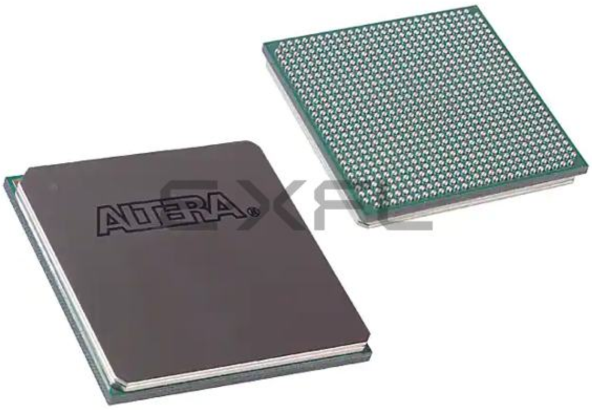





## Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

## Applications of Embedded - FPGAs

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

| Details | |
|---|---|
| Product Status | Active |
| Number of LABs/CLBs | 7155 |
| Number of Logic Elements/Cells | 114480 |
| Total RAM Bits | 3981312 |
| Number of I/O | 528 |
| Number of Gates | - |
| Voltage - Supply | 1.15V ~ 1.25V |
| Mounting Type | Surface Mount |
| Operating Temperature | 0°C ~ 85°C (TJ) |
| Package / Case | 780-BGA |
| Supplier Device Package | 780-FBGA (29x29) |
| Purchase URL | https://www.e-xfl.com/product-detail/intel/ep4ce115f29c7n |

# PLLs in Cyclone IV Devices

Cyclone IV GX devices offer two variations of PLLs: general purpose PLLs and multipurpose PLLs. Cyclone IV E devices only have the general purpose PLLs.

The general purpose PLLs are used for general-purpose applications in the FPGA fabric and periphery such as external memory interfaces. The multipurpose PLLs are used for clocking the transceiver blocks. When the multipurpose PLLs are not used for transceiver clocking, they can be used for general-purpose clocking.

For more details about the multipurpose PLLs used for transceiver clocking, refer to the *Cyclone IV Transceivers* chapter.

Cyclone IV GX devices contain up to eight general purpose PLLs and multipurpose PLLs while Cyclone IV E devices have up to four general purpose PLLs that provide robust clock management and synthesis for device clock management, external system clock management, and high-speed I/O interfaces.

For more information about the number of general purpose PLLs and multipurpose PLLs in each device density, refer to the *Cyclone IV Device Family Overview* chapter.

The general I/O pins cannot drive the PLL clock input pins.

Table 5–5 lists the features available in Cyclone IV GX PLLs.

**Table 5–5. Cyclone IV GX PLL Features (Part 1 of 2)**

| Features | Availability | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | General Purpose PLLs | | | | Multipurpose PLLs | | | | | |
| | PLL_1 (1), (10) | PLL_2 (1), (10) | PLL_3 (2) | PLL_4 (3) | PLL_1 (4) | PLL_2 (4) | PLL_5 (1), (10) | PLL_6 (1), (10) | PLL_7 (1) | PLL_8 (1) |
| C (output counters) | 5 | | | | | | | | | |
| M, N, C counter sizes | 1 to 512 (5) | | | | | | | | | |
| Dedicated clock outputs | 1 single-ended or 1 differential pair | | | | | | | | | |
| Clock input pins | 12 single-ended or 6 differential pairs (6) and 4 differential pairs (7) | | | | | | | | | |
| Spread-spectrum input clock tracking | ✓ (8) | | | | | | | | | |
| PLL cascading | Through GCLK | | | | | | | | | |
| Source-Synchronous Mode | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | — | — | ✓ |
| No Compensation Mode | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Normal Mode | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | — | — | ✓ |
| Zero Delay Buffer Mode | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | — | — | ✓ |
| Deterministic Latency Compensation Mode | ✓ | ✓ | — | — | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Phase shift resolution (9) | Down to 96 ps increments | | | | | | | | | |
| Programmable duty cycle | ✓ | | | | | | | | | |
| Output counter cascading | ✓ | | | | | | | | | |

**Table 5–10. Loop Filter Control of High Frequency Capacitor**

| LFC[1] | LFC[0] | Setting (Decimal) |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 1 | 3 |

## Bypassing a PLL Counter

Bypassing a PLL counter results in a divide (N, C0 to C4 counters) factor of one.

Table 5–11 lists the settings for bypassing the counters in PLLs of Cyclone IV devices.

**Table 5–11. PLL Counter Settings**

| PLL Scan Chain Bits [0..8] Settings | | | | | | | | | Description |
|---|---|---|---|---|---|---|---|---|---|
| LSB | | | | | | | | MSB | |
| X | X | X | X | X | X | X | X | 1 *(1)* | PLL counter bypassed |
| X | X | X | X | X | X | X | X | 0 *(1)* | PLL counter not bypassed |

**Note to Table 5–11:**

(1) Bypass bit.

To bypass any of the PLL counters, set the bypass bit to 1. The values on the other bits are then ignored.

## Dynamic Phase Shifting

The dynamic phase shifting feature allows the output phase of individual PLL outputs to be dynamically adjusted relative to each other and the reference clock without sending serial data through the scan chain of the corresponding PLL. This feature simplifies the interface and allows you to quickly adjust $t_{CO}$ delays by changing output clock phase shift in real time. This is achieved by incrementing or decrementing the VCO phase-tap selection to a given C counter or to the M counter. The phase is shifted by 1/8 the VCO frequency at a time. The output clocks are active during this phase reconfiguration process.

Table 5–12 lists the control signals that are used for dynamic phase shifting.

**Table 5–12. Dynamic Phase Shifting Control Signals (Part 1 of 2)**

| Signal Name | Description | Source | Destination |
|---|---|---|---|
| `phasecounterselect[2..0]` | Counter Select. Three bits decoded to select either the M or one of the C counters for phase adjustment. One address map to select all C counters. This signal is registered in the PLL on the rising edge of `scanclk`. | Logic array or I/O pins | PLL reconfiguration circuit |
| `phaseupdown` | Selects dynamic phase shift direction; 1= UP, 0 = DOWN. Signal is registered in the PLL on the rising edge of `scanclk`. | Logic array or I/O pins | PLL reconfiguration circuit |
| `phasestep` | Logic high enables dynamic phase shifting. | Logic array or I/O pins | PLL reconfiguration circuit |

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.

The `CLKIN/REFCLK` pins are powered by dedicated $V_{CC\_CLKIN3A}$, $V_{CC\_CLKIN3B}$, $V_{CC\_CLKIN8A}$, and $V_{CC\_CLKIN8B}$ power supplies separately in their respective I/O banks to avoid the different power level requirements in the same bank for GPIO.

**Table 6–10. Cyclone IV GX HSSI REFCLK I/O Standard Support Using GPIO CLKIN Pins** *(1), (2)*

| I/O Standard | HSSI Protocol | Coupling | Termination | VCC_CLKIN Level | | I/O Pin Type | | |
|---|---|---|---|---|---|---|---|---|
| | | | | Input | Output | Column I/O | Row I/O | Supported I/O Banks |
| LVDS | All | Differential AC (Need off chip resistor to restore $V_{CM}$) | Off chip | 2.5V | Not supported | Yes | No | 3A, 3B, 8A, 8B |
| LVPECL | All | | Off chip | 2.5V | Not supported | Yes | No | 3A, 3B, 8A, 8B |
| 1.2V, 1.5V, 3.3V PCML | All | | Off chip | 2.5V | Not supported | Yes | No | 3A, 3B, 8A, 8B |
| | All | | Off chip | 2.5V | Not supported | Yes | No | 3A, 3B, 8A, 8B |
| | All | | Off chip | 2.5V | Not supported | Yes | No | 3A, 3B, 8A, 8B |
| HCSL | PCIe | Differential DC | Off chip | 2.5V | Not supported | Yes | No | 3A, 3B, 8A, 8B |

**Notes to Table 6–10:**

(1) The EP4CGX15, EP4CGX22, and EP4CGX30 devices have two pairs of dedicated clock input pins in banks 3A and 8A for HSSI input reference clock. I/O banks 3B and 8B are not available in EP4CGX15, EP4CGX22, and EP4CGX30 devices.

(2) The EP4CGX50, EP4CGX75, EP4CGX110, and EP4CGX150 devices have four pairs of dedicated clock input pins in banks 3A, 3B, 8A, and 8B for HSSI input or single-ended clock input.

For more information about the AC-coupled termination scheme for the HSSI reference clock, refer to the *Cyclone IV Transceivers Architecture* chapter.

## LVDS I/O Standard Support in Cyclone IV Devices

The LVDS I/O standard is a high-speed, low-voltage swing, low power, and GPIO interface standard. Cyclone IV devices meet the ANSI/TIA/EIA-644 standard with the following exceptions:

- The maximum differential output voltage ($V_{OD}$) is increased to 600 mV. The maximum $V_{OD}$ for ANSI specification is 450 mV.
- The input voltage range is reduced to the range of 1.0 V to 1.6 V, 0.5 V to 1.85 V, or 0 V to 1.8 V based on different frequency ranges. The ANSI/TIA/EIA-644 specification supports an input voltage range of 0 V to 2.4 V.

For LVDS I/O standard electrical specifications in Cyclone IV devices, refer to the *Cyclone IV Device Datasheet* chapter.

### Designing with LVDS

Cyclone IV I/O banks support the LVDS I/O standard. The Cyclone IV GX right I/O banks support true LVDS transmitters while the Cyclone IV E left and right I/O banks support true LVDS transmitters. On the top and bottom I/O banks, the emulated LVDS transmitters are supported using two single-ended output buffers with external resistors. One of the single-ended output buffers is programmed to have opposite polarity. The LVDS receiver requires an external 100-Ω termination resistor between the two signals at the input buffer.

Figure 6–12 shows a point-to-point LVDS interface using Cyclone IV devices true LVDS output and input buffers.

**Figure 6–12. Cyclone IV Devices LVDS Interface with True Output Buffer on the Right I/O Banks**

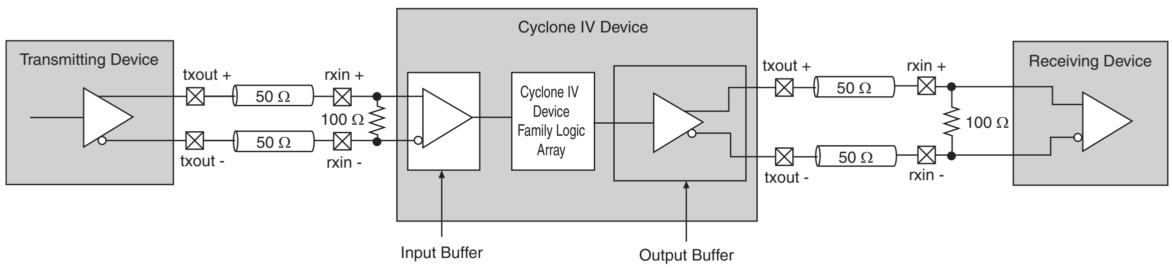
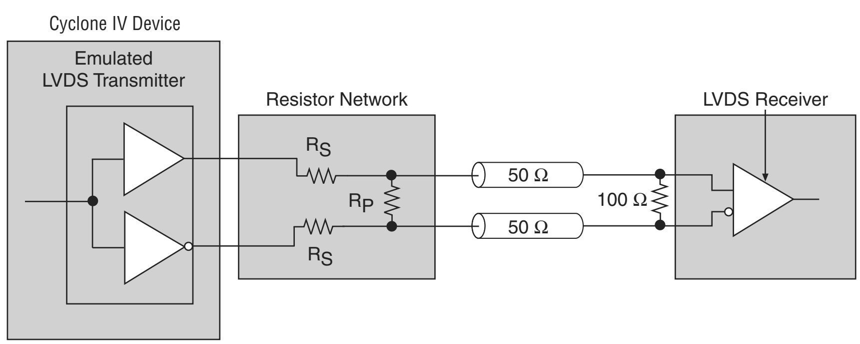



Figure 6–13 shows a point-to-point LVDS interface with Cyclone IV devices LVDS using two single-ended output buffers and external resistors.

**Figure 6–13. LVDS Interface with External Resistor Network on the Top and Bottom I/O Banks** *(1)*




**Note to Figure 6–13:**

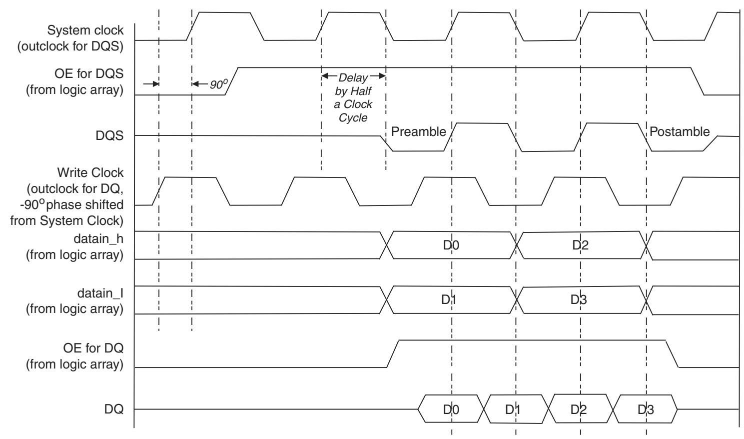
(1) $R_S$ = 120 Ω. $R_P$ = 170 Ω.

## BLVDS I/O Standard Support in Cyclone IV Devices

The BLVDS I/O standard is a high-speed differential data transmission technology that extends the benefits of standard point-to-point LVDS to multipoint configuration that supports bidirectional half-duplex communication. BLVDS differs from standard LVDS by providing a higher drive to achieve similar signal swings at the receiver while loaded with two terminations at both ends of the bus.

Figure 7–9 illustrates how the second output enable register extends the `DQS` high-impedance state by half a clock cycle during a write operation.

**Figure 7–9. Extending the OE Disable by Half a Clock Cycle for a Write Transaction** *(1)*




**Note to Figure 7–9:**

(1) The waveform reflects the software simulation result. The `OE` signal is an active low on the device. However, the Quartus II software implements the signal as an active high and automatically adds an inverter before the $A_{OE}$ register D input.

# OCT with Calibration

Cyclone IV devices support calibrated on-chip series termination ($R_S$ OCT) in both vertical and horizontal I/O banks. To use the calibrated OCT, you must use the RUP and RDN pins for each $R_S$ OCT control block (one for each side). You can use each OCT calibration block to calibrate one type of termination with the same $V_{CCIO}$ for that given side.

For more information about the Cyclone IV devices OCT calibration block, refer to the *Cyclone IV Device I/O Features* chapter.

# PLL

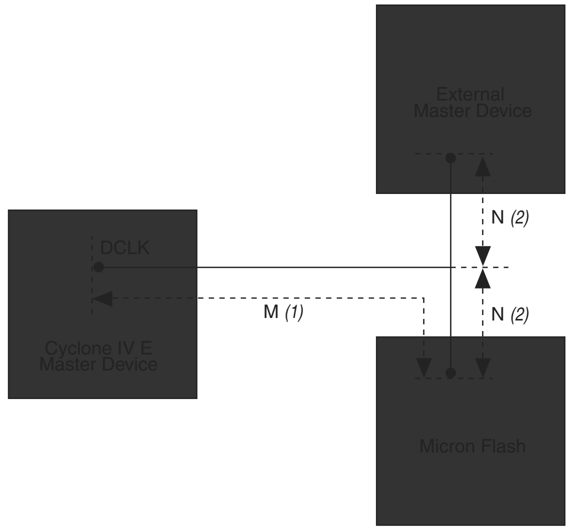
When interfacing with external memory, the PLL is used to generate the memory system clock, the write clock, the capture clock and the logic-core clock. The system clock generates the `DQS` write signals, commands, and addresses. The write-clock is shifted by -90° from the system clock and generates the `DQ` signals during writes. You can use the PLL reconfiguration feature to calibrate the read-capture phase shift to balance the setup and hold margins.

The PLL is instantiated in the ALTMEMPHY megafunction. All outputs of the PLL are used when the ALTMEMPHY megafunction is instantiated to interface with external memories. PLL reconfiguration is used in the ALTMEMPHY megafunction to calibrate and track the read-capture phase to maintain the optimum margin.

For more information about usage of PLL outputs by the ALTMEMPHY megafunction, refer to the *External Memory Interface Handbook*.

Figure 8–11 shows the recommended balanced star routing for multiple bus master interfaces to minimize signal integrity issues.

**Figure 8–11. Balanced Star Routing**




**Notes to Figure 8–11:**

(1) Altera recommends that *M* does not exceed 6 inches, as listed in Table 8–11 on page 8–28.

(2) Altera recommends using a balanced star routing. Keep the *N* length equal and as short as possible to minimize reflection noise from the transmission line. The *M* length is applicable for this setup.

## Estimating AP Configuration Time

AP configuration time is dominated by the time it takes to transfer data from the parallel flash to Cyclone IV E devices. This parallel interface is clocked by the Cyclone IV E `DCLK` output (generated from an internal oscillator). The `DCLK` minimum frequency when using the 40-MHz oscillator is 20 MHz (50 ns). In word-wide cascade programming, the `DATA[15..0]` bus transfers a 16-bit word and essentially cuts configuration time to approximately 1/16 of the AS configuration time. Equation 8–4 and Equation 8–5 show the configuration time calculations.

**Equation 8–4.**

$$\text{Size} \times \left(\frac{\text{maximum DCLK period}}{\text{16 bits per DCLK cycle}}\right) = \text{estimated maximum configuration time}$$

**Equation 8–5.**

$$9{,}600{,}000 \text{ bits} \times \left(\frac{50 \text{ ns}}{16 \text{ bit}}\right) = 30 \text{ ms}$$

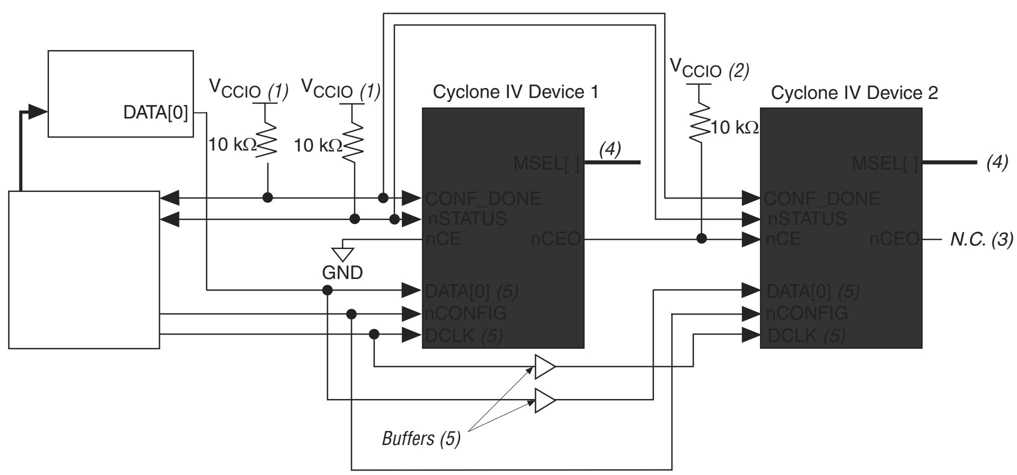
To ensure DCLK and DATA[0] are not left floating at the end of configuration, the MAX II device must drive them either high or low, whichever is convenient on your board. The DATA[0] pin is available as a user I/O pin after configuration. In the PS scheme, the DATA[0] pin is tri-stated by default in 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.

#### Reconfiguration

After the configuration data is successfully written into the serial configuration device, the Cyclone IV device does not automatically start reconfiguration. The intelligent host issues the `PULSE_NCONFIG` JTAG instruction to initialize the reconfiguration process. During reconfiguration, the master device is reset and the SFL design no longer exists in the Cyclone IV device and the serial configuration device configures all the devices in the chain with the user design.

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

### JTAG Instructions

For more information about the JTAG binary instruction code, refer to the *JTAG Boundary-Scan Testing for Cyclone IV Devices* chapter.

#### I/O Reconfiguration

Use the `CONFIG_IO` instruction to reconfigure the I/O configuration shift register (IOCSR) chain. This instruction allows you to perform board-level testing prior to configuring the Cyclone IV device or waiting for a configuration device to complete configuration. After the configuration is interrupted and JTAG testing is complete, you must reconfigure the part through the `PULSE_NCONFIG` JTAG instruction or by pulsing the `nCONFIG` pin low.

You can issue the `CONFIG_IO` instruction any time during user mode.

You must meet the following timing restrictions when using the `CONFIG_IO` instruction:

- The `CONFIG_IO` instruction cannot be issued when the `nCONFIG` pin is low
- You must observe a 230 µs minimum wait time after any of the following conditions:
  - `nCONFIG` pin goes high
  - Issuing the `PULSE_NCONFIG` instruction
  - Issuing the `ACTIVE_ENGAGE` instruction, before issuing the `CONFIG_IO` instruction
- You must wait 230 µs after power up, with the `nCONFIG` pin high before issuing the `CONFIG_IO` instruction (or wait for the `nSTATUS` pin to go high)

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.

Actual lock time depends on the transition density of the incoming data and the ppm difference between the receiver input reference clock and the upstream transmitter reference clock.

Transition from the LTD state to the LTR state occurs when either of the following conditions is met:

- Signal detection circuitry indicates the absence 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 not within the configured ppm frequency threshold setting with respect to CDR clocks from multipurpose PLLs.

In automatic lock mode, the switch from LTR to LTD states is indicated by the assertion of the `rx_freqlocked` signal and the switch from LTD to LTR states indicated by the de-assertion of the `rx_freqlocked` signal.

### Manual Lock Mode

State transitions are controlled manually by using `rx_locktorefclk` and `rx_locktodata` ports. The LTR/LTD controller sets the CDR state depending on the logic level on the `rx_locktorefclk` and `rx_locktodata` ports. This mode provides the flexibility to control the CDR for a reduced lock time compared to the automatic lock mode. In automatic lock mode, the LTR/LTD controller relies on the ppm detector and the phase relationship detector to set the CDR in LTR or LTD mode. The ppm detector and phase relationship detector reaction times can be too long for some applications that require faster CDR lock time.

In manual lock mode, the `rx_freqlocked` signal is asserted when the CDR is in LTD state and de-asserted when CDR is in LTR state. For descriptions of `rx_locktorefclk` and `rx_locktodata` port controls, refer to Table 1–27 on page 1–87.

If you do not enable the optional `rx_locktorefclk` and `rx_locktodata` ports, the Quartus II software automatically configures the LTR/LTD controller in automatic lock mode.

The recommended transceiver reset sequence varies depending on the CDR lock mode. For more information about the reset sequence recommendations, refer to the *Reset Control and Power Down for Cyclone IV GX Devices* chapter.

## Deserializer

The deserializer converts received serial data from the receiver input buffer to parallel 8- or 10-bit data. Serial data is assumed to be received from the LSB to the MSB. The deserializer operates with the high-speed recovered clock from the CDR with the frequency at half of the serial data rate.

## Bonded Channel Configuration

In bonded channel configuration, the low-speed clock for the bonded channels share a common bonded clock path that reduces clock skew between the bonded channels. The phase compensation FIFOs in bonded channels share a set of pointers and control logic that results in equal FIFO latency between the bonded channels. These features collectively result in lower channel-to-channel skew when implementing multi-channel serial interface in bonded channel configuration.

In a transceiver block, the high-speed clock for each bonded channels is distributed independently from one of the two multipurpose PLLs directly adjacent to the block. The low-speed clock for bonded channels is distributed from a common bonded clock path that selects from one of the two multipurpose PLLs directly adjacent to the block. Transceiver channels for devices in F484 and larger packages support additional clocking flexibility for ×2 bonded channels. In these packages, the ×2 bonded channels support high-speed and low-speed bonded clock distribution from PLLs beyond the two multipurpose PLLs directly adjacent to the block. Table 1–10 lists the high- and low-speed clock sources for the bonded channels.

**Table 1–10. High- and Low-Speed Clock Sources for Bonded Channels in Bonded Channel Configuration**

| Package | Transceiver Block | Bonded Channels | High- and Low-Speed Clocks Source | |
|---|---|---|---|---|
| | | | Option 1 | Option 2 |
| F324 and smaller | GXBL0 | ×2 in channels 0, 1<br>×4 in all channels | MPLL_1 | MPLL_2 |
| F484 and larger | GXBL0 | ×2 in channels 0, 1 | MPLL_5/<br>GPLL_1 | MPLL_6 |
| | | ×4 in all channels | MPLL_5 | MPLL_6 |
| | GXBL1 (1) | ×2 in channels 0, 1 | MPLL_7/<br>MPLL_6 | MPLL_8 |
| | | ×4 in all channels | MPLL_7 | MPLL_8 |

**Note to Table 1–10:**

(1) GXBL1 is not available for transceivers in F484 package.

☞ When implementing ×2 bonded channel configuration in a transceiver block, remaining channels 2 and 3 are available to implement other non-bonded channel configuration.

In configuration with rate match FIFO, the transmitter datapath clocking is identical to Transmitter Only operation as shown in Figure 1–38. In each bonded receiver channel, the CDR unit recovers the clock from serial received data and generates the high- and low-speed recovered clock for each bonded channel. The high-speed recovered clock feeds the channel's deserializer, and low-speed recovered clock is forwarded to receiver PCS. The individual low-speed recovered clock feeds to the following blocks in the receiver PCS:

- word aligner
- write clock of rate match FIFO

The common bonded low-speed clock that is used in all bonded transmitter PCS datapaths feeds the following blocks in each bonded receiver PCS:

- read clock of rate match FIFO
- 8B/10B decoder
- write clock of byte deserializer
- byte ordering
- write clock of RX phase compensation FIFO

When the byte deserializer is enabled, the common bonded low-speed clock frequency is halved before feeding to the write clock of RX phase compensation FIFO. The common bonded low-speed clock is available in FPGA fabric as `coreclkout` port, which can be used in FPGA fabric to send transmitter data and control signals, and capture receiver data and status signals from the bonded channels.

## FPGA Fabric-Transceiver Interface Clocking

The FPGA fabric-transceiver interface clocks consists of clock signals from the FPGA fabric to the transceiver blocks, and from the transceiver blocks to the FPGA fabric. These clock resources use the global clock networks (GCLK) in the FPGA core.

For information about the GCLK resources in the Cyclone IV GX devices, refer to *Clock Networks and PLLs in Cyclone IV Devices* chapter.

Table 1–11 lists the FPGA fabric-transceiver interface clocks.

**Table 1–11. FPGA Fabric-Transceiver Interface Clocks (Part 1 of 2)**

| Clock Name | Clock Description | Interface Direction |
|---|---|---|
| tx_clkout | Phase compensation FIFO clock | Transceiver to FPGA fabric |
| rx_clkout | Phase compensation FIFO clock | Transceiver to FPGA fabric |
| coreclkout | Phase compensation FIFO clock | Transceiver to FPGA fabric |
| fixed_clk | 125MHz receiver detect clock in PIPE mode | FPGA fabric to transceiver |
| reconfig_clk (1), (2) | Transceiver dynamic reconfiguration and offset cancellation clock | FPGA fabric to transceiver |

## Signal Detect at Receiver

In PIPE mode, signal detection is supported with the built-in signal threshold detection circuitry. When electrical idle inference is not enabled, the `rx_signaldetect` signal is inverted and available as `pipeelecidle` port in the PIPE interface.

## Lane Synchronization

In PIPE mode, the word aligner is configured in automatic synchronization state machine mode that complies with the PCIe specification. Table 1–16 lists the synchronization state machine parameters that implement the PCIe-compliant synchronization.

**Table 1–16. Synchronization State Machine Parameters (1)**

| Parameter | Value |
|---|---|
| Number of valid synchronization (/K28.5/) code groups received to achieve synchronization | 4 |
| Number of erroneous code groups received to lose synchronization | 17 |
| Number of continuous good code groups received to reduce the error count by one | 16 |

**Note to Table 1–16:**

(1) The word aligner supports 10-bit pattern lengths in PIPE mode.

## Clock Rate Compensation

In PIPE mode, the rate match FIFO compensates up to ±300 ppm (600 ppm total) difference between the upstream transmitter and the local receiver reference clock. In PIPE mode, the rate match FIFO operation is compliant to the version 2.0 of the PCIe Base Specification. The PCIe protocol requires the receiver to recognize a skip (`SKP`) ordered set, and inserts or deletes only one `SKP` symbol per `SKP` ordered set received to prevent the rate match FIFO from overflowing or underflowing. The `SKP` ordered set is a /K28.5/ comma (`COM`) symbol followed by one to five consecutive /K28.0/ `SKP` symbols, which are sent by transmitter during the inter-packet gap.

The rate match operation begins after the synchronization state machine in the word aligner indicates synchronization is acquired, as indicated with logic high on `rx_syncstatus` signal. Rate match FIFO insertion and deletion events are communicated to FPGA fabric on the `pipestatus[2..0]` port from each channel.

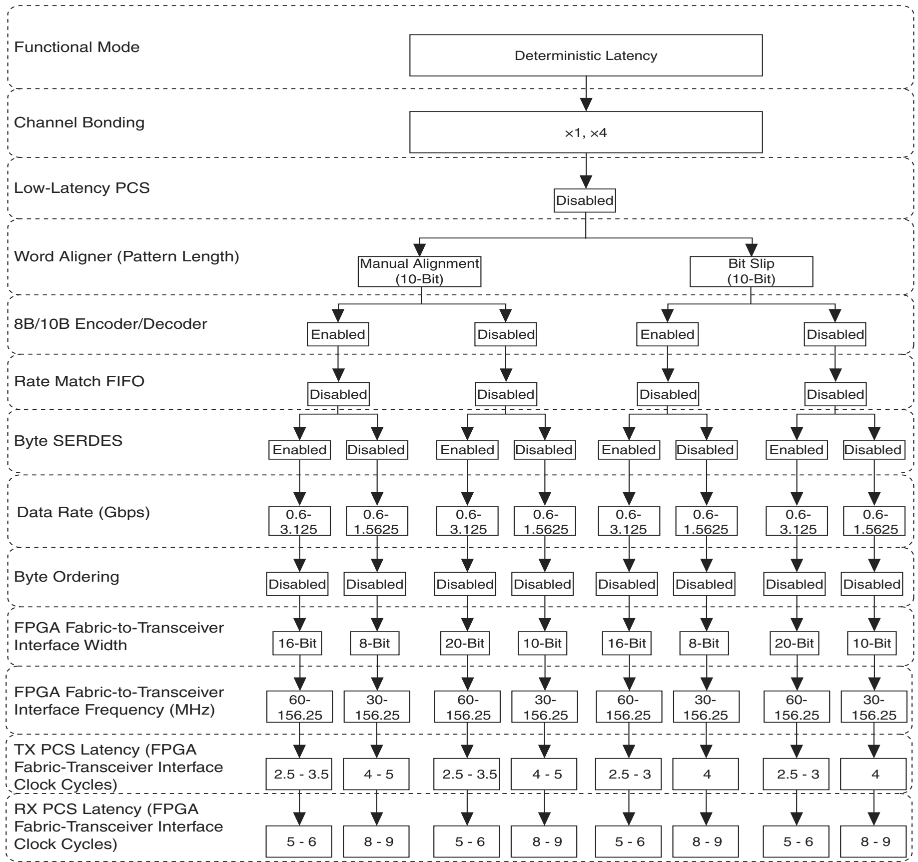
## Low-Latency Synchronous PCIe

In PIPE mode, the Cyclone IV GX transceiver supports a lower latency in synchronous PCIe by reducing the latency across the rate match FIFO. In synchronous PCIe, the system uses a common reference clocking that gives a 0 ppm difference between the upstream transmitter's and local receiver's reference clock.

When using common reference clocking, the transceiver supports spread-spectrum clocking. For more information about the SSC support in PCIe Express (PIPE) mode, refer to the *Cyclone IV Device Data Sheet*.

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*.

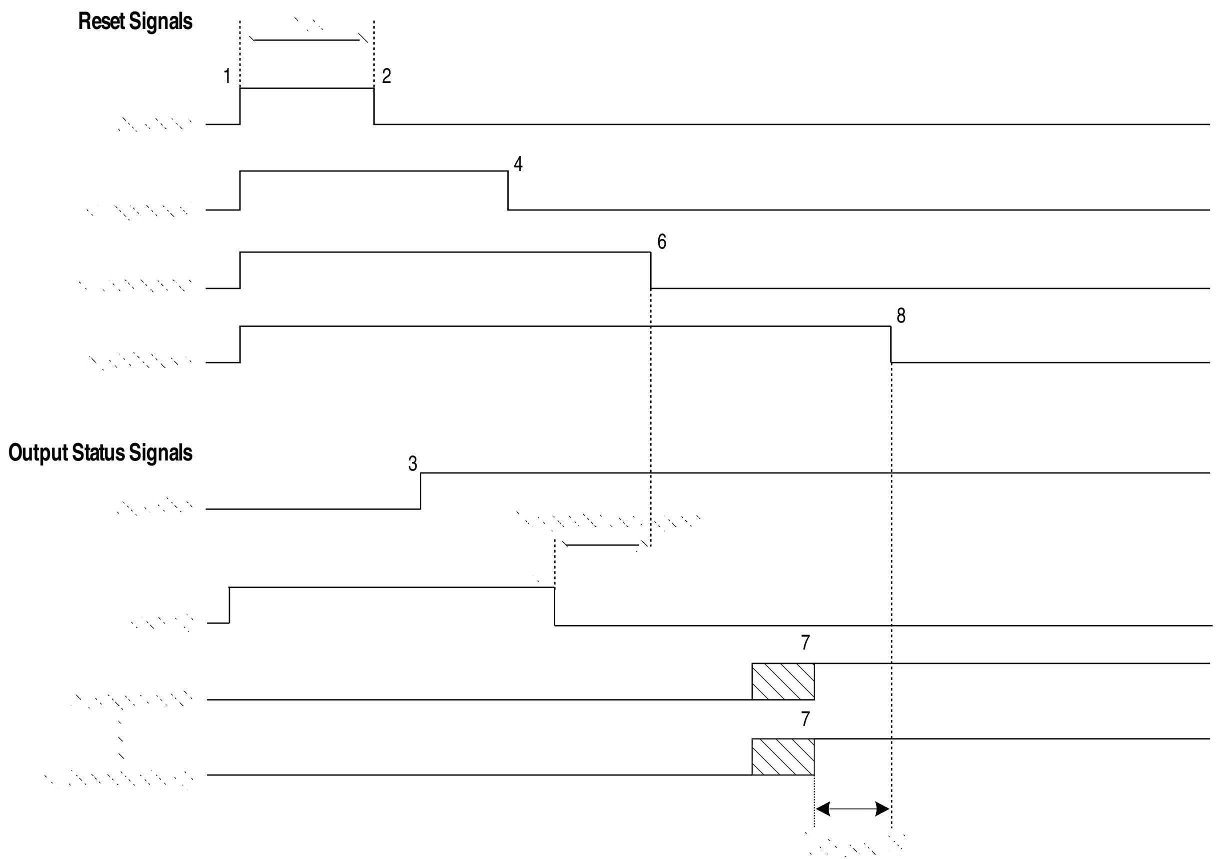
## Registered Mode Phase Compensation FIFO

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.

### 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.

Table 3–5 describes the rx_dataoutfull[31..0] FPGA fabric-Transceiver channel interface signals.

**Table 3–5. rx_dataoutfull[31..0] FPGA Fabric-Transceiver Channel Interface Signal Descriptions (Part 1 of 3)**

| FPGA Fabric-Transceiver Channel Interface Description | Receive Signal Description (Based on Cyclone IV GX Supported FPGA Fabric-Transceiver Channel Interface Widths) |
|---|---|
| 8-bit FPGA fabric-Transceiver Channel Interface | **The following signals are used in 8-bit 8B/10B modes:** |
| | rx_dataoutfull[7:0]: 8-bit decoded data (rx_dataout) |
| | rx_dataoutfull[8]: Control bit (rx_ctrldetect) |
| | rx_dataoutfull[9]: Code violation status signal (rx_errdetect) |
| | rx_dataoutfull[10]: rx_syncstatus |
| | rx_dataoutfull[11]: Disparity error status signal (rx_disperr) |
| | rx_dataoutfull[12]: Pattern detect status signal (rx_patterndetect) |
| | rx_dataoutfull[13]: Rate Match FIFO deletion status indicator (rx_rmfifodatadeleted) in non-PCI Express (PIPE) functional modes. |
| | rx_dataoutfull[14]: Rate Match FIFO insertion status indicator (rx_rmfifodatainserted) in non-PCI Express (PIPE) functional modes. |
| | rx_dataoutfull[14:13]: PCI Express (PIPE) functional mode (rx_pipestatus) |
| | rx_dataoutfull[15]: 8B/10B running disparity indicator (rx_runningdisp) |
| 10-bit FPGA fabric-Transceiver Channel Interface | rx_dataoutfull[9:0]: 10-bit un-encoded data (rx_dataout) |
| | rx_dataoutfull[10]: rx_syncstatus |
| | rx_dataoutfull[11]: 8B/10B disparity error indicator (rx_disperr) |
| | rx_dataoutfull[12]: rx_patterndetect |
| | rx_dataoutfull[13]: Rate Match FIFO deletion status indicator (rx_rmfifodatadeleted) in non-PCI Express (PIPE) functional modes |
| | rx_dataoutfull[14]: Rate Match FIFO insertion status indicator (rx_rmfifodatainserted) in non-PCI Express (PIPE) functional modes |
| | rx_dataoutfull[15]: 8B/10B running disparity indicator (rx_runningdisp) |

# 1. Cyclone IV Device Datasheet

CYIV-53001-2.1

This chapter describes the electrical and switching characteristics for Cyclone® IV devices. Electrical characteristics include operating conditions and power consumption. Switching characteristics include transceiver specifications, core, and periphery performance. This chapter also describes I/O timing, including programmable I/O element (IOE) delay and programmable output buffer delay.

This chapter includes the following sections:

- "Operating Conditions" on page 1–1
- "Power Consumption" on page 1–16
- "Switching Characteristics" on page 1–16
- "I/O Timing" on page 1–37
- "Glossary" on page 1–37

## Operating Conditions

When Cyclone IV devices are implemented in a system, they are rated according to a set of defined parameters. To maintain the highest possible performance and reliability of Cyclone IV devices, you must consider the operating requirements described in this chapter.

Cyclone IV devices are offered in commercial, industrial, extended industrial and, automotive grades. Cyclone IV E devices offer –6 (fastest), –7, –8, –8L, and –9L speed grades for commercial devices, –8L speed grades for industrial devices, and –7 speed grade for extended industrial and automotive devices. Cyclone IV GX devices offer –6 (fastest), –7, and –8 speed grades for commercial devices and –7 speed grade for industrial devices.

For more information about the supported speed grades for respective Cyclone IV devices, refer to the *Cyclone IV FPGA Device Family Overview* chapter.

Cyclone IV E devices are offered in core voltages of 1.0 and 1.2 V. Cyclone IV E devices with a core voltage of 1.0 V have an 'L' prefix attached to the speed grade.

In this chapter, a prefix associated with the operating temperature range is attached to the speed grades; commercial with a "C" prefix, industrial with an "I" prefix, and automotive with an "A" prefix. Therefore, commercial devices are indicated as C6, C7, C8, C8L, or C9L per respective speed grade. Industrial devices are indicated as I7, I8, or I8L. Automotive devices are indicated as A7.

**Table 1–31. RSDS Transmitter Timing Specifications for Cyclone IV Devices (1), (2), (4) (Part 2 of 2)**

| Symbol | Modes | C6 | | | C7, I7 | | | C8, A7 | | | C8L, I8L | | | C9L | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | |
| $t_{LOCK}$ (3) | — | — | — | 1 | — | — | 1 | — | — | 1 | — | — | 1 | — | — | 1 | ms |

**Notes to Table 1–31:**

(1) Applicable for true RSDS and emulated RSDS_E_3R transmitter.

(2) Cyclone IV E devices—true RSDS transmitter is only supported at the output pin of Row I/O Banks 1, 2, 5, and 6. Emulated RSDS transmitter is supported at the output pin of all I/O Banks.
Cyclone IV GX devices—true RSDS transmitter is only supported at the output pin of Row I/O Banks 5 and 6. Emulated RSDS transmitter is supported at the output pin of I/O Banks 3, 4, 5, 6, 7, 8, and 9.

(3) $t_{LOCK}$ is the time required for the PLL to lock from the end-of-device configuration.

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

**Table 1–32. Emulated RSDS_E_1R Transmitter Timing Specifications for Cyclone IV Devices (1), (3) (Part 1 of 2)**

| Symbol | Modes | C6 | | | C7, I7 | | | C8, A7 | | | C8L, I8L | | | C9L | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | |
| $f_{HSCLK}$ (input clock frequency) | ×10 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 72.5 | MHz |
| | ×8 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 72.5 | MHz |
| | ×7 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 72.5 | MHz |
| | ×4 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 72.5 | MHz |
| | ×2 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 85 | 5 | — | 72.5 | MHz |
| | ×1 | 5 | — | 170 | 5 | — | 170 | 5 | — | 170 | 5 | — | 170 | 5 | — | 145 | MHz |
| Device operation in Mbps | ×10 | 100 | — | 170 | 100 | — | 170 | 100 | — | 170 | 100 | — | 170 | 100 | — | 145 | Mbps |
| | ×8 | 80 | — | 170 | 80 | — | 170 | 80 | — | 170 | 80 | — | 170 | 80 | — | 145 | Mbps |
| | ×7 | 70 | — | 170 | 70 | — | 170 | 70 | — | 170 | 70 | — | 170 | 70 | — | 145 | Mbps |
| | ×4 | 40 | — | 170 | 40 | — | 170 | 40 | — | 170 | 40 | — | 170 | 40 | — | 145 | Mbps |
| | ×2 | 20 | — | 170 | 20 | — | 170 | 20 | — | 170 | 20 | — | 170 | 20 | — | 145 | Mbps |
| | ×1 | 10 | — | 170 | 10 | — | 170 | 10 | — | 170 | 10 | — | 170 | 10 | — | 145 | Mbps |
| $t_{DUTY}$ | — | 45 | — | 55 | 45 | — | 55 | 45 | — | 55 | 45 | — | 55 | 45 | — | 55 | % |
| TCCS | — | — | — | 200 | — | — | 200 | — | — | 200 | — | — | 200 | — | — | 200 | ps |
| Output jitter (peak to peak) | — | — | — | 500 | — | — | 500 | — | — | 550 | — | — | 600 | — | — | 700 | ps |
| $t_{RISE}$ | 20 – 80%, $C_{LOAD}$ = 5 pF | — | 500 | — | — | 500 | — | — | 500 | — | — | 500 | — | — | 500 | — | ps |
| $t_{FALL}$ | 20 – 80%, $C_{LOAD}$ = 5 pF | — | 500 | — | — | 500 | — | — | 500 | — | — | 500 | — | — | 500 | — | ps |

Table 1–44 and Table 1–45 list the IOE programmable delay for Cyclone IV GX devices.

**Table 1–44. IOE Programmable Delay on Column Pins for Cyclone IV GX Devices** (1), (2)

| Parameter | Paths Affected | Number of Settings | Min Offset | Max Offset | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Fast Corner | | Slow Corner | | | | |
| | | | | C6 | I7 | C6 | C7 | C8 | I7 | |
| Input delay from pin to internal cells | Pad to I/O dataout to core | 7 | 0 | 1.313 | 1.209 | 2.184 | 2.336 | 2.451 | 2.387 | ns |
| Input delay from pin to input register | Pad to I/O input register | 8 | 0 | 1.312 | 1.208 | 2.200 | 2.399 | 2.554 | 2.446 | ns |
| Delay from output register to output pin | I/O output register to pad | 2 | 0 | 0.438 | 0.404 | 0.751 | 0.825 | 0.886 | 0.839 | ns |
| Input delay from dual-purpose clock pin to fan-out destinations | Pad to global clock network | 12 | 0 | 0.713 | 0.682 | 1.228 | 1.41 | 1.566 | 1.424 | ns |

**Notes to Table 1–44:**

(1) The incremental values for the settings are generally linear. For exact values of each setting, use the latest version of the Quartus II software.

(2) The minimum and maximum offset timing numbers are in reference to setting **0** as available in the Quartus II software.

**Table 1–45. IOE Programmable Delay on Row Pins for Cyclone IV GX Devices** (1), (2)

| Parameter | Paths Affected | Number of Settings | Min Offset | Max Offset | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Fast Corner | | Slow Corner | | | | |
| | | | | C6 | I7 | C6 | C7 | C8 | I7 | |
| Input delay from pin to internal cells | Pad to I/O dataout to core | 7 | 0 | 1.314 | 1.210 | 2.209 | 2.398 | 2.526 | 2.443 | ns |
| Input delay from pin to input register | Pad to I/O input register | 8 | 0 | 1.313 | 1.208 | 2.205 | 2.406 | 2.563 | 2.450 | ns |
| Delay from output register to output pin | I/O output register to pad | 2 | 0 | 0.461 | 0.421 | 0.789 | 0.869 | 0.933 | 0.884 | ns |
| Input delay from dual-purpose clock pin to fan-out destinations | Pad to global clock network | 12 | 0 | 0.712 | 0.682 | 1.225 | 1.407 | 1.562 | 1.421 | ns |

**Notes to Table 1–45:**

(1) The incremental values for the settings are generally linear. For exact values of each setting, use the latest version of Quartus II software.

(2) The minimum and maximum offset timing numbers are in reference to setting **0** as available in the Quartus II software