in one configuration cycle. Connect the nCE input pins of all the Cyclone IV devices to GND. You can either leave the nCEO output pins on all the Cyclone IV devices unconnected or use the nCEO output pins as normal user I/O pins. The DATA and DCLK pins are connected in parallel to all the Cyclone IV devices.

Altera recommends putting a buffer before the DATA and DCLK output from the master device to avoid signal strength and signal integrity issues. The buffer must not significantly change the DATA-to-DCLK relationships or delay them with respect to other AS signals (ASDI and nCS). Also, the buffer must only drive the slave devices to ensure that the timing between the master device and the serial configuration device is unaffected.

This configuration method supports both compressed and uncompressed **.sof**. Therefore, if the configuration bitstream size exceeds the capacity of a serial configuration device, you can enable the compression feature in the **.sof** or you can select a larger serial configuration device.

Guidelines for Connecting a Serial Configuration Device to Cyclone IV Devices for an AS Interface

For single- and multi-device AS configurations, the board trace length and loading between the supported serial configuration device and Cyclone IV device must follow the recommendations listed in Table 8–7.

Table 8–7. Maximum Trace Length and Loading for AS Configuration

Cyclone IV Device AS Pins	Maximum Board Trace Length from a Cyclone IV Device to a Serial Configuration Device (Inches)		Maximum Board Load (pF)
	Cyclone IV E	Cyclone IV GX	
DCLK	10	6	15
DATA[0]	10	6	30
nCSO	10	6	30
ASDO	10	6	30

Note to Table 8–7:

(1) For multi-devices AS configuration using Cyclone IV E with 1,0 V core voltage, the maximum board trace-length from the serial configuration device to the junction-split on both DCLK and Data0 line is 3.5 inches.

Estimating AS Configuration Time

AS configuration time is dominated by the time it takes to transfer data from the serial configuration device to the Cyclone IV device. This serial interface is clocked by the Cyclone IV device DCLK output (generated from a 40-MHz internal oscillator for Cyclone IV E devices, a 20- or 40-MHz internal oscillator, or an external CLKUSR of up to 40 MHz for Cyclone IV GX devices).

Equation 8–2 and Equation 8–3 show the configuration time calculations.

Equation 8–2.

$$\text{Size} \times \left(\frac{\text{maximum DCLK period}}{\text{1 bit}}\right) = \text{estimated maximum configuration ti}$$

Equation 8–3.

$$9{,}600{,}000 \text{ bits} \times \left(\frac{50 \text{ ns}}{\text{1 bit}}\right) = 480 \text{ ms}$$

The **.rbf** used by the JRunner software driver cannot be a compressed **.rbf** because the JRunner software driver uses JTAG-based configuration. During JTAG-based configuration, the real-time decompression feature is not available.

For more information about the JRunner software driver, refer to *AN 414: JRunner Software Driver: An Embedded Solution for PLD JTAG Configuration* and the source files on the Altera website at (www.altera.com).

Combining JTAG and AS Configuration Schemes

You can combine the AS configuration scheme with the JTAG-based configuration (Figure 8–28). This setup uses two 10-pin download cable headers on the board. One download cable is used in JTAG mode to configure the Cyclone IV device directly through the JTAG interface. The other download cable is used in AS mode to program the serial configuration device in-system through the AS programming interface. If you try configuring the device using both schemes simultaneously, JTAG configuration takes precedence and AS configuration terminates.

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:

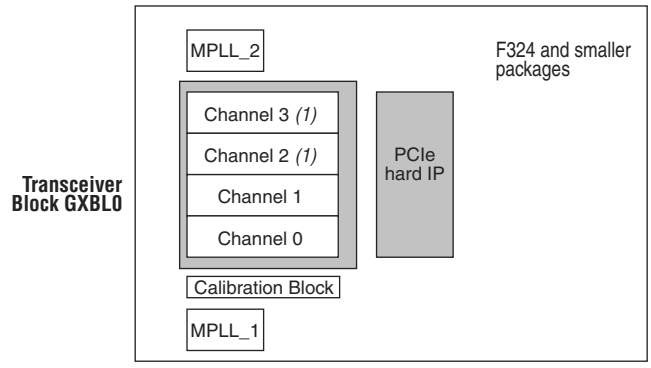
- In AS configuration scheme, you can re-engage the AS controller by moving the JTAG TAP controller to the reset state or by issuing the ACTIVE_ENGAGE instruction.

The Cyclone IV GX device includes a hard intellectual property (IP) implementation of the PCIe MegaCore® functions, supporting Gen1 ×1, ×2, and ×4 initial lane widths configured in the root port or endpoint mode. For more information, refer to "PCI-Express Hard IP Block" on page 1–46.

Transceiver Architecture

Cyclone IV GX devices offer either one or two transceiver blocks per device, depending on the package. Each block consists of four full-duplex (transmitter and receiver) channels, located on the left side of the device (in a die-top view). Figure 1–1 and Figure 1–2 show the die-top view of the transceiver block and related resource locations in Cyclone IV GX devices.

Figure 1–1. F324 and Smaller Packages with Transceiver Channels for Cyclone IV GX Devices



Note to Figure 1–1:

(1) Channel 2 and Channel 3 are not available in the F169 and smaller packages.

PIPE Interface

The PIPE interface provides a standard interface between the PCIe-compliant PHY and MAC layer as defined by the version 2.00 of the PIPE Architecture specification for Gen1 (2.5 Gbps) signaling rate. Any core or IP implementing the PHY MAC, data link, and transaction layers that supports PIPE 2.00 can be connected to the Cyclone IV GX transceiver configured in PIPE mode. Table 1–15 lists the PIPE-specific ports available from the Cyclone IV GX transceiver configured in PIPE mode and the corresponding port names in the PIPE 2.00 specification.

Table 1–15. Transceiver-FPGA Fabric Interface Ports in PIPE Mode

Transceiver Port Name	PIPE 2.00 Port Name
tx_datain[15..0] (1)	TxData[15..0]
tx_ctrlenable[1..0] (1)	TxDataK[1..0]
rx_dataout[15..0] (1)	RxData[15..0]
rx_ctrldetect[1..0] (1)	RxDataK[1..0]
tx_detectrxloop	TxDetectRx/Loopback
tx_forceelecidle	TxElecIdle
tx_forcedispcompliance	TxCompliance
pipe8b10binvpolarity	RxPolarity
powerdn[1..0] (2)	PowerDown[1..0]
pipedatavalid	RxValid
pipephydonestatus	PhyStatus
pipeelecidle	RxElecIdle
pipestatus	RxStatus[2..0]

Notes to Table 1–15:

(1) When used with PCIe hard IP block, the byte SERDES is not used. In this case, the data ports are 8 bits wide and control identifier is 1 bit wide.

(2) Cyclone IV GX transceivers do not implement power saving measures in lower power states (P0s, P1, and P2), except when putting the transmitter buffer in electrical idle in the lower power states.

Receiver Detection Circuitry

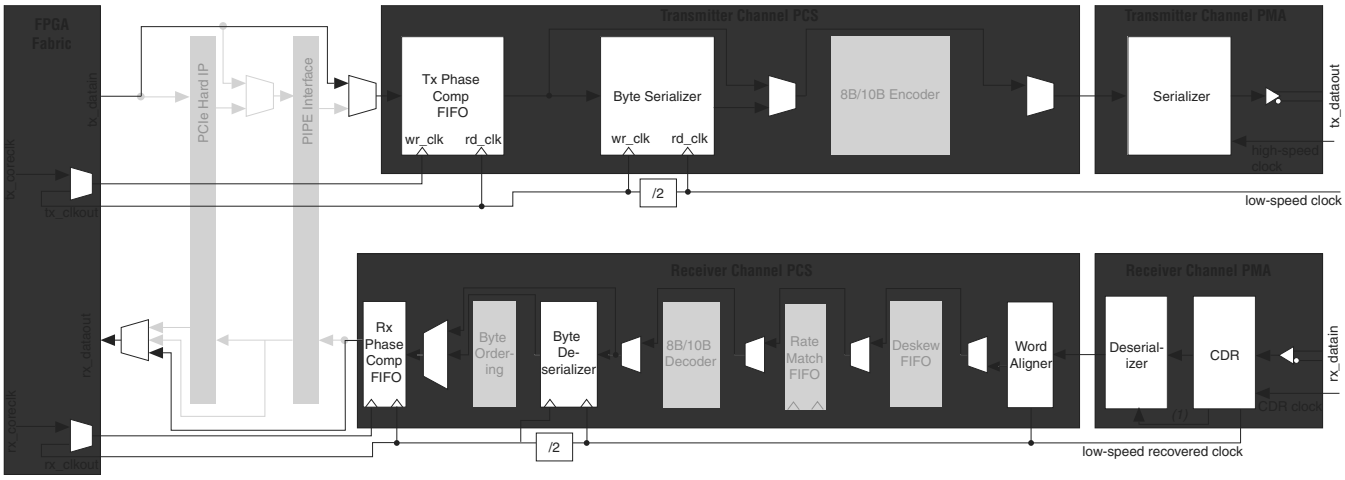
In PIPE mode, the transmitter supports receiver detection function with a built-in circuitry in the transmitter PMA. The PCIe protocol requires the transmitter to detect if a receiver is present at the far end of each lane as part of the link training and synchronization state machine sequence. This feature requires the following conditions:

- transmitter output buffer to be tri-stated
- have OCT utilization
- 125 MHz clock on the `fixedclk` port

The circuit works by sending a pulse on the common mode of the transmitter. If an active PCIe receiver is present at the far end, the time constant of the step voltage on the trace is higher compared to when the receiver is not present. The circuitry monitors the time constant of the step signal seen on the trace to decide if a receiver was detected.

Figure 1–68 shows the transceiver channel datapath and clocking when configured in SDI mode.

Figure 1–68. Transceiver Channel Datapath and Clocking when Configured in SDI Mode



Note to Figure 1–68:

(1) High-speed recovered clock.

Table 3–1. Glossary of Terms Used in this Chapter (Part 2 of 2)

Term	Description
Memory Initialization File, also known as **.mif**	A file with the **.mif** extension will be generated for **.mif**-based reconfiguration mode. It can be either in Channel Reconfiguration mode or PLL Reconfiguration mode. ■ Channel Reconfiguration mode—this file contains information about the various ALTGX MegaWizard Plug-In Manager options that you set. Each word in the **.mif** is 16 bits wide. The dynamic reconfiguration controller writes information from the **.mif** into the transceiver channel. ■ PLL Reconfiguration mode—this file contains information about the various PLL parameters and settings that you use to configure the transceiver PLL to different output frequency. The **.mif** file is 144 × 1-bit size. During PLL reconfiguration mode, the PLL reconfiguration controller shifts these 144-bit serially into the transceiver PLL.
PMA controls	Represents **analog controls** (**Voltage Output Differential** [V_{OD}], **Pre-emphasis**, **DC Gain**, and **Manual Equalization**) as displayed in both the ALTGX and ALTGX_RECONFIG MegaWizard Plug-In Managers.
Transceiver channel	Refers to a transmitter channel, a receiver channel, or a duplex channel that has both PMA and PCS blocks.

Dynamic Reconfiguration Controller Architecture

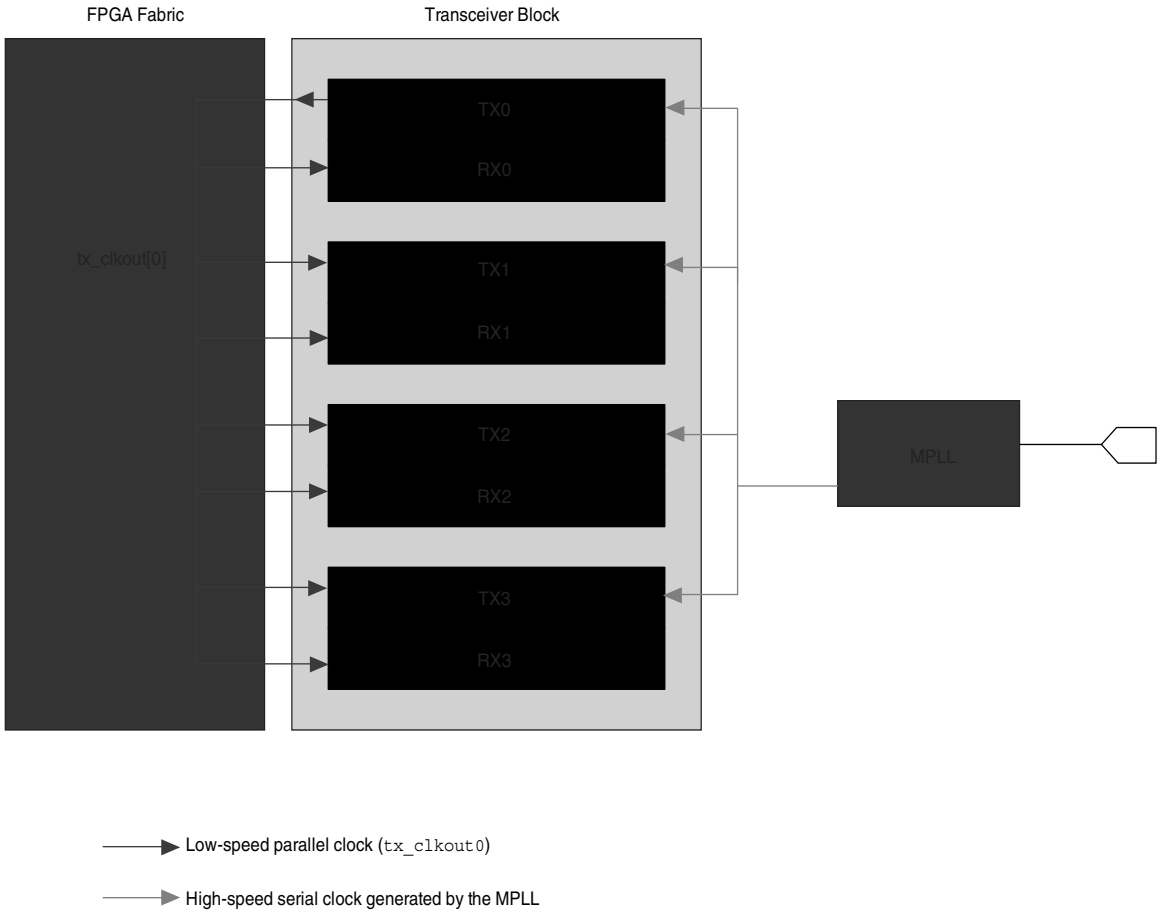
The dynamic reconfiguration controller is a soft intellectual property (IP) that utilizes FPGA-fabric resources. You can use only one controller per transceiver block. You cannot use the dynamic reconfiguration controller to control multiple Cyclone IV devices or any off-chip interfaces.

Option 1: Share a Single Transmitter Core Clock Between Receivers

- Enable this option if you want `tx_clkout` of the first channel (channel 0) of the transceiver block to provide the read clock to the Receive Phase Compensation FIFOs of the remaining receiver channels in the transceiver block.
- This option is typically enabled when all the channels of a transceiver block are in a Basic or Protocol configuration with rate matching enabled and are reconfigured to another Basic or Protocol configuration with rate matching enabled.

Figure 3–13 shows the sharing of channel 0's `tx_clkout` between all four channels of a transceiver block.

Figure 3–13. Option 1 for Receiver Core Clocking (Channel Reconfiguration Mode)



DC Characteristics

This section lists the I/O leakage current, pin capacitance, on-chip termination (OCT) tolerance, and bus hold specifications for Cyclone IV devices.

Supply Current

The device supply current requirement is the minimum current drawn from the power supply pins that can be used as a reference for power size planning. Use the Excel-based early power estimator (EPE) to get the supply current estimates for your design because these currents vary greatly with the resources used. Table 1–6 lists the I/O pin leakage current for Cyclone IV devices.

Table 1–6. I/O Pin Leakage Current for Cyclone IV Devices *(1), (2)*

Symbol	Parameter	Conditions	Device	Min	Typ	Max	Unit
I_I	Input pin leakage current	V_I = 0 V to $V_{CCIOMAX}$	—	–10	—	10	µA
I_{OZ}	Tristated I/O pin leakage current	V_O = 0 V to $V_{CCIOMAX}$	—	–10	—	10	µA

Notes to Table 1–6:

(1) This value is specified for normal device operation. The value varies during device power-up. This applies for all V_{CCIO} settings (3.3, 3.0, 2.5, 1.8, 1.5, and 1.2 V).

(2) The 10 µA I/O leakage current limit is applicable when the internal clamping diode is off. A higher current can be observed when the diode is on.

Bus Hold

The bus hold retains the last valid logic state after the source driving it either enters the high impedance state or is removed. Each I/O pin has an option to enable bus hold in user mode. Bus hold is always disabled in configuration mode.

Table 1–7 lists bus hold specifications for Cyclone IV devices.

Table 1–7. Bus Hold Parameter for Cyclone IV Devices (Part 1 of 2) *(1)*

Parameter	Condition	V_{CCIO} (V)												Unit
		1.2		1.5		1.8		2.5		3.0		3.3		
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
Bus hold low, sustaining current	$V_{IN} > V_{IL}$ (maximum)	8	—	12	—	30	—	50	—	70	—	70	—	µA
Bus hold high, sustaining current	$V_{IN} < V_{IL}$ (minimum)	–8	—	–12	—	–30	—	–50	—	–70	—	–70	—	µA
Bus hold low, overdrive current	0 V < V_{IN} < V_{CCIO}	—	125	—	175	—	200	—	300	—	500	—	500	µA
Bus hold high, overdrive current	0 V < V_{IN} < V_{CCIO}	—	–125	—	–175	—	–200	—	–300	—	–500	—	–500	µA

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

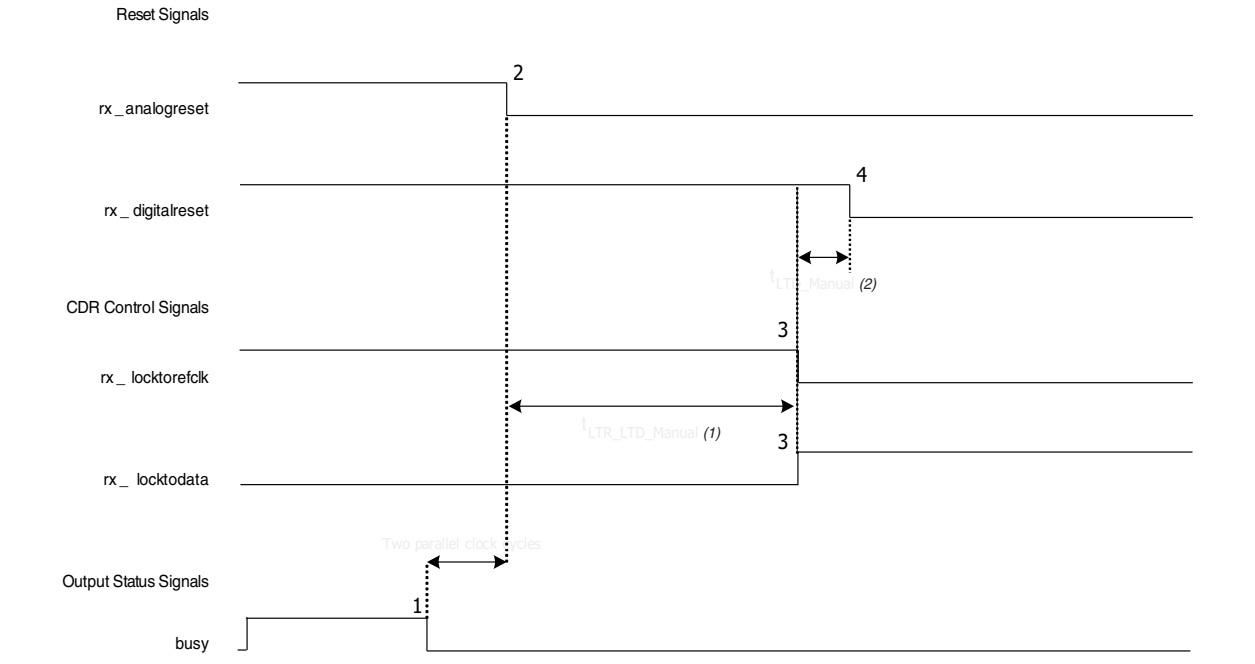


Figure 1–3 shows the lock time parameters in automatic mode.

Figure 1–3. Lock Time Parameters for Automatic Mode