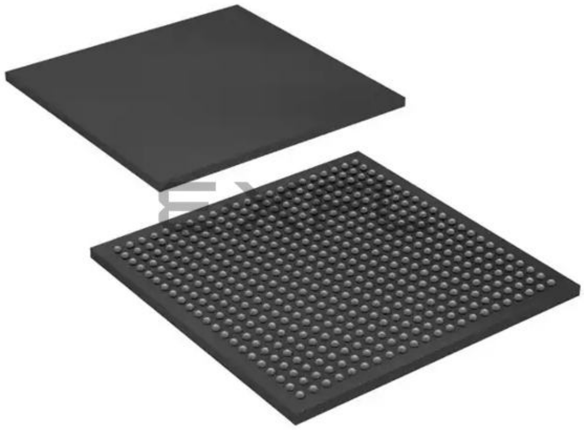


Welcome to E-XFL.COM

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	4713
Number of Logic Elements/Cells	75408
Total RAM Bits	2810880
Number of I/O	292
Number of Gates	-
Voltage - Supply	0.97V ~ 1.03V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	484-BGA
Supplier Device Package	484-FBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep4ce75f23c9l

Clock Switchover

The clock switchover feature allows the PLL to switch between two reference input clocks. Use this feature for clock redundancy or for a dual-clock domain application, such as a system that turns on the redundant clock if the previous clock stops running. Your design can automatically perform clock switchover when the clock is no longer toggling, or based on the user control signal, `clkswitch`.

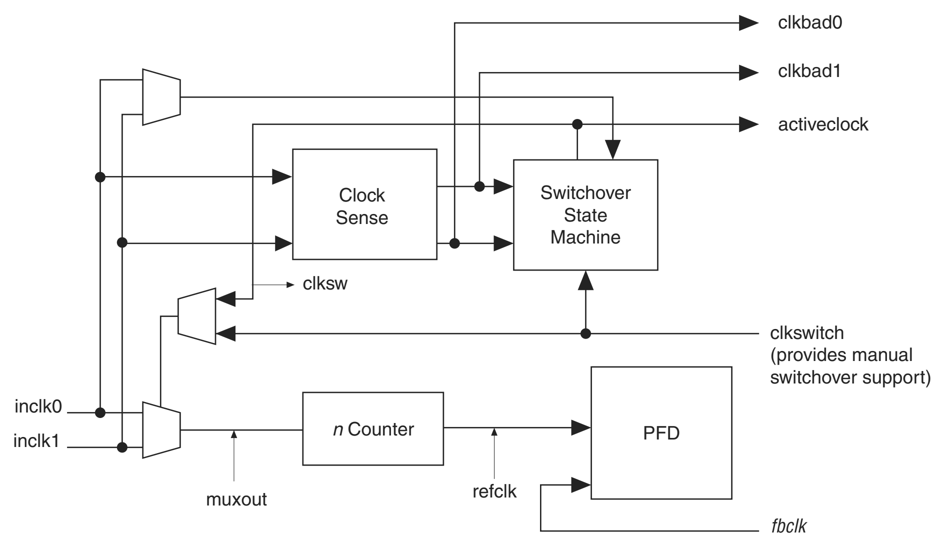
Automatic Clock Switchover

PLLs of Cyclone IV devices support a fully configurable clock switchover capability.

When the current reference clock is not present, the clock-sense block automatically switches to the backup clock for PLL reference. The clock switchover circuit also sends out three status signals—`clkbad0`, `clkbad1`, and `activeclock`—from the PLL to implement a custom switchover circuit. You can select a clock source at the backup clock by connecting it to the `inclk1` port of the PLL in your design.

Figure 5–17 shows the block diagram of the switchover circuit built into the PLL.

Figure 5–17. Automatic Clock Switchover Circuit

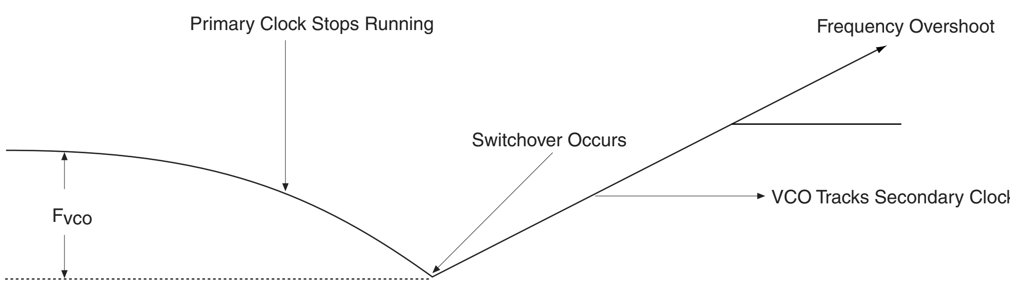


There are two ways to use the clock switchover feature:

- Use the switchover circuitry for switching from `inclk0` to `inclk1` running at the same frequency. For example, in applications that require a redundant clock with the same frequency as the reference clock, the switchover state machine generates a signal that controls the multiplexer select input shown in Figure 5–17. In this case, `inclk1` becomes the reference clock for the PLL. This automatic switchover can switch back and forth between the `inclk0` and `inclk1` clocks any number of times, when one of the two clocks fails and the other clock is available.
- Use the `clkswitch` input for user- or system-controlled switch conditions. This is possible for same-frequency switchover or to switch between inputs of different frequencies. For example, if `inclk0` is 66 MHz and `inclk1` is 200 MHz, you must control the switchover because the automatic clock-sense circuitry cannot monitor primary and secondary clock frequencies with a frequency difference of more than

- When using manual clock switchover, the difference between `inclk0` and `inclk1` can be more than 20%. However, differences between the two clock sources (frequency, phase, or both) can cause the PLL to lose lock. Resetting the PLL ensures that the correct phase relationships are maintained between the input and output clocks.
- Both `inclk0` and `inclk1` must be running when the `clkswitch` signal goes high to start the manual clock switchover event. Failing to meet this requirement causes the clock switchover to malfunction.
- Applications that require a clock switchover feature and a small frequency drift must use a low-bandwidth PLL. When referencing input clock changes, the low-bandwidth PLL reacts slower than a high-bandwidth PLL. When the switchover happens, the low-bandwidth PLL propagates the stopping of the clock to the output slower than the high-bandwidth PLL. The low-bandwidth PLL filters out jitter on the reference clock. However, the low-bandwidth PLL also increases lock time.
- After a switchover occurs, there may be a finite resynchronization period for the PLL to lock onto a new clock. The exact amount of time it takes for the PLL to re-lock is dependent on the PLL configuration.
- If the phase relationship between the input clock to the PLL and output clock from the PLL is important in your design, assert `areset` for 10 ns after performing a clock switchover. Wait for the locked signal (or gated lock) to go high before re-enabling the output clocks from the PLL.
- Figure 5–20 shows how the VCO frequency gradually decreases when the primary clock is lost and then increases as the VCO locks on to the secondary clock. After the VCO locks on to the secondary clock, some overshoot can occur (an over-frequency condition) in the VCO frequency.

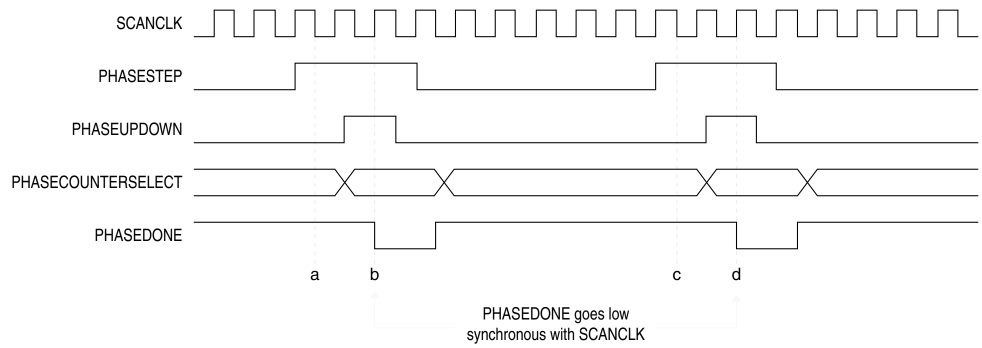
Figure 5–20. VCO Switchover Operating Frequency



- Disable the system during switchover if the system is not tolerant to frequency variations during the PLL resynchronization period. You can use the `clkbad0` and `clkbad1` status signals to turn off the PFD (`pfdena = 0`) so the VCO maintains its last frequency. You can also use the switchover state machine to switch over to the secondary clock. Upon enabling the PFD, output clock enable signals (`clkena`) can disable clock outputs during the switchover and resynchronization period. After the lock indication is stable, the system can re-enable the output clock or clocks.

Figure 5–26 shows the dynamic phase shifting waveform.

Figure 5–26. PLL Dynamic Phase Shift



The PHASESTEP signal is latched on the negative edge of SCANCLK (a,c) and must remain asserted for at least two SCANCLK cycles. Deassert PHASESTEP after PHASEDONE goes low. On the second SCANCLK rising edge (b,d) after PHASESTEP is latched, the values of PHASEUPDOWN and PHASECOUNTERSELECT are latched and the PLL starts dynamic phase-shifting for the specified counters, and in the indicated direction. PHASEDONE is deasserted synchronous to SCANCLK at the second rising edge (b,d) and remains low until the PLL finishes dynamic phase-shifting. Depending on the VCO and SCANCLK frequencies, PHASEDONE low time may be greater than or less than one SCANCLK cycle.

You can perform another dynamic phase-shift after the PHASEDONE signal goes from low to high. Each PHASESTEP pulse enables one phase shift. PHASESTEP pulses must be at least one SCANCLK cycle apart.

For information about the ALTPLL_RECONFIG MegaWizard™ Plug-In Manager, refer to the *ALTPLL_RECONFIG Megafunction User Guide*.

Spread-Spectrum Clocking

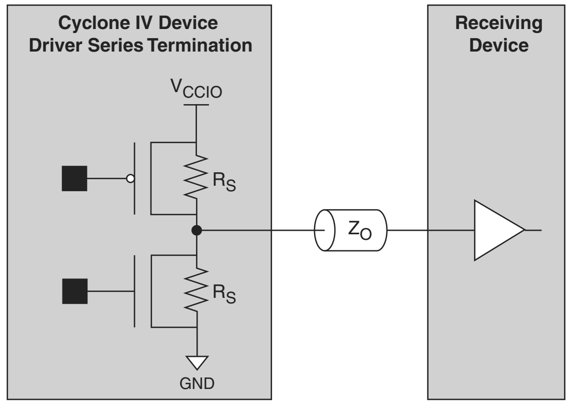
Cyclone IV devices can accept a spread-spectrum input with typical modulation frequencies. However, the device cannot automatically detect that the input is a spread-spectrum signal. Instead, the input signal looks like deterministic jitter at the input of the PLL. PLLs of Cyclone IV devices can track a spread-spectrum input clock as long as it is in the input jitter tolerance specifications and the modulation frequency of the input clock is below the PLL bandwidth, that is specified in the fitter report. Cyclone IV devices cannot generate spread-spectrum signals internally.

PLL Specifications

For information about PLL specifications, refer to the *Cyclone IV Device Datasheet* chapter.

Figure 6–4 shows the single-ended I/O standards for OCT without calibration. The R_S shown is the intrinsic transistor impedance.

Figure 6–4. Cyclone IV Devices R_S OCT Without Calibration



All I/O banks and I/O pins support impedance matching and series termination. Dedicated configuration pins and JTAG pins do not support impedance matching or series termination.

R_S OCT is supported on any I/O bank. V_{CCIO} and V_{REF} must be compatible for all I/O pins to enable R_S OCT in a given I/O bank. I/O standards that support different R_S values can reside in the same I/O bank as long as their V_{CCIO} and V_{REF} do not conflict.

Impedance matching is implemented using the capabilities of the output driver and is subject to a certain degree of variation, depending on the process, voltage, and temperature.

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

I/O Standards

Cyclone IV devices support multiple single-ended and differential I/O standards. Cyclone IV devices support 3.3-, 3.0-, 2.5-, 1.8-, 1.5-, and 1.2-V I/O standards.

Table 6–3 summarizes I/O standards supported by Cyclone IV devices and which I/O pins support them.

Table 6–3. Cyclone IV Devices Supported I/O Standards and Constraints (Part 1 of 3)

I/O Standard	Type	Standard Support	V_{CCIO} Level (in V)		Column I/O Pins			Row I/O Pins (1)	
			Input	Output	CLK, DQS	PLL_OUT	User I/O Pins	CLK, DQS	User I/O Pins
3.3-V LVTTL, 3.3-V LVCMOS (2)	Single-ended	JESD8-B	3.3/3.0/2.5 (3)	3.3	✓	✓	✓	✓	✓
3.0-V LVTTL, 3.0-V LVCMOS (2)	Single-ended	JESD8-B	3.3/3.0/2.5 (3)	3.0	✓	✓	✓	✓	✓

I/O Banks

I/O pins on Cyclone IV devices are grouped together into I/O banks. Each bank has a separate power bus.

Cyclone IV E devices have eight I/O banks, as shown in Figure 6–9. Each device I/O pin is associated with one I/O bank. All single-ended I/O standards are supported in all banks except HSTL-12 Class II, which is only supported in column I/O banks. All differential I/O standards are supported in all banks. The only exception is HSTL-12 Class II, which is only supported in column I/O banks.

Cyclone IV GX devices have up to ten I/O banks and two configuration banks, as shown in Figure 6–10 on page 6–18 and Figure 6–11 on page 6–19. The Cyclone IV GX configuration I/O bank contains three user I/O pins that can be used as normal user I/O pins if they are not used in configuration modes. Each device I/O pin is associated with one I/O bank. All single-ended I/O standards are supported except HSTL-12 Class II, which is only supported in column I/O banks. All differential I/O standards are supported in top, bottom, and right I/O banks. The only exception is HSTL-12 Class II, which is only supported in column I/O banks.

The entire left side of the Cyclone IV GX devices contain dedicated high-speed transceiver blocks for high speed serial interface applications. There are a total of 2, 4, and 8 transceiver channels for Cyclone IV GX devices, depending on the density and package of the device. For more information about the transceiver channels supported, refer to Figure 6–10 on page 6–18 and Figure 6–11 on page 6–19.

Each Cyclone IV I/O bank has a VREF bus to accommodate voltage-referenced I/O standards. Each `VREF` pin is the reference source for its V_{REF} group. If you use a V_{REF} group for voltage-referenced I/O standards, connect the `VREF` pin for that group to the appropriate voltage level. If you do not use all the V_{REF} groups in the I/O bank for voltage-referenced I/O standards, you can use the `VREF` pin in the unused voltage-referenced groups as regular I/O pins. For example, if you have SSTL-2 Class I input pins in I/O bank 1 and they are all placed in the `VREFB1N[0]` group, `VREFB1N[0]` must be powered with 1.25 V, and the remaining `VREFB1N[1..3]` pins (if available) are used as I/O pins. If multiple V_{REF} groups are used in the same I/O bank, the `VREF` pins must all be powered by the same voltage level because the `VREF` pins are shorted together within the same I/O bank.

When `VREF` pins are used as regular I/Os, they have higher pin capacitance than regular user I/O pins. This has an impact on the timing if the pins are used as inputs and outputs.

For more information about `VREF` pin capacitance, refer to the pin capacitance section in the *Cyclone IV Device Datasheet* chapter.

For information about how to identify V_{REF} groups, refer to the Cyclone IV **Device Pin-Out** files or the **Quartus II Pin Planner** tool.

Table 6–4 and Table 6–5 summarize the number of `VREF` pins in each I/O bank for the Cyclone IV device family.

Table 6–4. Number of VREF Pins Per I/O Bank for Cyclone IV E Devices (Part 1 of 2)

Device	EP4CE6			EP4CE10			EP4CE15						EP4CE22			EP4CE30			EP4CE40				EP4CE55			EP4CE75			EP4CE115	
I/O Bank *(1)*	144-EQPF	256-UBGA	256-FBGA	144-EQPF	256-UBGA	256-FBGA	144-EQPF	164-MBGA	256-MBGA	256-UBGA	256-FBGA	484-FBGA	144-EQPF	256-UBGA	256-FBGA	324-FBGA	484-FBGA	780-FBGA	324-FBGA	484-UBGA	484-FBGA	780-FBGA	484-UBGA	484-FBGA	780-FBGA	484-UBGA	484-FBGA	780-FBGA	484-FBGA	780-FBGA
1	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1	4	4	4	4	4	4	4	2	2	2	3	3	3	3	3
2	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1	4	4	4	4	4	4	4	2	2	2	3	3	3	3	3
3	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1	4	4	4	4	4	4	4	2	2	2	3	3	3	3	3
4	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1	4	4	4	4	4	4	4	2	2	2	3	3	3	3	3
5	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1	4	4	4	4	4	4	4	2	2	2	3	3	3	3	3
6	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1	4	4	4	4	4	4	4	2	2	2	3	3	3	3	3
7	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1	4	4	4	4	4	4	4	2	2	2	3	3	3	3	3

In Cyclone IV devices, the DM pins are preassigned in the device pinouts. The Quartus II Fitter treats the `DQ` and DM pins in a `DQS` group equally for placement purposes. The preassigned `DQ` and DM pins are the preferred pins to use.

Some DDR2 SDRAM and DDR SDRAM devices support error correction coding (ECC), a method of detecting and automatically correcting errors in data transmission. In 72-bit DDR2 or DDR SDRAM, there are eight ECC pins and 64 data pins. Connect the DDR2 and DDR SDRAM ECC pins to a separate `DQS` or `DQ` group in Cyclone IV devices. The memory controller needs additional logic to encode and decode the ECC data.

Address and Control/Command Pins

The address signals and the control or command signals are typically sent at a single data rate. You can use any of the user I/O pins on all I/O banks of Cyclone IV devices to generate the address and control or command signals to the memory device.

☞ Cyclone IV devices do not support QDR II SRAM in the burst length of two.

Memory Clock Pins

In DDR2 and DDR SDRAM memory interfaces, the memory clock signals (CK and CK#) are used to capture the address signals and the control or command signals. Similarly, QDR II SRAM devices use the write clocks (K and K#) to capture the address and command signals. The CK/CK# and K/K# signals are generated to resemble the write-data strobe using the DDIO registers in Cyclone IV devices.

☞ CK/CK# pins must be placed on differential I/O pins (DIFFIO in Pin Planner) and in the same bank or on the same side as the data pins. You can use either side of the device for wraparound interfaces. As seen in the Pin Planner Pad View, `CK0` cannot be located in the same row and column pad group as any of the interfacing DQ pins.

For more information about memory clock pin placement, refer to *Volume 2: Device, Pin, and Board Layout Guidelines* of the *External Memory Interface Handbook.*

Cyclone IV Devices Memory Interfaces Features

This section discusses Cyclone IV memory interfaces, including DDR input registers, DDR output registers, OCT, and phase-lock loops (PLLs).

DDR Input Registers

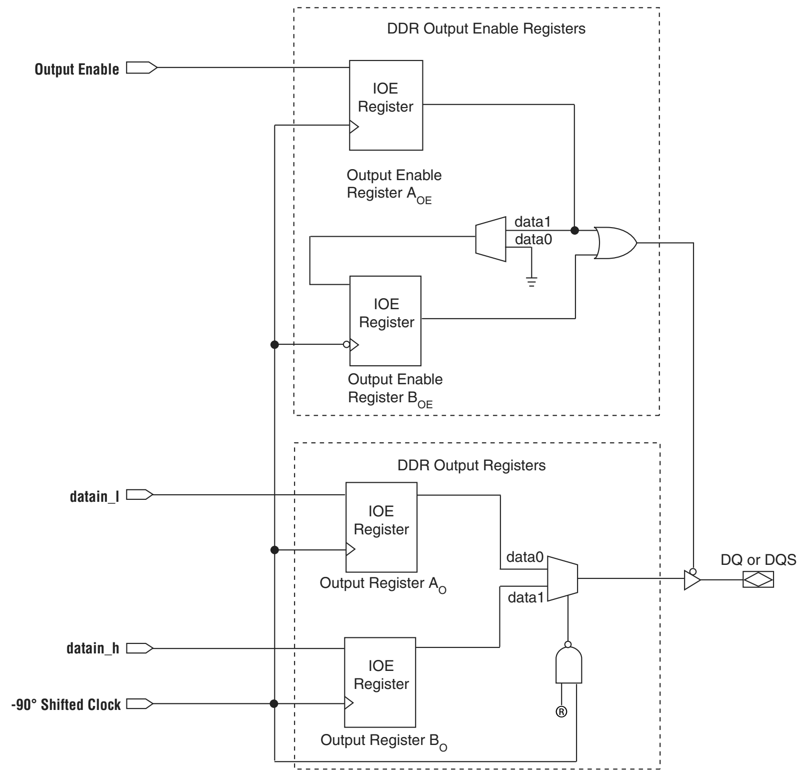
The DDR input registers are implemented with three internal logic element (LE) registers for every `DQ` pin. These LE registers are located in the logic array block (LAB) adjacent to the DDR input pin.

DDR Output Registers

A dedicated write DDIO block is implemented in the DDR output and output enable paths.

Figure 7–8 shows how a Cyclone IV dedicated write DDIO block is implemented in the I/O element (IOE) registers.

Figure 7–8. Cyclone IV Dedicated Write DDIO



The two DDR output registers are located in the I/O element (IOE) block. Two serial data streams routed through `datain_l` and `datain_h`, are fed into two registers, `output register Ao` and `output register Bo`, respectively, on the same clock edge. The output from `output register Ao` is captured on the falling edge of the clock, while the output from `output register Bo` is captured on the rising edge of the clock. The registered outputs are multiplexed by the common clock to drive the DDR output pin at twice the data rate.

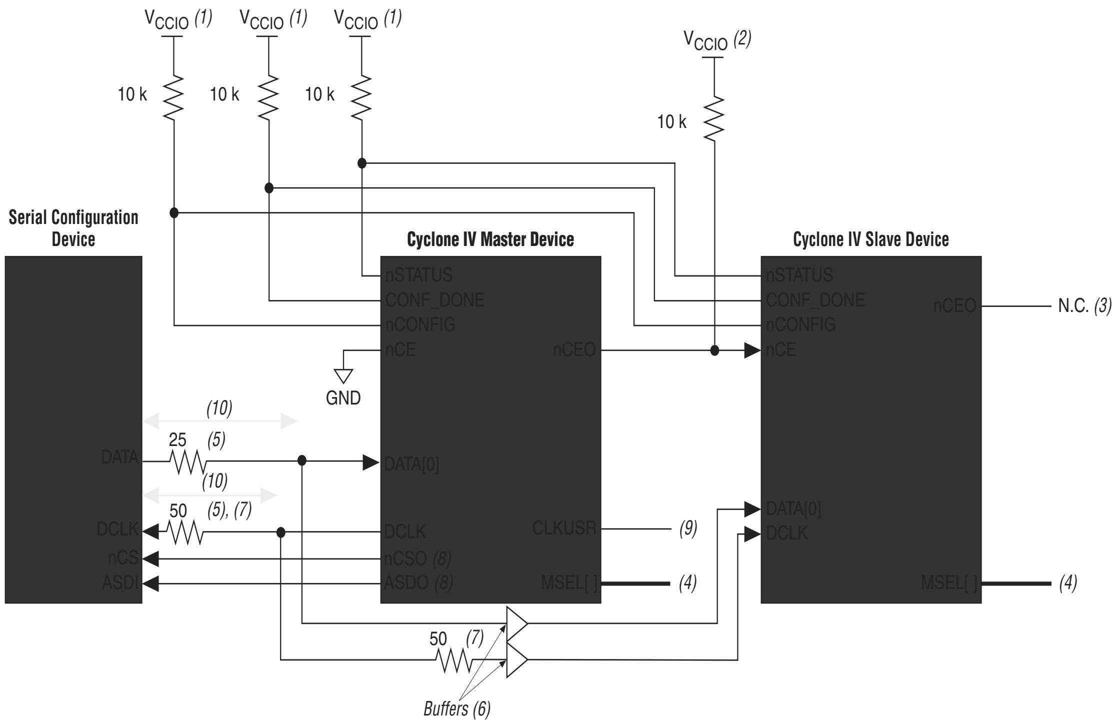
The DDR output enable path has a similar structure to the DDR output path in the IOE block. The second output enable register provides the write preamble for the `DQS` strobe in DDR external memory interfaces. This active-low output enable register extends the high-impedance state of the pin by half a clock cycle to provide the external memory's `DQS` write preamble time specification.

For more information about Cyclone IV IOE registers, refer to the *Cyclone IV Device I/O Features* chapter.

Multi-Device AS Configuration

You can configure multiple Cyclone IV devices with a single serial configuration device. When the first device captures all its configuration data from the bitstream, it drives the `nCEO` pin low, enabling the next device in the chain. If the last device in the chain is a Cyclone IV device, you can leave the `nCEO` pin of the last device unconnected or use it as a user I/O pin after configuration. The `nCONFIG`, `nSTATUS`, `CONF_DONE`, `DCLK`, and `DATA[0]` pins of each device in the chain are connected together (Figure 8–3).

Figure 8–3. Multi-Device AS Configuration



Notes to Figure 8–3:

(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 I/O bank in which the `nCE` pin resides.

(3) You can leave the `nCEO` pin unconnected or use it 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 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.

(5) Connect the series resistor at the near end of the serial configuration device.

(6) Connect the repeater buffers between the master and slave devices of the Cyclone IV device 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.

(7) 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.

(8) 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.

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

(10) 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.

Device Configuration Pins

Table 8–18 through Table 8–21 describe the connections and functionality of all the configuration related pins on Cyclone IV devices. Table 8–18 and Table 8–19 list the device pin configuration for the Cyclone IV GX and Cyclone IV E, respectively.

Table 8–18. Configuration Pin Summary for Cyclone IV GX Devices

Bank	Description	Input/Output	Dedicated	Powered By	Configuration Mode
8	Data[4:2]	Input	—	V_{CCIO}	FPP
3	Data[7:5]	Input	—	V_{CCIO}	FPP
9	nCSO (2)	Output	—	V_{CCIO}	AS
3	CRC_ERROR	Output	—	V_{CCIO}/Pull-up (1)	Optional, all modes
9	DATA[0] (2)	Input	Yes	V_{CCIO}	PS, FPP, AS
9	DATA[1]/ASDO (2)	Input	—	V_{CCIO}	FPP
		Output		V_{CCIO}	AS
3	INIT_DONE	Output	—	Pull-up	Optional, all modes
3	nSTATUS	Bidirectional	Yes	Pull-up	All modes
9	nCE	Input	Yes	V_{CCIO}	All modes
9	DCLK (2)	Input	Yes	V_{CCIO}	PS, FPP
		Output		V_{CCIO}	AS
3	CONF_DONE	Bidirectional	Yes	Pull-up	All modes
9	TDI	Input	Yes	V_{CCIO}	JTAG
9	TMS	Input	Yes	V_{CCIO}	JTAG
9	TCK	Input	Yes	V_{CCIO}	JTAG
9	nCONFIG	Input	Yes	V_{CCIO}	All modes
8	CLKUSR	Input	—	V_{CCIO}	Optional
3	nCEO	Output	—	V_{CCIO}	Optional, all modes
3	MSEL	Input	Yes	V_{CCINT}	All modes
9	TDO	Output	Yes	V_{CCIO}	JTAG
6	DEV_OE	Input	—	V_{CCIO}	Optional
6	DEV_CLRn	Input	—	V_{CCIO}	Optional

Notes to Table 8–18:

(1) The CRC_ERROR pin is a dedicated open-drain output or an optional user I/O pin. 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.

(2) To tri-state AS configuration pins in the AS configuration scheme, turn on the **Enable input tri-state on active configuration pins in user mode** option from the **Device and Pin Options** dialog box. This tri-states DCLK, nCSO, Data[0], and Data[1]/ASDO pins. Dual-purpose pins settings for these pins are ignored. To set these pins to different settings, turn off the **Enable input tri-state on active configuration pins in user mode** option and set the desired setting from the Dual-purpose Pins Setting menu.

Table 8–19. Configuration Pin Summary for Cyclone IV E Devices (Part 1 of 3)

Bank	Description	Input/Output	Dedicated	Powered By	Configuration Mode
1	nCSO (1) FLASH_nCE (2)	Output	—	V_{CCIO}	AS, AP
6	CRC_ERROR (3)	Output	—	V_{CCIO}/Pull-up (4)	Optional, all modes

Remote System Upgrade Registers

The remote system upgrade block contains a series of registers that stores the configuration addresses, watchdog timer settings, and status information. Table 8–22 lists these registers.

Table 8–22. Remote System Upgrade Registers

Register	Description
Shift register	This register is accessible by the logic array and allows the update, status, and control registers to be written and sampled by user logic. Write access is enabled in remote update mode for factory configurations to allow writing to the update register. Write access is disabled for all application configurations in remote update mode.
Control register	This register contains the current configuration address, the user watchdog timer settings, one option bit for checking early `CONF_DONE`, and one option bit for selecting the internal oscillator as the startup state machine clock. During a read operation in an application configuration, this register is read into the shift register. When a reconfiguration cycle is started, the contents of the update register are written into the control register.
Update register	This register contains data similar to that in the control register. However, it can only be updated by the factory configuration by shifting data into the shift register and issuing an update operation. When a reconfiguration cycle is triggered by the factory configuration, the control register is updated with the contents of the update register. During a read in a factory configuration, this register is read into the shift register.
Status register	This register is written by the remote system upgrade circuitry on every reconfiguration to record the cause of the reconfiguration. This information is used by the factory configuration to determine the appropriate action following a reconfiguration. During a capture cycle, this register is read into the shift register.

The control and status registers of the remote system upgrade are clocked by the 10-MHz internal oscillator (the same oscillator that controls the user watchdog timer) or the `CLKUSR`. However, the shift and update registers of the remote system upgrade are clocked by the maximum frequency of 40-MHz user clock input (`RU_CLK`). There is no minimum frequency for `RU_CLK`.

Remote System Upgrade Control Register

The remote system upgrade control register stores the application configuration address, the user watchdog timer settings, and option bits for a application configuration. In remote update mode for the AS configuration scheme, the control register address bits are set to all zeros (`24'b0`) at power up to load the AS factory configuration. In remote update mode for the AP configuration scheme, the control register address bits are set to `24'h010000` (`24'b1 0000 0000 0000 0000`) at power up to load the AP default factory configuration. However, for the AP configuration scheme, you can change the default factory configuration address to any desired address using the `APFC_BOOT_ADDR` JTAG instruction. Additionally, a factory configuration in remote update mode has write access to this register.

☞ WYSIWYG is an optimization technique that performs optimization on a VQM (Verilog Quartus Mapping) netlist in the Quartus II software.

Error Detection Block

Table 9–3 lists the types of CRC detection to check the configuration bits.

Table 9–3. Types of CRC Detection to Check the Configuration Bits

First Type of CRC Detection	Second Type of CRC Detection
■ CRAM error checking ability (32-bit CRC) during user mode, for use by the `CRC_ERROR` pin. ■ There is only one 32-bit CRC value. This value covers all the CRAM data.	■ 16-bit CRC embedded in every configuration data frame. ■ During configuration, after a frame of data is loaded into the device, the pre-computed CRC is shifted into the CRC circuitry. ■ Simultaneously, the CRC value for the data frame shifted-in is calculated. If the pre-computed CRC and calculated CRC values do not match, `nSTATUS` is set low. ■ Every data frame has a 16-bit CRC. Therefore, there are many 16-bit CRC values for the whole configuration bit stream. ■ Every device has a different length of configuration data frame.

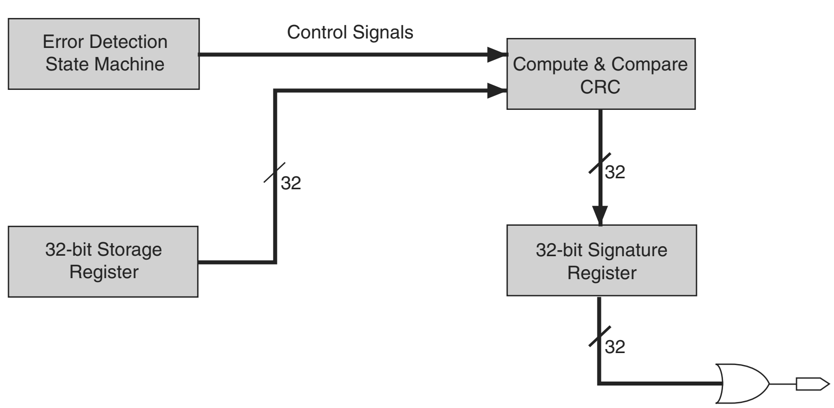
This section focuses on the first type—the 32-bit CRC when the device is in user mode.

Error Detection Registers

There are two sets of 32-bit registers in the error detection circuitry that store the computed CRC signature and pre-calculated CRC value. A non-zero value on the signature register causes the `CRC_ERROR` pin to set high.

Figure 9–1 shows the block diagram of the error detection block and the two related 32-bit registers: the signature register and the storage register.

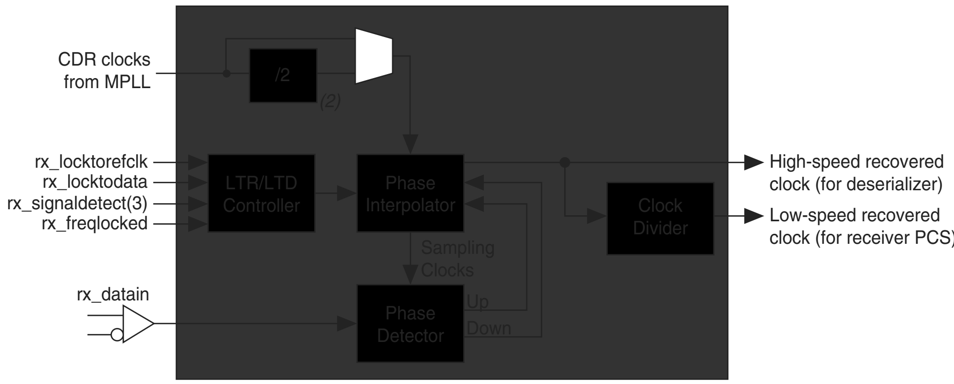
Figure 9–1. Error Detection Block Diagram



Clock Data Recovery

Each receiver channel has an independent CDR unit to recover the clock from the incoming serial data stream. The high-speed recovered clock is used to clock the deserializer for serial-to-parallel conversion of the received input data, and low-speed recovered clock to clock the receiver PCS blocks. Figure 1–15 illustrates the CDR unit block diagram.

Figure 1–15. CDR Unit Block Diagram



Notes to Figure 1–15:

(1) Optional RX local divider for CDR clocks from multipurpose PLL is only available in each CDR unit for EP4CGX30 (F484 package), EP4CGX50, and EP4CGX75 devices. This block is used with the transceiver dynamic reconfiguration feature. For more information, refer to the *Cyclone IV Dynamic Reconfiguration* chapter and *AN 609: Implementing Dynamic Reconfiguration in Cyclone IV GX Devices*.

(2) CDR state transition in automatic lock mode is not dependent on `rx_signaldetect` signal, except when configured in PCI Express (PIPE) mode only.

Each CDR unit gets the reference clock from one of the two multipurpose phase-locked loops (PLLs) adjacent to the transceiver block. The CDR works by tracking the incoming data with a phase detector and finding the optimum sampling clock phase from the phase interpolator unit. The CDR operations are controlled by the LTR/LTD controller block, where the CDR may operate in the following states:

- Lock-to-reference (LTR) state—phase detector disabled and CDR ignores incoming data
- Lock-to-data (LTD) state—phase detector enabled and CDR tracks incoming data to find the optimum sampling clock phase

State transitions are supported with automatic lock mode and manual lock mode.

Automatic Lock Mode

Upon receiver power-up and reset cycle, the CDR is put into LTR state. Transition to the LTD state is performed automatically when both of the following conditions are met:

- Signal detection circuitry indicates the presence of valid signal levels at the receiver input buffer. This condition is valid for PCI Express (PIPE) mode only. CDR transitions are not dependent on signal detection circuitry in other modes.
- The recovered clock is within the configured part per million (ppm) frequency threshold setting with respect to the CDR clocks from multipurpose PLL.

The byte ordering block operates in either word-alignment-based byte ordering or user-controlled byte ordering modes.

In word-alignment-based byte ordering mode, the byte ordering block starts looking for the byte ordering pattern in the byte-deserialized data and restores the order if necessary when it detects a rising edge on the `rx_syncstatus` signal. Whenever the byte ordering pattern is found, the `rx_byteorderalignstatus` signal is asserted regardless if the pad byte insertion is necessary. If the byte ordering block detects another rising edge on the `rx_syncstatus` signal from the word aligner, it deasserts the `rx_byteorderalignstatus` signal and repeats the byte ordering operation.

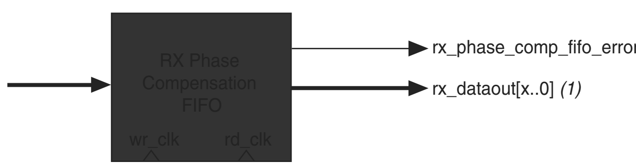
In user-controlled byte ordering mode, the byte ordering operation is user-triggered using `rx_enabyteord` port. A rising edge on `rx_enabyteord` port triggers the byte ordering block to start looking for the byte ordering pattern in the byte-deserialized data and restores the order if necessary. When the byte ordering pattern is found, the `rx_byteorderalignstatus` signal is asserted regardless if a pad byte insertion is necessary.

RX Phase Compensation FIFO

The RX phase compensation FIFO compensates for the phase difference between the parallel receiver clock and the FPGA fabric interface clock, when interfacing the receiver channel to the FPGA fabric (directly or through the PIPE and PCIe hard IP blocks). The FIFO is four words deep, with latency between two to three parallel clock cycles.

Figure 1–24 shows the RX phase compensation FIFO block diagram.

Figure 1–24. RX Phase Compensation FIFO Block Diagram



Note to Figure 1–24:

(1) Parameter x refers to the transceiver channel width, where 8, 10, 16, or 20 bits are supported.

☞ The FIFO can operate in registered mode, contributing to only one parallel clock cycle of latency in the Deterministic Latency functional mode. For more information, refer to “Deterministic Latency Mode” on page 1–73. For more information about FIFO clocking, refer to “FPGA Fabric-Transceiver Interface Clocking” on page 1–43.

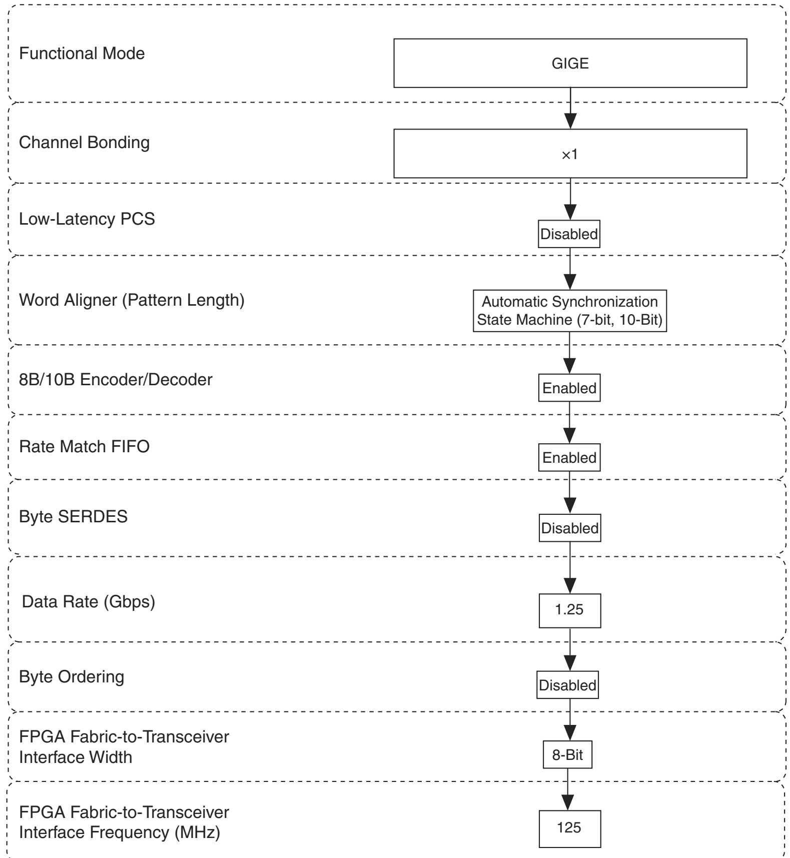
Miscellaneous Receiver PCS Feature

The receiver PCS supports the following additional feature:

- Output bit-flip—reverses the bit order at a byte level at the output of the receiver phase compensation FIFO. For example, if the 16-bit parallel receiver data at the output of the receiver phase compensation FIFO is '10111100 10101101' (16'hBCAD), enabling this option reverses the data on `rx_dataout` port to '00111101 10110101' (16'h3DB5).

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

Figure 1–56. Transceiver Configuration in GIGE Mode

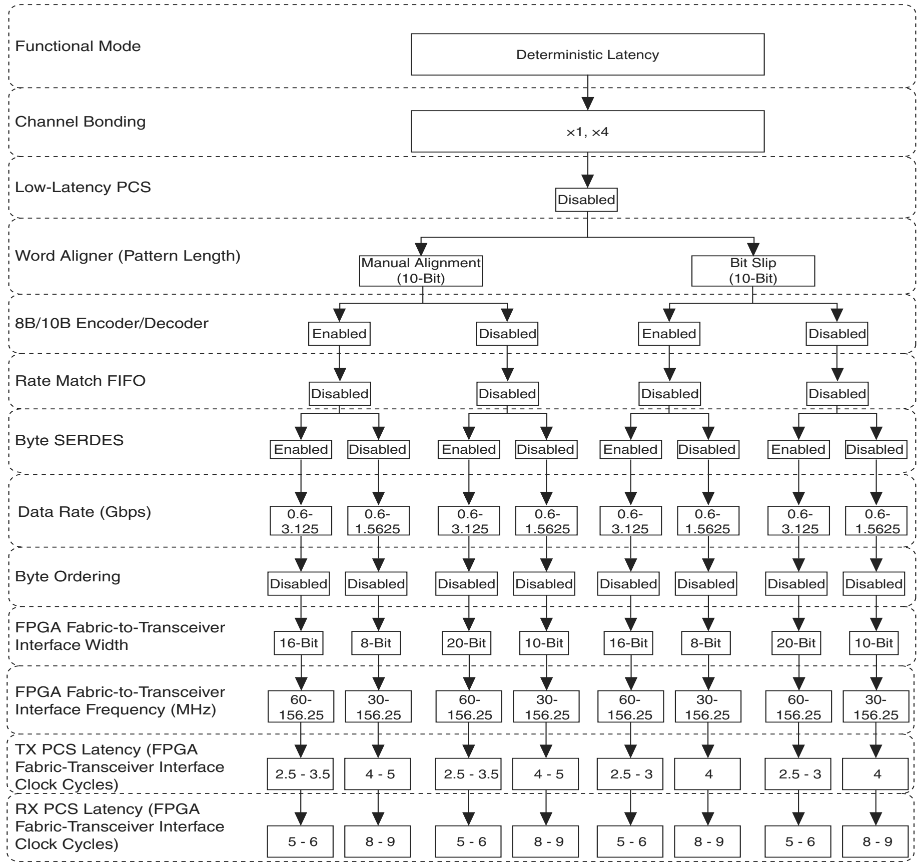


When configured in GIGE mode, three encoded comma (/K28.5/) code groups are transmitted automatically after deassertion of `tx_digitalreset` and before transmitting user data on the `tx_datain` port. This could affect the synchronization state machine behavior at the receiver.

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

Figure 1–67 shows the transceiver configuration in Deterministic Latency mode.

Figure 1–67. Transceiver Configuration in Deterministic Latency Mode



Both CPRI and OBSAI protocols define the serial interface connecting the base station component (specifically channel cards) and remote radio heads (specifically radio frequency cards) in a radio base station system with fiber optics. The protocols require the accuracy of round trip delay measurement for single-hop and multi-hop connections to be within ± 16.276 ns. The Cyclone IV GX transceivers support the following CPRI and OBSAI line rates using Deterministic Latency mode:

- CPRI —614.4 Mbps, 1.2288 Gbps, 2.4576 Gbps, and 3.072 Gbps
- OBSAI—768 Mbps, 1.536 Gbps, and 3.072 Gbps

For more information about deterministic latency implementation, refer to *AN 610: Implementing Deterministic Latency for CPRI and OBSAI Protocols in Stratix IV, HardCopy IV, Arria II GX, and Cyclone IV Devices*.

Registered Mode Phase Compensation FIFO

In Deterministic Latency mode, the RX phase compensation FIFO is set to registered mode while the TX phase compensation FIFO supports optional registered mode. When set into registered mode, the phase compensation FIFO acts as a register and eliminates the latency uncertainty through the FIFOs.

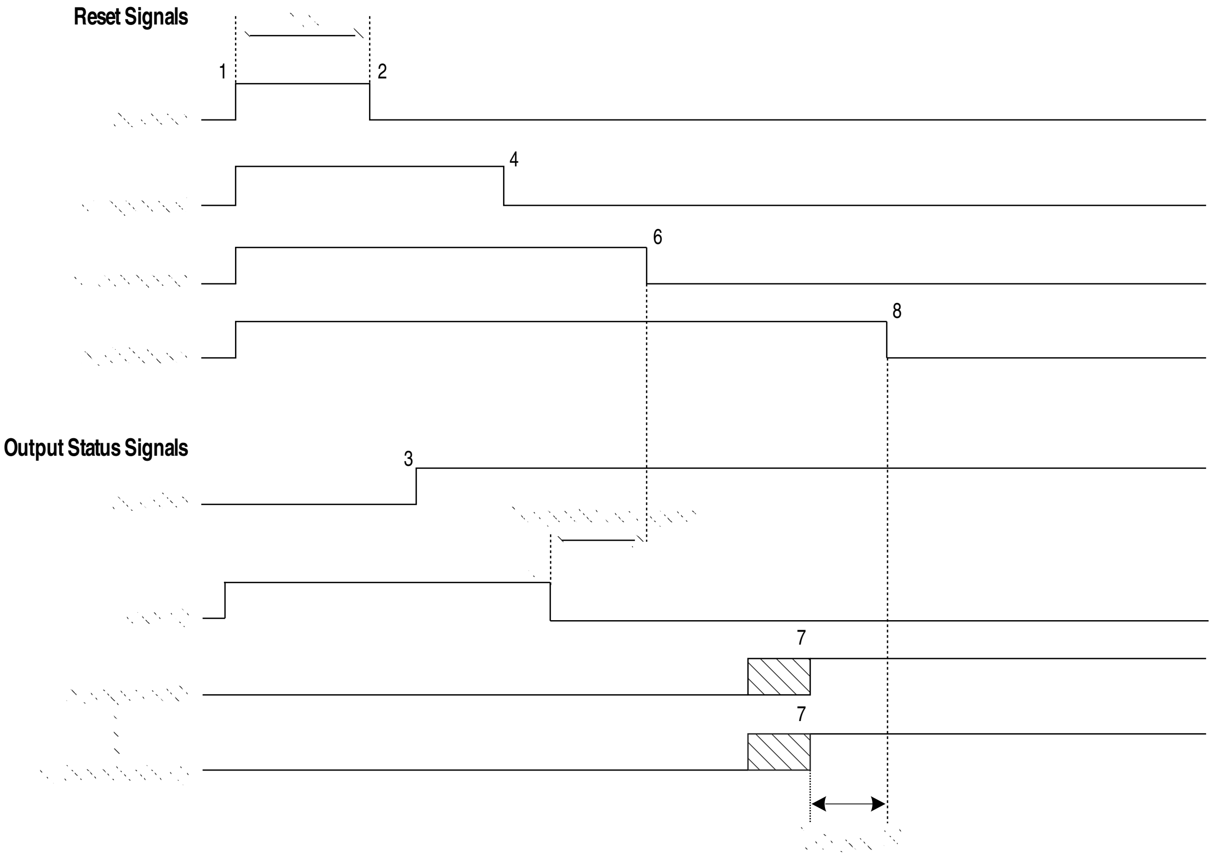
Table 1–27. Receiver Ports in ALTGX Megafunction for Cyclone IV GX (Part 3 of 3)

<table>
<tr><th>Block</th><th>Port Name</th><th>Input/ Output</th><th>Clock Domain</th><th>Description</th></tr>
<tr><td rowspan="3">RX PCS</td><td>rx_coreclk</td><td>Output</td><td>Clock signal</td><td>Optional read clock port for the RX phase compensation FIFO.</td></tr>
<tr><td>rx_phase_comp_fifo_error</td><td>Output</td><td>Synchronous to <code>tx_clkout</code> (non-bonded modes) or <code>coreclkout</code> (bonded modes)</td><td>RX phase compensation FIFO full or empty indicator.
■ A high level indicates FIFO is either full or empty.</td></tr>
<tr><td>rx_bitslipboundaryselectout</td><td>Output</td><td>Asynchronous signal.</td><td>Indicate the number of bits slipped in the word aligner configured in manual alignment mode.
■ Values range from 0 to 9.</td></tr>
<tr><td rowspan="6">RX PMA</td><td>rx_datain</td><td>Input</td><td>N/A</td><td>Receiver serial data input port.</td></tr>
<tr><td>rx_freqlocked</td><td>Output</td><td>Asynchronous signal</td><td>Receiver CDR lock state indicator
■ A high level indicates the CDR is in LTD state.
■ A low level indicates the CDR is in LTR state.</td></tr>
<tr><td>rx_locktodata</td><td>Input</td><td>Asynchronous signal</td><td>Receiver CDR LTD state control signal
■ A high level forces the CDR to LTD state
■ When deasserted, the receiver CDR lock state depends on the <code>rx_locktorefclk</code> signal level.</td></tr>
<tr><td>rx_locktorefclk</td><td>Input</td><td>Asynchronous signal</td><td>Receiver CDR LTR state control signal.
■ The <code>rx_locktorefclk</code> and <code>rx_locktodata</code> signals control whether the receiver CDR states as follows: [<code>rx_locktodata</code>:<code>rx_locktorefclk</code>]
  ■ 2'b00—receiver CDR is in automatic lock mode
  ■ 2b'01—receiver CDR is in manual lock mode (LTR state)
  ■ 2b'1x—receiver CDR is in manual lock mode (LTD state)</td></tr>
<tr><td>rx_signaldetect</td><td>Output</td><td>Asynchronous signal</td><td>Signal threshold detect indicator.
■ Available in Basic mode when 8B/10B encoder/decoder is used, and in PIPE mode.
■ A high level indicates that the signal present at the receiver input buffer is above the programmed signal detection threshold value.</td></tr>
<tr><td>rx_recovclkout</td><td>Output</td><td>Clock signal</td><td>CDR low-speed recovered clock
■ Only available in the GIGE mode for applications such as Synchronous Ethernet.</td></tr>
</table>

Receiver and Transmitter Channel—Receiver CDR in Automatic Lock Mode

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

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



Notes to Figure 2–4:

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

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

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

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

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

Read Transaction

If you want to read the existing values from a specific channel connected to the ALTGX_RECONFIG instance, observe the corresponding byte positions of the PMA control output port after the read transaction is completed.

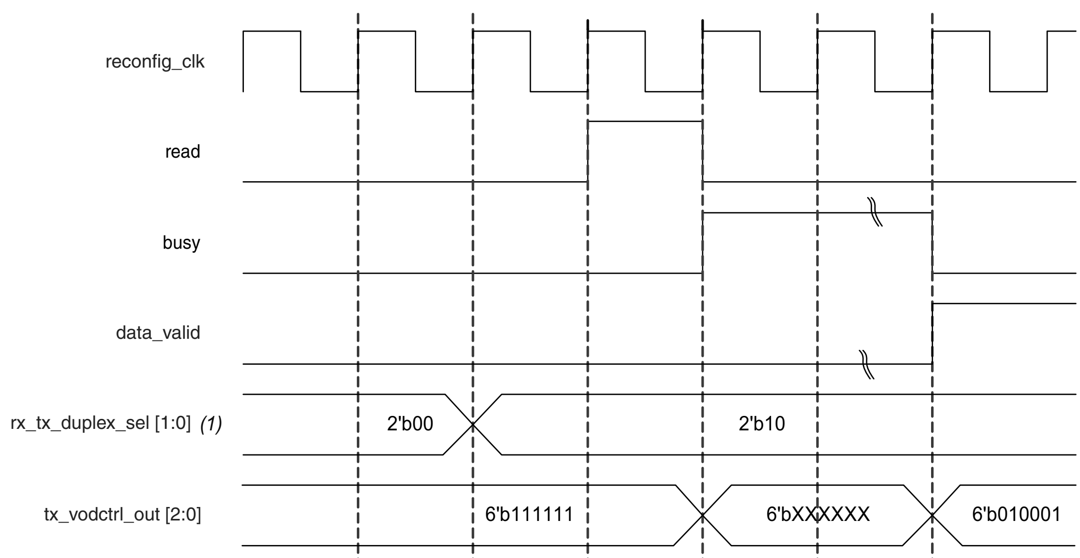
For example, if the number of channels controlled by the ALTGX_RECONFIG is two, the `tx_vodctrl_out` is 6 bits wide. The `tx_vodctrl_out[2:0]` signal corresponds to channel 1 and the `tx_vodctrl_out[5:3]` signal corresponds to channel 2.

To complete a read transaction to the V_{OD} values of the second channel, perform the following steps:

1. Before you initiate a read transaction, set the `rx_tx_duplex_sel` port to **2'b10** so that only the transmit PMA controls are read from the transceiver channel.
2. Ensure that the `busy` signal is low before you start a read transaction.
3. Assert the read signal for one `reconfig_clk` clock cycle. This initiates the read transaction.
4. The `busy` output status signal is asserted high to indicate that the dynamic reconfiguration controller is busy reading the PMA control settings.
5. When the read transaction has completed, the `busy` signal goes low. The `data_valid` signal is asserted, indicating that the data available at the read control signal is valid.
6. To read the current V_{OD} values in channel 2, observe the values in `tx_vodctrl_out[5:3]`.

In the waveform example shown in Figure 3–7, the transmit V_{OD} settings written in channels 1 and 2 prior to the read transaction are 3'b001 and 3'b010, respectively.

Figure 3–7. Read Transaction Waveform—Use the same control signal for all the channels Option Enabled



Note to Figure 3–7:

(1) In this waveform example, you want to read from only the transmitter portion of all the channels.

☞ Simultaneous write and read transactions are not allowed.

Table 1–21. Transceiver Specification for Cyclone IV GX Devices (Part 2 of 4)

Symbol/ Description	Conditions	C6			C7, I7			C8			Unit
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Receiver											
Supported I/O Standards	1.4 V PCML, 1.5 V PCML, 2.5 V PCML, LVPECL, LVDS										
Data rate (F324 and smaller package) *(15)*	—	600	—	2500	600	—	2500	600	—	2500	Mbps
Data rate (F484 and larger package) *(15)*	—	600	—	3125	600	—	3125	600	—	2500	Mbps
Absolute V_{MAX} for a receiver pin *(3)*	—	—	—	1.6	—	—	1.6	—	—	1.6	V
Operational V_{MAX} for a receiver pin	—	—	—	1.5	—	—	1.5	—	—	1.5	V
Absolute V_{MIN} for a receiver pin	—	–0.4	—	—	–0.4	—	—	–0.4	—	—	V
Peak-to-peak differential input voltage V_{ID} (diff p-p)	V_{ICM} = 0.82 V setting, Data Rate = 600 Mbps to 3.125 Gbps	0.1	—	2.7	0.1	—	2.7	0.1	—	2.7	V
V_{ICM}	V_{ICM} = 0.82 V setting	—	820 ± 10%	—	—	820 ± 10%	—	—	820 ± 10%	—	mV
Differential on-chip termination resistors	100–Ω setting	—	100	—	—	100	—	—	100	—	Ω
	150–Ω setting	—	150	—	—	150	—	—	150	—	Ω
Differential and common mode return loss	PIPE, Serial Rapid I/O SR, SATA, CPRI LV, SDI, XAUI	Compliant									—
Programmable ppm detector *(4)*	—	± 62.5, 100, 125, 200, 250, 300									ppm
Clock data recovery (CDR) ppm tolerance (without spread-spectrum clocking enabled)	—	—	—	±300 *(5)*, ±350 *(6)*, *(7)*	—	—	±300 *(5)*, ±350 *(6)*, *(7)*	—	—	±300 *(5)*, ±350 *(6)*, *(7)*	ppm
CDR ppm tolerance (with synchronous spread-spectrum clocking enabled) *(8)*	—	—	—	350 to –5350 *(7)*, *(9)*	—	—	350 to –5350 *(7)*, *(9)*	—	—	350 to –5350 *(7)*, *(9)*	ppm
Run length	—	—	80	—	—	80	—	—	80	—	UI
Programmable equalization	No Equalization	—	—	1.5	—	—	1.5	—	—	1.5	dB
	Medium Low	—	—	4.5	—	—	4.5	—	—	4.5	dB
	Medium High	—	—	5.5	—	—	5.5	—	—	5.5	dB
	High	—	—	7	—	—	7	—	—	7	dB