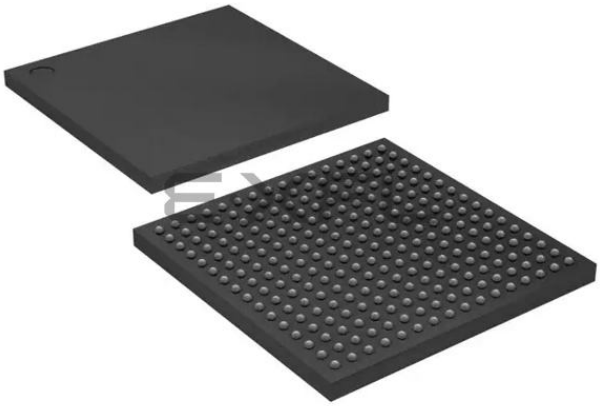


Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

Applications of Embedded - FPGAs

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

Details	
Product Status	Active
Number of LABs/CLBs	392
Number of Logic Elements/Cells	6272
Total RAM Bits	276480
Number of I/O	179
Number of Gates	-
Voltage - Supply	1.15V ~ 1.25V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	256-LBGA
Supplier Device Package	256-FBGA (17x17)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce6f17c6n

- Cyclone IV GX devices offer up to eight high-speed transceivers that provide:
  - Data rates up to 3.125 Gbps
  - 8B/10B encoder/decoder
  - 8-bit or 10-bit physical media attachment (PMA) to physical coding sublayer (PCS) interface
  - Byte serializer/deserializer (SERDES)
  - Word aligner
  - Rate matching FIFO
  - TX bit slipper for Common Public Radio Interface (CPRI)
  - Electrical idle
  - Dynamic channel reconfiguration allowing you to change data rates and protocols on-the-fly
  - Static equalization and pre-emphasis for superior signal integrity
  - 150 mW per channel power consumption
  - Flexible clocking structure to support multiple protocols in a single transceiver block
- Cyclone IV GX devices offer dedicated hard IP for PCI Express (PIPE) (PCIe) Gen 1:
  - ×1, ×2, and ×4 lane configurations
  - End-point and root-port configurations
  - Up to 256-byte payload
  - One virtual channel
  - 2 KB retry buffer
  - 4 KB receiver (Rx) buffer
- Cyclone IV GX devices offer a wide range of protocol support:
  - PCIe (PIPE) Gen 1 ×1, ×2, and ×4 (2.5 Gbps)
  - Gigabit Ethernet (1.25 Gbps)
  - CPRI (up to 3.072 Gbps)
  - XAUI (3.125 Gbps)
  - Triple rate serial digital interface (SDI) (up to 2.97 Gbps)
  - Serial RapidIO (3.125 Gbps)
  - Basic mode (up to 3.125 Gbps)
  - V-by-One (up to 3.0 Gbps)
  - DisplayPort (2.7 Gbps)
  - Serial Advanced Technology Attachment (SATA) (up to 3.0 Gbps)
  - OBSAI (up to 3.072 Gbps)

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.

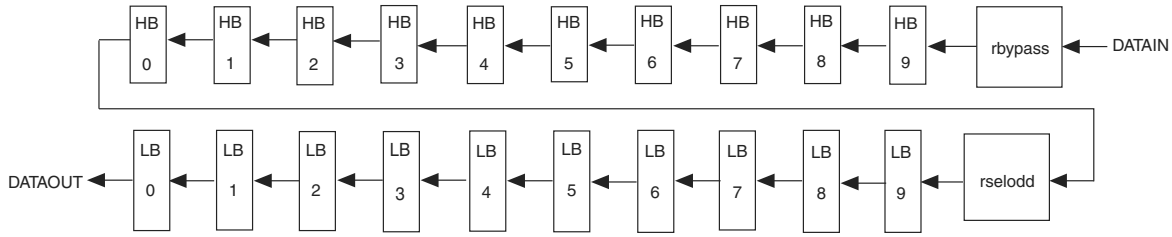
For information about the number of M9K memory blocks for Cyclone IV devices, refer to the *Cyclone IV Device Family Overview* chapter in volume 1 of the *Cyclone IV Device Handbook*.

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	—	—	—	—	—	—	—	—	✓	—	✓	✓	—	—	—	—	—	—	—	—	✓	—	✓	✓	—	—	—	—	—	—

Figure 5–25 shows the scan chain bit order sequence for one PLL post-scale counter in PLLs of Cyclone IV devices.

Figure 5–25. Scan Chain Bit Order



Charge Pump and Loop Filter

You can reconfigure the charge pump and loop filter settings to update the PLL bandwidth in real time. Table 5–8 through Table 5–10 list the possible settings for charge pump current (I_{CP}), loop filter resistor (R), and capacitor (C) values for PLLs of Cyclone IV devices.

Table 5–8. Charge Pump Bit Control

CP[2]	CP[1]	CP[0]	Setting (Decimal)
0	0	0	0
1	0	0	1
1	1	0	3
1	1	1	7

Table 5–9. Loop Filter Resistor Value Control

LFR[4]	LFR[3]	LFR[2]	LFR[1]	LFR[0]	Setting (Decimal)
0	0	0	0	0	0
0	0	0	1	1	3
0	0	1	0	0	4
0	1	0	0	0	8
1	0	0	0	0	16
1	0	0	1	1	19
1	0	1	0	0	20
1	1	0	0	0	24
1	1	0	1	1	27
1	1	1	0	0	28
1	1	1	1	0	30

The IOE registers in each I/O block share the same source for the preset or clear features. You can program preset or clear for each individual IOE, but you cannot use both features simultaneously. You can also program the registers to power-up high or low after configuration is complete. If programmed to power-up low, an asynchronous clear can control the registers. If programmed to power-up high, an asynchronous preset can control the registers. This feature prevents the inadvertent activation of the active-low input of another device upon power-up. If one register in an IOE uses a preset or clear signal, all registers in the IOE must use that same signal if they require preset or clear. Additionally, a synchronous reset signal is available for the IOE registers.

For more information about the input and output pin delay settings, refer to the *Area and Timing Optimization* chapter in volume 2 of the *Quartus II Handbook*.

PCI-Clamp Diode

Cyclone IV devices provide an optional PCI-clamp diode enabled input and output for each I/O pin. Dual-purpose configuration pins support the diode in user mode if the specific pins are not used as configuration pins for the selected configuration scheme. For example, if you are using the active serial (AS) configuration scheme, you cannot use the clamp diode on the `ASDO` and `nCSO` pins in user mode. Dedicated configuration pins do not support the on-chip diode.

The PCI-clamp diode is available for the following I/O standards:

- 3.3-V LVTTL
- 3.3-V LVCMOS
- 3.0-V LVTTL
- 3.0-V LVCMOS
- 2.5-V LVTTL/LVCMOS
- PCI
- PCI-X

If the input I/O standard is one of the listed standards, the PCI-clamp diode is enabled by default in the Quartus II software.

OCT Support

Cyclone IV devices feature OCT to provide I/O impedance matching and termination capabilities. OCT helps prevent reflections and maintain signal integrity while minimizing the need for external resistors in high pin-count ball grid array (BGA) packages. Cyclone IV devices provide I/O driver on-chip impedance matching and R_S OCT for single-ended outputs and bidirectional pins.

When using R_S OCT, programmable current strength is not available.

There are two ways to implement OCT in Cyclone IV devices:

- OCT with calibration
- OCT without calibration

When designing LVTTL/LVCMOS inputs with Cyclone IV devices, refer to the following guidelines:

- All pins accept input voltage (V_I) up to a maximum limit (3.6 V), as stated in the recommended operating conditions provided in the *Cyclone IV Device Datasheet* chapter.
- Whenever the input level is higher than the bank V_{CCIO}, expect higher leakage current.
- The LVTTL/LVCMOS I/O standard input pins can only meet the V_{IH} and V_{IL} levels according to bank voltage level.

Voltage-referenced standards are supported in an I/O bank using any number of single-ended or differential standards, as long as they use the same V_{REF} and V_{CCIO} values. For example, if you choose to implement both SSTL-2 and SSTL-18 in your Cyclone IV devices, I/O pins using these standards—because they require different V_{REF} values—must be in different banks from each other. However, the same I/O bank can support SSTL-2 and 2.5-V LVCMOS with the V_{CCIO} set to 2.5 V and the V_{REF} set to 1.25 V.

☞ When using Cyclone IV devices as a receiver in 3.3-, 3.0-, or 2.5-V LVTTL/LVCMOS systems, you are responsible for managing overshoot or undershoot to stay in the absolute maximum ratings and the recommended operating conditions, provided in the *Cyclone IV Device Datasheet* chapter.

☞ The PCI clamping diode is enabled by default in the Quartus II software for input signals with bank V_{CCIO} at 2.5, 3.0, or 3.3 V.

High-Speed Differential Interfaces

Cyclone IV devices can send and receive data through LVDS signals. For the LVDS transmitter and receiver, the input and output pins of Cyclone IV devices support serialization and deserialization through internal logic.

The BLVDS extends the benefits of LVDS to multipoint applications such as bidirectional backplanes. The loading effect and the need to terminate the bus at both ends for multipoint applications require BLVDS to drive out a higher current than LVDS to produce a comparable voltage swing. All the I/O banks of Cyclone IV devices support BLVDS for user I/O pins.

The RSDS and mini-LVDS standards are derivatives of the LVDS standard. The RSDS and mini-LVDS I/O standards are similar in electrical characteristics to LVDS, but have a smaller voltage swing and therefore provide increased power benefits and reduced electromagnetic interference (EMI).

The PPDS standard is the next generation of the RSDS standard introduced by National Semiconductor Corporation. Cyclone IV devices meet the National Semiconductor Corporation PPDS Interface Specification and support the PPDS standard for outputs only. All the I/O banks of Cyclone IV devices support the PPDS standard for output pins only.

The LVDS standard does not require an input reference voltage, but it does require a 100-Ω termination resistor between the two signals at the input buffer. An external resistor network is required on the transmitter side for the top and bottom I/O banks.

Table 6–11. High-Speed I/O Timing Definitions (Part 2 of 2)

Parameter	Symbol	Description
Input jitter tolerance (peak-to-peak)	—	Allowed input jitter on the input clock to the PLL that is tolerable while maintaining PLL lock.
Output jitter (peak-to-peak)	—	Peak-to-peak output jitter from the PLL.

Note to Table 6–11:

(1) The TCCS specification applies to the entire bank of differential I/O as long as the SERDES logic is placed in the logic array block (LAB) adjacent to the output pins.

Figure 6–21. High-Speed I/O Timing Diagram

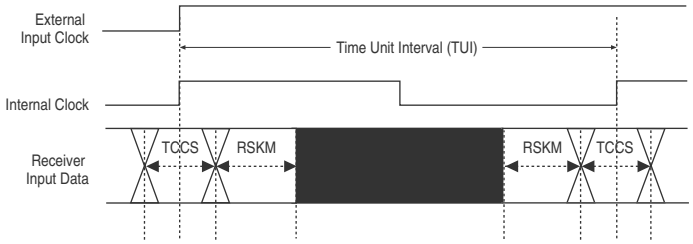
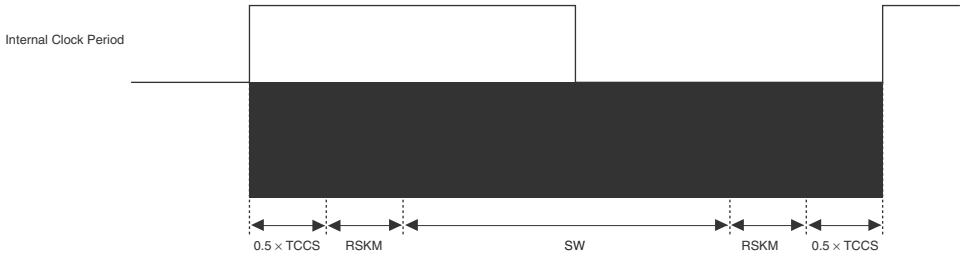


Figure 6–22 shows the Cyclone IV devices high-speed I/O timing budget.

Figure 6–22. Cyclone IV Devices High-Speed I/O Timing Budget *(1)*



Note to Figure 6–22:

(1) The equation for the high-speed I/O timing budget is:
eriod = $0.5 \times TCCS + RSKM + SW + RSKM + 0.5 \times TCCS$

For more information, refer to the *Cyclone IV Device Datasheet* chapter.

Design Guidelines

This section provides guidelines for designing with Cyclone IV devices.

Differential Pad Placement Guidelines

To maintain an acceptable noise level on the V_{CCIO} supply, you must observe some restrictions on the placement of single-ended I/O pins in relation to differential pads.

For guidelines on placing single-ended pads with respect to differential pads in Cyclone IV devices, refer to "Pad Placement and DC Guidelines" on page 6–23.

■ "FPP Configuration" on page 8–40
■ "JTAG Configuration" on page 8–45
■ "Device Configuration Pins" on page 8–62

Configuration Features

Table 8–1 lists the configuration methods you can use in each configuration scheme.

Table 8–1. Configuration Features in Cyclone IV Devices

Configuration Scheme	Configuration Method	Decompression	Remote System Upgrade (1)
AS	Serial Configuration Device	✓	✓
AP	Supported Flash Memory (2)	—	✓
PS	External Host with Flash Memory	✓	✓ (3)
	Download Cable	✓	—
FPP	External Host with Flash Memory	—	✓ (3)
JTAG based configuration	External Host with Flash Memory	—	—
	Download Cable	—	—

Notes to Table 8–1:

(1) Remote update mode is supported when you use the Remote System Upgrade feature. You can enable or disable remote update mode with an option setting in the Quartus® II software.
(2) For more information about the supported device families for the Micron commodity parallel flash, refer to Table 8–10 on page 8–22.
(3) Remote update mode is supported externally using the Parallel Flash Loader (PFL) with the Quartus II software.

Configuration Data Decompression

Cyclone IV devices support configuration data decompression, which saves configuration memory space and time. This feature allows you to store compressed configuration data in configuration devices or other memory and send the compressed bitstream to Cyclone IV devices. During configuration, Cyclone IV devices decompress the bitstream in real time and program the SRAM cells.

☞ Compression may reduce the configuration bitstream size by 35 to 55%.

When you enable compression, the Quartus II software generates configuration files with compressed configuration data. This compressed file reduces the storage requirements in the configuration device or flash memory and decreases the time required to send the bitstream to the Cyclone IV device. The time required by a Cyclone IV device to decompress a configuration file is less than the time required to send the configuration data to the device. There are two methods for enabling compression for the Cyclone IV device bitstreams in the Quartus II software:

■ Before design compilation (through the Compiler Settings menu)
■ After design compilation (through the **Convert Programming Files** dialog box)

To enable compression in the compiler settings of the project in the Quartus II software, perform the following steps:

1. On the Assignments menu, click **Device**. The **Settings** dialog box appears.
2. Click **Device and Pin Options**. The **Device and Pin Options** dialog box appears.

Configuration Process

This section describes Cyclone IV device configuration requirements and includes the following topics:

For more information about the Altera® FPGA configuration cycle state machine, refer to the *Configuring Altera FPGAs* chapter in volume 1 of the *Configuration Handbook*.

Power Up

If the device is powered up from the power-down state, V_{CCINT}, V_{CCA}, and V_{CCIO} (for the I/O banks in which the configuration and JTAG pins reside) must be powered up to the appropriate level for the device to exit from POR.

Reset

After power up, Cyclone IV devices go through POR. POR delay depends on the MSEL pin settings, which correspond to your configuration scheme. During POR, the device resets, holds nSTATUS and CONF_DONE low, and tri-states all user I/O pins (for PS and FPP configuration schemes only).

To tri-state the configuration bus for AS and AP configuration schemes, you must tie nCE high and nCONFIG low.

The user I/O pins and dual-purpose I/O pins have weak pull-up resistors, which are always enabled (after POR) before and during configuration. When the device exits POR, all user I/O pins continue to tri-state. While nCONFIG is low, the device is in reset. When nCONFIG goes high, the device exits reset and releases the open-drain nSTATUS pin, which is then pulled high by an external 10-kΩ pull-up resistor. After nSTATUS is released, the device is ready to receive configuration data and the configuration stage starts.

For more information about the value of the weak pull-up resistors on the I/O pins that are on before and during configuration, refer to the *Cyclone IV Device Datasheet* chapter.

Configuration

Configuration data is latched into the Cyclone IV device at each DCLK cycle. However, the width of the data bus and the configuration time taken for each scheme are different. After the device receives all the configuration data, the device releases the open-drain CONF_DONE pin, which is pulled high by an external 10-kΩ pull-up resistor. A low-to-high transition on the CONF_DONE pin indicates that the configuration is complete and initialization of the device can begin.

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).

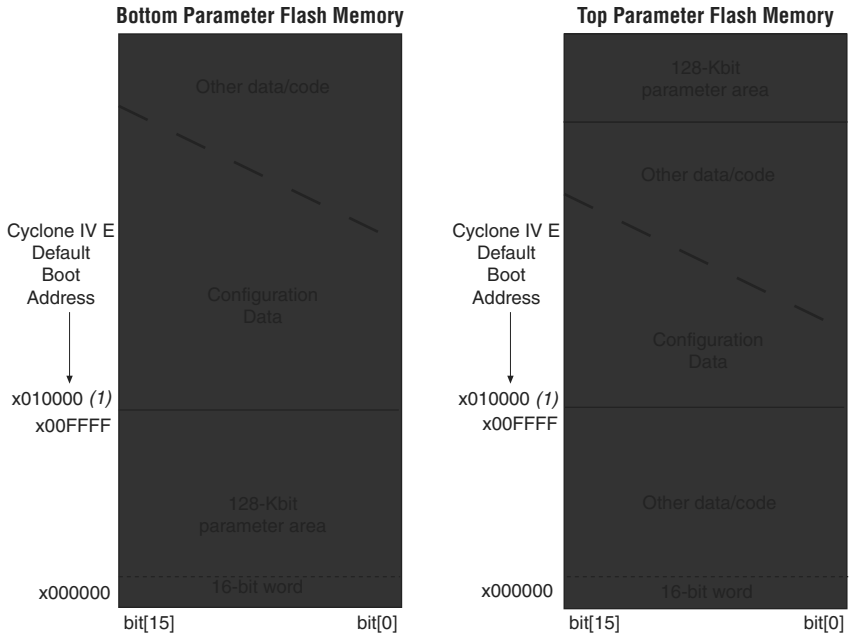
If you want to use the setup shown in Figure 8–6 to perform in-system programming of a serial configuration device and single- or multi-device AS configuration, you do not require a series resistor on the DATA line at the near end of the serial configuration device. The existing diodes and capacitors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

The default configuration boot address allows the system to use special parameter blocks in the flash memory map. Parameter blocks are at the top or bottom of the memory map. Figure 8–12 shows the configuration boot address in the AP configuration scheme. You can change the default configuration default boot address 0×010000 to any desired address using the APFC_BOOT_ADDR JTAG instruction. For more information about the APFC_BOOT_ADDR JTAG instruction, refer to "JTAG Instructions" on page 8–57.

Figure 8–12. Configuration Boot Address in AP Flash Memory Map



Note to Figure 8–12:

(1) The default configuration boot address is x010000 when represented in 16-bit word addressing.

PS Configuration

You can perform PS configuration on Cyclone IV devices with an external intelligent host, such as a MAX® II device, microprocessor with flash memory, or a download cable. In the PS scheme, an external host controls the configuration. Configuration data is clocked into the target Cyclone IV device through DATA[0] at each rising edge of DCLK.

If your system already contains a common flash interface (CFI) flash memory, you can use it for Cyclone IV device configuration storage as well. The MAX II PFL feature provides an efficient method to program CFI flash memory devices through the JTAG interface and the logic to control the configuration from the flash memory device to the Cyclone IV device.

For more information about the PFL, refer to *AN 386: Using the Parallel Flash Loader with the Quartus II Software*.

Cyclone IV devices do not support enhanced configuration devices for PS configuration.

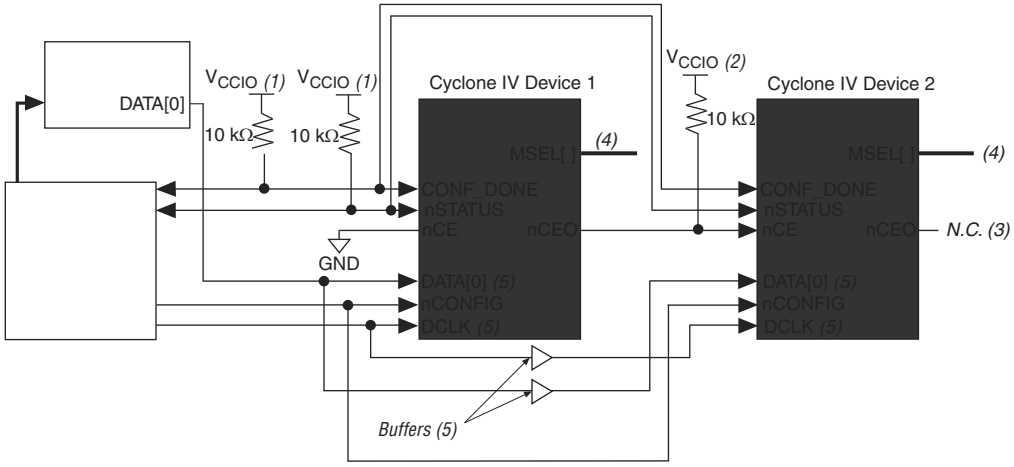
To ensure DCLK and DATA[0] are not left floating at the end of configuration, the MAX II device must drive them either high or low, whichever is convenient on your board. The DATA[0] pin is available as a user I/O pin after configuration. In the PS scheme, the DATA[0] pin is tri-stated by default in 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 configuration clock (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 CONF_DONE and INIT_DONE 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 8–14 shows how to configure multiple devices using an external host device. This circuit is similar to the PS configuration circuit for a single device, except that Cyclone IV devices are cascaded for multi-device configuration.

Figure 8–14. Multi-Device PS Configuration Using an External Host



Notes to Figure 8–14:

(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 8–3 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.

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

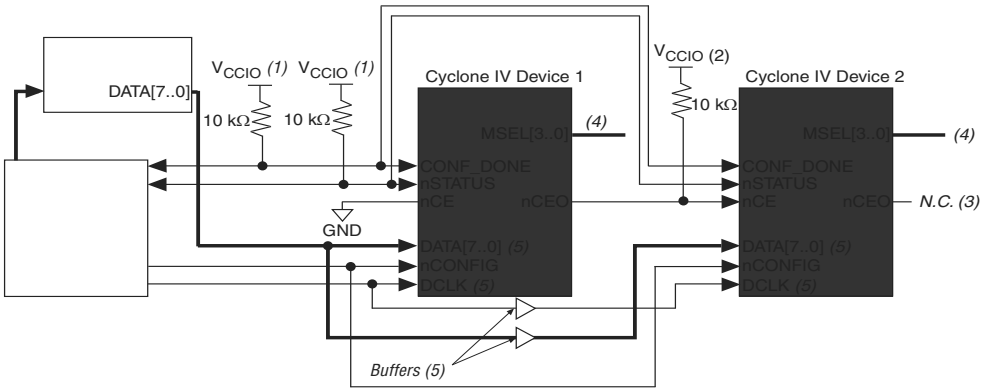
To ensure that DCLK and DATA[0] are not left floating at the end of the configuration, the MAX II device must drive them either high or low, whichever is convenient on your board. The DATA[0] pin is available as a user I/O pin after configuration. When you choose the FPP scheme in the Quartus II software, 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,

- 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

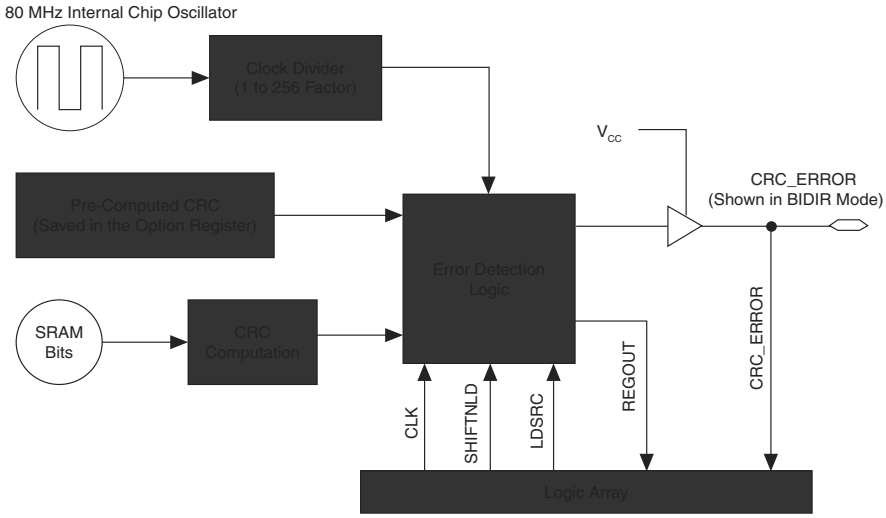
☞ When using the EN_ACTIVE_CLK and DIS_ACTIVE_CLK JTAG instructions to override the internal oscillator, you must clock the CLKUSR pin at two times the expected DCLK frequency. The CLKUSR pin allows a maximum frequency of 40 MHz (40 MHz DCLK).

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.

Figure 9–3 shows the error detection block diagram in FPGA devices and shows the interface that the WYSIWYG atom enables in your design.

Figure 9–3. Error Detection Block Diagram



The user logic is affected by the soft error failure, so reading out the 32-bit CRC signature through the `regout` should not be relied upon to detect a soft error. You should rely on the `CRC_ERROR` output signal itself, because this `CRC_ERROR` output signal cannot be affected by a soft error.

To enable the `cycloneiv_crcblock` WYSIWYG atom, you must name the atom for each Cyclone IV device accordingly.

Example 9–1 shows an example of how to define the input and output ports of a WYSIWYG atom in a Cyclone IV device.

Example 9–1. Error Detection Block Diagram

```
cycloneiv_crcblock<crcblock_name>

(

.clk(<clock source>),

.shiftnld(<shiftnld source>),

.ldsrc(<ldsrc source>),

.crcerror(<crcerror out destination>),

.regout(<output destination>),

);
```

Table 9–7 lists the input and output ports that you must include in the atom.

Table 9–7. CRC Block Input and Output Ports

Port	Input/Output	Definition
<crcblock_name>	Input	Unique identifier for the CRC block, and represents any identifier name that is legal for the given description language (for example, Verilog HDL, VHDL, and AHDL). This field is required.
.clk(*<clock source>*	Input	This signal designates the clock input of this cell. All operations of this cell are with respect to the rising edge of the clock. Whether it is the loading of the data into the cell or data out of the cell, it always occurs on the rising edge. This port is required.
.shiftnld (*<shiftnld source>*)	Input	This signal is an input into the error detection block. If `shiftnld=1`, the data is shifted from the internal shift register to the `regout` at each rising edge of `clk`. If `shiftnld=0`, the shift register parallel loads either the pre-calculated CRC value or the update register contents, depending on the `ldsrc` port input. To do this, the `shiftnld` must be driven low for at least two clock cycles. This port is required.
.ldsrc (*<ldsrc source>*)	Input	This signal is an input into the error detection block. If `ldsrc=0`, the pre-computed CRC register is selected for loading into the 32-bit shift register at the rising edge of `clk` when `shiftnld=0`. If `ldsrc=1`, the signature register (result of the CRC calculation) is selected for loading into the shift register at the rising edge of `clk` when `shiftnld=0`. This port is ignored when `shiftnld=1`. This port is required.
.crcerror (*<crcerror indicator output>*)	Output	This signal is the output of the cell that is synchronized to the internal oscillator of the device (80-MHz internal oscillator) and not to the `clk` port. It asserts high if the error block detects that a SRAM bit has flipped and the internal CRC computation has shown a difference with respect to the pre-computed value. You must connect this signal either to an output pin or a bidirectional pin. If it is connected to an output pin, you can only monitor the CRC_ERROR pin (the core cannot access this output). If the CRC_ERROR signal is used by core logic to read error detection logic, you must connect this signal to a `BIDIR` pin. The signal is fed to the core indirectly by feeding a `BIDIR` pin that has its output enable port connected to V_{CC} (see Figure 9–3 on page 9–8).
.regout (*<registered output>*)	Output	This signal is the output of the error detection shift register synchronized to the `clk` port to be read by core logic. It shifts one bit at each cycle, so you should clock the `clk` signal 31 cycles to read out the 32 bits of the shift register.

Recovering from CRC Errors

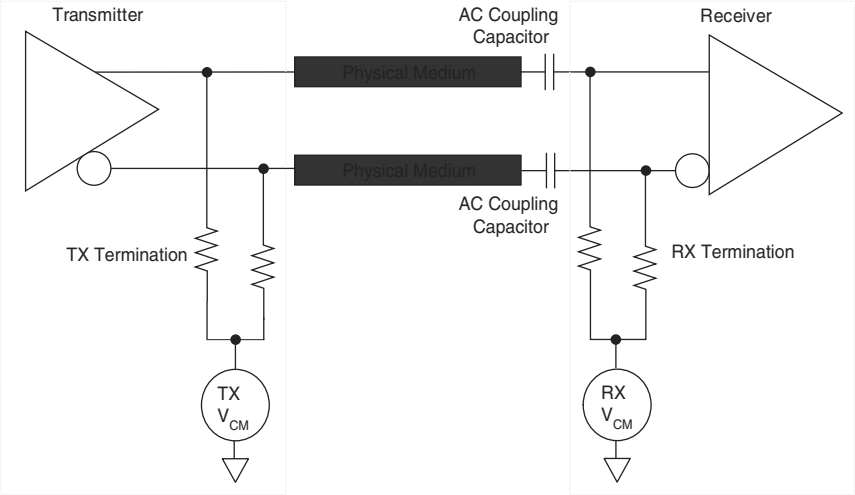
The system that the Altera FPGA resides in must control device reconfiguration. After detecting an error on the CRC_ERROR pin, strobing the nCONFIG low directs the system to perform the reconfiguration at a time when it is safe for the system to reconfigure the FPGA.

When the data bit is rewritten with the correct value by reconfiguring the device, the device functions correctly.

While soft errors are uncommon in Altera devices, certain high-reliability applications might require a design to account for these errors.

The high-speed serial link can be AC- or DC-coupled, depending on the serial protocol implementation. In an AC-coupled link, the AC-coupling capacitor blocks the transmitter DC common mode voltage as shown in Figure 1–12. Receiver OCT and on-chip biasing circuitry automatically restores the common mode voltage. The biasing circuitry is also enabled by enabling OCT. If you disable the OCT, then you must externally terminate and bias the receiver. AC-coupled links are required for PCIe, GbE, Serial RapidIO, SDI, XAUI, SATA, V-by-One and Display Port protocols.

Figure 1–12. AC-Coupled Link with OCT



In a DC-coupled link, the transmitter DC common mode voltage is seen unblocked at the receiver input buffer as shown in Figure 1–13. The link common mode voltage depends on the transmitter common mode voltage and the receiver common mode voltage. When using the receiver OCT and on-chip biasing circuitry in a DC coupled link, you must ensure the transmitter common mode voltage is compatible with the receiver common mode requirements. If you disable the OCT, you must terminate and bias the receiver externally and ensure compatibility between the transmitter and the receiver common mode voltage.

Figure 1–13. DC-Coupled Link with OCT

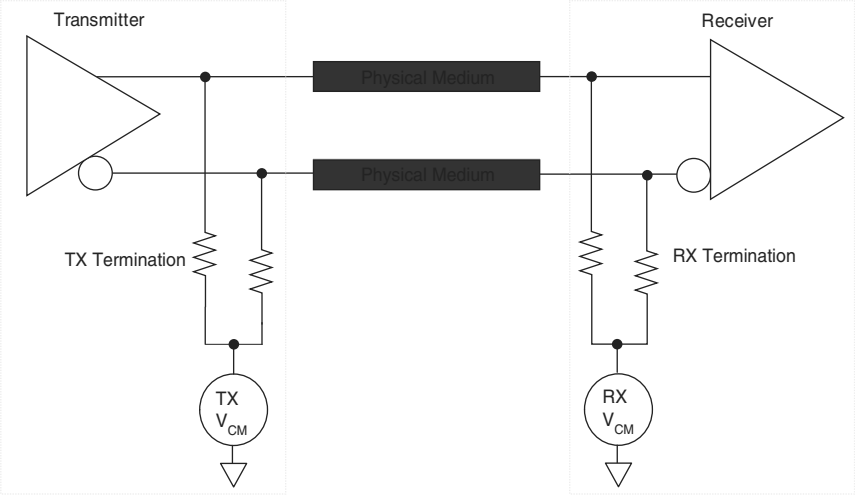
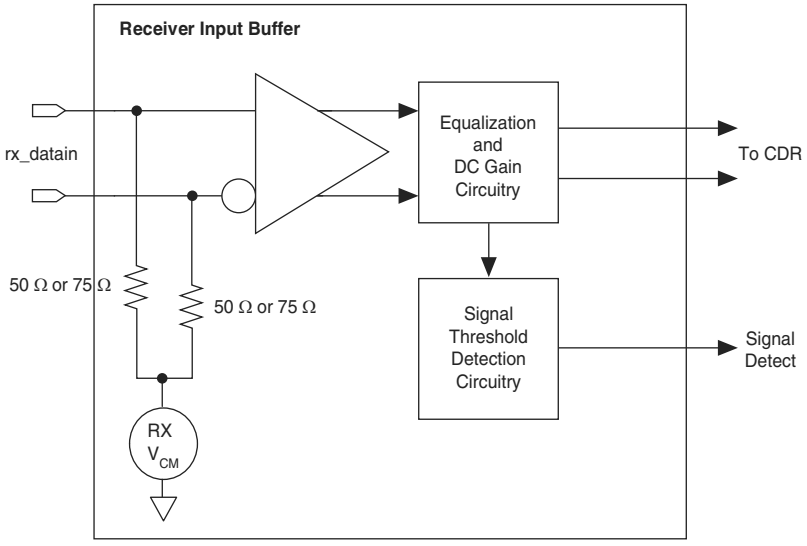


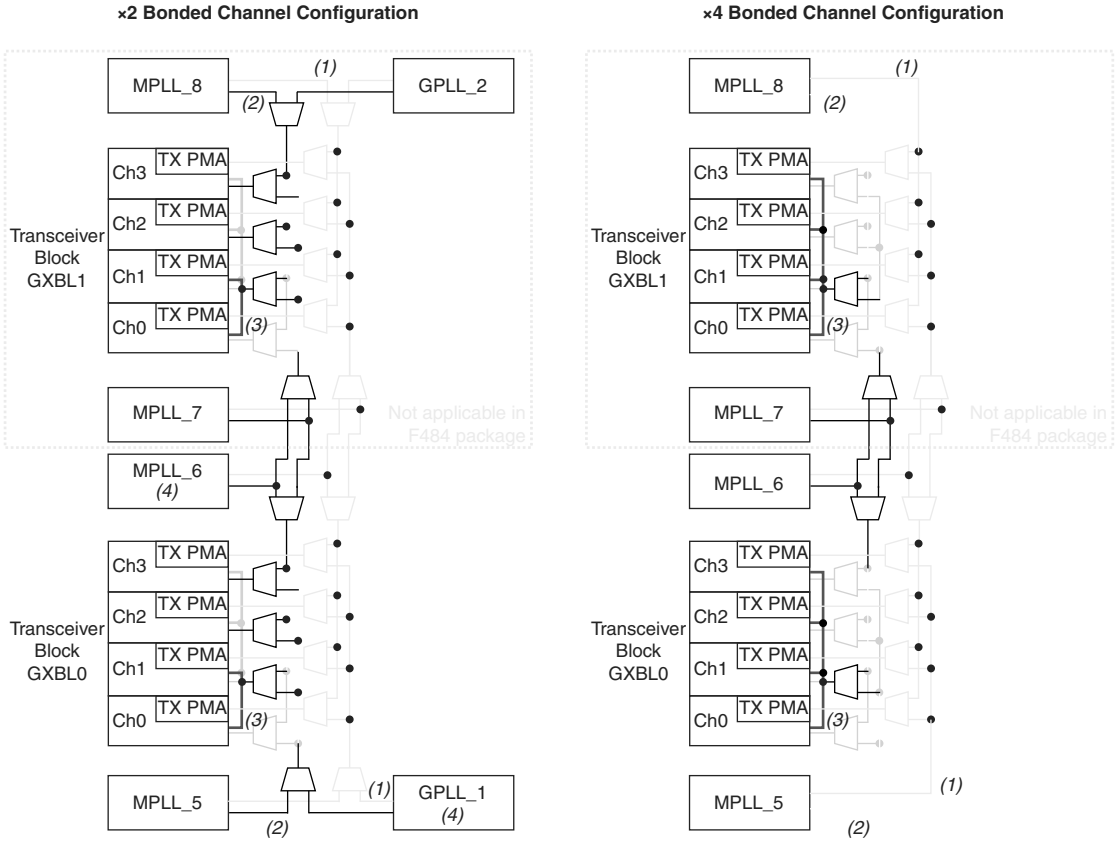
Figure 1–14 shows the receiver input buffer block diagram.

Figure 1–14. Receiver Input Buffer Block Diagram



The receiver input buffers support the following features:

Figure 1–37. Clock Distribution in Bonded (×2 and ×4) Channel Configuration for Transceivers in F484 and Larger Packages



Notes to Figure 1–37:

(1) High-speed clock.
(2) Low-speed clock.
(3) Bonded common low-speed clock path.
(4) These PLLs have restricted clock driving capability and may not reach all connected channels. For details, refer to Table 1–10.

The channel datapath clocking is similar between bonded channels in ×2 and ×4 configurations.

Figure 1–38 shows the datapath clocking in Transmitter Only operation for ×2 and ×4 bonded configurations. In these configurations, each bonded channel selects the high-speed clock from one the supported PLLs. The high-speed clock in each bonded channel feeds the respective serializer for parallel to serial operation. The common bonded low-speed clock feeds to each bonded channel that is used for the following blocks in each transmitter PCS channel:

- 8B/10B encoder
- read clock of byte serializer
- read clock of TX phase compensation FIFO

Embedded Multiplier Specifications

Table 1–26 lists the embedded multiplier specifications for Cyclone IV devices.

Table 1–26. Embedded Multiplier Specifications for Cyclone IV Devices

Mode	Resources Used	Performance					Unit
	Number of Multipliers	C6	C7, I7, A7	C8	C8L, I8L	C9L	
9 × 9-bit multiplier	1	340	300	260	240	175	MHz
18 × 18-bit multiplier	1	287	250	200	185	135	MHz

Memory Block Specifications

Table 1–27 lists the M9K memory block specifications for Cyclone IV devices.

Table 1–27. Memory Block Performance Specifications for Cyclone IV Devices

Memory	Mode	Resources Used		Performance					Unit
		LEs	M9K Memory	C6	C7, I7, A7	C8	C8L, I8L	C9L	
M9K Block	FIFO 256 × 36	47	1	315	274	238	200	157	MHz
	Single-port 256 × 36	0	1	315	274	238	200	157	MHz
	Simple dual-port 256 × 36 CLK	0	1	315	274	238	200	157	MHz
	True dual port 512 × 18 single CLK	0	1	315	274	238	200	157	MHz

Configuration and JTAG Specifications

Table 1–28 lists the configuration mode specifications for Cyclone IV devices.

Table 1–28. Passive Configuration Mode Specifications for Cyclone IV Devices (1)

Programming Mode	V_{CCINT} Voltage Level (V)	DCLK f_{MAX}	Unit
Passive Serial (PS)	1.0 (3)	66	MHz
	1.2	133	MHz
Fast Passive Parallel (FPP) (2)	1.0 (3)	66	MHz
	1.2 (4)	100	MHz

Notes to Table 1–28:

(1) For more information about PS and FPP configuration timing parameters, refer to the *Configuration and Remote System Upgrades in Cyclone IV Devices* chapter.

(2) FPP configuration mode supports all Cyclone IV E devices (except for E144 package devices) and EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 only.

(3) V_{CCINT} = 1.0 V is only supported for Cyclone IV E 1.0 V core voltage devices.

(4) Cyclone IV E devices support 1.2 V V_{CCINT}. Cyclone IV E 1.2 V core voltage devices support 133 MHz DCLK f_{MAX} for EP4CE6, EP4CE10, EP4CE15, EP4CE22, EP4CE30, and EP4CE40 only.