AN 479: Design Guidelines for Implementing LVDS Interfaces in Cyclone Series Devices

© July 2013 AN-479-1.2

Introduction

This application note describes the methods to use Cyclone® series (Cyclone III, Cyclone III LS, Cyclone II, and Cyclone) devices for high-performance LVDS interfaces. LVDS is a signaling standard that provides high-speed data transfers. Cyclone series devices offer easy integration of LVDS interfaces at speeds up to 875 Mbps for the receiver and 840 Mbps for the transmitter. This application note also includes step-by-step design flow and interface guidelines. With simple settings and considerations using the ALTLVDS megafunction in the Quartus® II software, Cyclone series devices become part of the high-speed setup in your system, with increased system integration at reduced cost.

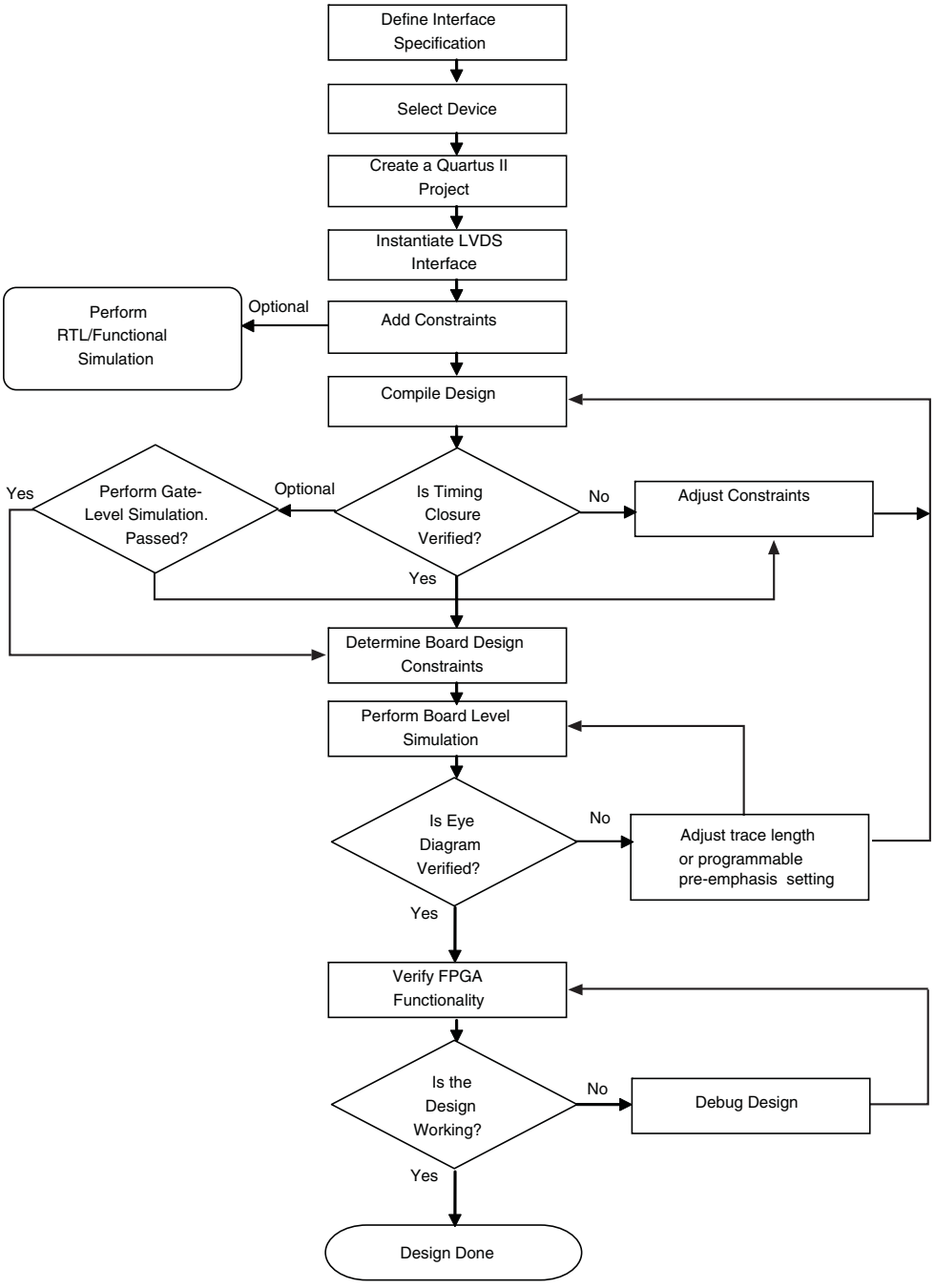
Use this application note with the following chapters from the respective device family handbooks:

- *High-Speed Differential Interfaces in Cyclone III Devices* chapter in volume 1 of the *Cyclone III Device Handbook*
- *High-Speed Differential Interfaces in Cyclone II Devices* chapter in volume 1 of the *Cyclone II Device Handbook*
- *High-Speed Differential Signaling in Cyclone Devices* chapter in volume 1 of the *Cyclone Device Handbook*

FPGA Design Flow

This section describes the design flow to implement an LVDS interface in a Cyclone series device using the ALTLVDS megafunction as shown in Figure 1. The following subsections detail the step-by-step design flow and simple guidelines to integrate an LVDS interface block to a Cyclone series device.

Figure 1. Design Flow for Implementing LVDS Interfaces in Cyclone Series Devices



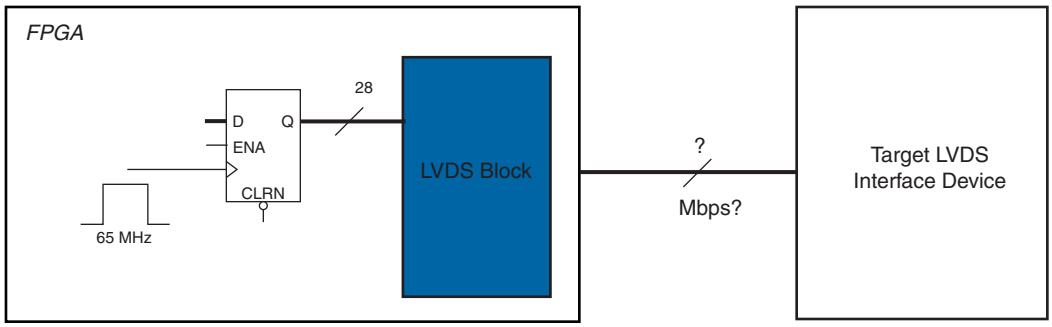
Step 1: Define Interface Specification

Properly defining the interface requirement helps you to allocate and optimize the resources used. At the first stage of integrating the LVDS interface, you can identify the number of channels required and the speed of data transmission. This determines the number of physical pairs of differential traces (or connections) that are required between your FPGA and the target device.

Begin by identifying the parallel data width and the frequency of the bus that you want to implement as an LVDS interface. You must determine the number of required channels and the deserialization factor to use. The ALTLVDS megafunction supports ×1 to ×10 deserialization factors, with the exception of ×3. Higher deserialization factor translates to less channels and faster data rates (in bits per second per channel).

Figure 2 shows a 28-bit transmit bus running at 65 MHz as an example to determine the suitable number of channels and the deserialization factor.

Figure 2. Example of a LVDS Interface Setup



You must determine the suitable deserialization factor for the system. Divide the data width (28) with each available deserialization factors (from ×1 to ×10). The deserialization factors that yield zero remainder are suitable for the system.

Calculate the number of channels required for each selected deserialization factor. This value is the result of the data width divided with the associated deserialization factor.

Multiply the bus frequency (65 MHz) with the deserialization factor to obtain the data-rates per channel.

Table 1 shows the deserialization factor that yields zero remainder (×1, ×2, ×4, and ×7).

Table 1. Deserialization Factor, Number of Channels, and Data Rates per Channel Combinations *(Note 1)*, *(2)*

Deserialization Factor	Number of Channels *(1)*, *(2)*	Data Rates per Channel (Mbps)
×1	28	65
×2	14	130
×4	7	260
×7	4	455

Notes to Table 1:

(1) Each channel uses a pair of pins; one pin each for inverted (n) and non-inverted (p) signal.

(2) For ×2 and higher deserialization factors, LVDS transmission employs double-data rate, in which the data is clocked at rising and falling edges. Data rate is doubled without increasing the number of channels.

For optimum resource utilization and performance, select the deserialization factor that yields the least number of channels and the highest data rate that can be supported. As shown in the Table 1, ×7 with four channels at 455 Mbps is the best choice because it only requires four pairs of data outputs for a 28-bit bus chip-to-chip transmission.

Step 2: Select Device

You must identify which FPGA to use after you have determined the interface specifications (deserialization factor, channels, and data rates). The following sections describe the four categories used for device selection in Cyclone series devices:

- "Timing Performance" on page 4
- "I/O Resources" on page 6
- "Phase-Locked Loop (PLL) Resources" on page 6
- "Clock Resources" on page 7

Timing Performance

Timing performance is often associated with data rate because it determines the number of data bits that can be transferred in one second. However, there are skew-related performance parameters that require similar attention in device selection. This section discusses the transmission performance options and the skew-related parameters consideration.

Altera® offers Cyclone series FPGAs, in various performance options that are differentiated by speed grade. For example, the commercial speed grade ranges in -6, -7 and -8 (slowest). Each device family and each speed grade supports a specific timing performance.

For more information on Cyclone III LS LVDS timing performance, refer to the *Cyclone III LS Device Data Sheet* chapter in volume 2 of the *Cyclone III Device Handbook*.

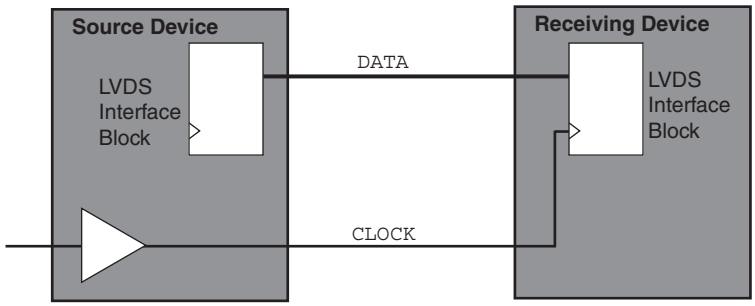
For more information on Cyclone III LVDS timing performance, refer to the *Cyclone III Device Data Sheet* chapter in volume 2 of the *Cyclone III Device Handbook*.

For more information on Cyclone II LVDS timing performance, refer to the *DC Characteristic and Timing Specifications* chapter in volume 1 of the *Cyclone II Device Handbook*.

For more information on Cyclone LVDS timing performance, refer to the *DC Characteristics and Timing Specifications* chapter in volume 1 of the *Cyclone Device Handbook*.

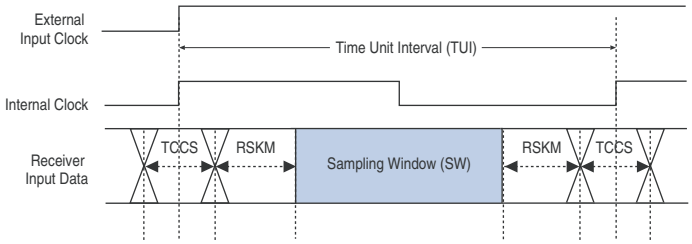
LVDS is a source-synchronous signaling interface in which the clock is sourced from the same device as the data rather than another source, such as a common clock network as shown in Figure 3.

Figure 3. Simplified Source-Synchronous Interface



Therefore, the LVDS interface timing is based on the skew between the data and clock signals. The skew-related parameters are transmitter channel-to-channel skew (TCCS) and sampling window (SW). Lower device-level skew values give you more timing margin to account for various board-level effects such as skew, jitter, and noise. The timing margin for the interface is represented by the receiver input skew margin (RSKM) as shown in Figure 4.

Figure 4. High-Speed I/O Timing Diagram



For more information on the timing parameters for Cyclone III family devices, refer to the *High-Speed I/O Timing in Cyclone III Devices* section in the *High-Speed Differential Interfaces in Cyclone III Devices* chapter in volume 1 of the *Cyclone III Device Handbook*.

For more information on the timing parameters for Cyclone II devices, refer to the *High-Speed I/O Timing in Cyclone II Devices* section in the *High-Speed Differential Interfaces in Cyclone II Devices* chapter in volume 1 of the Cyclone II Device Handbook.

For more information on the timing parameters for Cyclone devices, refer to *High-Speed I/O Timing in Cyclone Devices* section in the *High-Speed Differential Signaling in Cyclone Devices* chapter in volume 1 of the *Cyclone Device Handbook*.

I/O Resources

The number of differential channels varies across device family, density, and package. Use the device pin-outs to determine if the number of differential channels available in the FPGA meets your system requirements. Differential channels are identified by the `DIFFIO_xx` pins as **Optional Function(s)** in Cyclone III device family and `LVDSxx` as **Optional Function(s)** in Cyclone II and Cyclone devices. Requirements that affect the differential channels usage are:

- Bank V_{CCIO} level for LVDS inputs and outputs: LVDS input and output must be powered with a 2.5-V V_{CCIO} supply in the banks where it resides, except for the following conditions in which a 2.5-V V_{CCIO} supply is not required:
  - All LVDS inputs for the Cyclone device are powered by V_{CCINT}
  - All LVDS dedicated clock inputs for Cyclone II are powered by V_{CCINT}

Supplying higher than supported voltage to V_{CCIO} for LVDS violates the rated specifications and may damage the buffer.

- Differential pad placement guidelines: There are requirements for differential pad placement for single-ended I/O pads and total differential outputs per V_{CCIO} and ground pair. These requirements ensure an acceptable noise level on the V_{CCIO} supply in the banks where the differential I/Os reside. Only placement requirements of single-ended I/O pads to differential pads are automatically checked by the Quartus II software. Violating the guidelines may put your system at risk of functional failure due to undesirable noise form the signals.

For more information on the differential pad placement requirements for Cyclone III family devices, refer to the *Differential Pad Placement Guidelines* section in the *Cyclone III Device I/O Features* chapter in volume 1 of the *Cyclone III Device Handbook*.

For more information on the differential pad placement requirements for Cyclone II devices, refer to *Differential Pad Placement Guidelines* section in the *Selectable I/O Standards in Cyclone II Devices* chapter in volume 1 of the *Cyclone II Device Handbook*.

For more information on the differential pad placement requirements for Cyclone devices, refer to *Differential Pad Placement Guidelines* section in the *Using Selectable I/O Standards in Cyclone Devices* chapter in volume 1 of the *Cyclone Device Handbook*.

Phase-Locked Loop (PLL) Resources

The ALTLVDS megafunction uses a PLL to create various clocks for SERDES operations in the LVDS interface. You need at least one PLL for ×4 and higher deserialization factor operations.

The number of PLL varies according to device densities. There are a maximum of two PLLs available in Cyclone devices and four PLLs in Cyclone III family devices and Cyclone II devices.

For more information on the exact PLL resources for a specific device density for Cyclone III family devices, refer to the *Clock Networks and PLLs* chapter in volume 1 of the *Cyclone III Device Handbook*.

For more information on the exact PLL resources for a specific device density for Cyclone II devices, refer to the *PLLS in Cyclone II Devices* chapter in volume 1 of the *Cyclone II Device Handbook*.

For more information on the exact PLL resources for a specific device density for Cyclone devices, refer to the *Using PLLs in Cyclone Devices* chapter in volume 1 of the *Cyclone Device Handbook*.

Cyclone III family devices and Cyclone II devices support source-synchronous compensation mode PLL. Altera recommends using the source-synchronous compensation mode PLL for LVDS receiver applications to maintain the clock and data relationship at the pin.

Cyclone devices do not support source-synchronous compensation mode PLL. Therefore, use the normal compensation mode PLL in Cyclone devices for LVDS receiver applications. In such cases, you must ensure the correct phase settings in the PLL to account for the delay from the clock pin to the input data register clock port.

For more information on how to make the PLL phase settings for Cyclone devices (or Cyclone series devices that use normal compensation mode PLL) refer to "Step 6: Adjust Constraints" on page 15.

Clock Resources

One pair of differential dedicated clock input pins is required for the destination device. The clock output pins are used to route the received clock directly to the PLL. Cyclone III family devices and Cyclone II devices offer up to 16 pairs of dedicated differential clock inputs, while Cyclone devices offer up to four pairs.

Use general purpose differential channels on the source Cyclone series device to implement the differential clock output to maintain the skew between clock and data outputs.

Step 3: Create a Quartus II Project and Instantiate the Interface

Once you have selected the appropriate FPGA device, create a Quartus II project for the target FPGA device.

For a step-by-step guide on creating a Quartus II project, refer to the Tutorial in the Quartus II software.

There are two ways to instantiate the components for the LVDS interface:

- With ALTLVDS megafunction, which builds the SERDES functions and instantiates the PLL. You can make the necessary PLL settings in the MegaWizard® Plug-In Manager under the **Frequency/PLL Settings** and **Transmitter** settings tab. This option integrates the PLL into the LVDS block, therefore simplifying the clocking setup for the system. The drawback is the reduced flexibility in the usage of PLL as the PLL can only be used for that particular LVDS block instantiation and cannot be shared for other operations.
- With ALTLVDS megafunction with external PLL option, which only builds the SERDES functions. A notification window appears when you turn on the **Use External PLL** check box. Follow the required clock setting to the input ports listed in the notification window. You can create your own clocking source with ALTPLL megafunction. Use this option to optimize the usage of the PLL with other operations in the core.

Figure 5 shows the modules of a typical LVDS interface design using internal PLL.

Figure 5. Typical LVDS Interface Modules with ALTLVDS Megafunction using Internal PLL Option

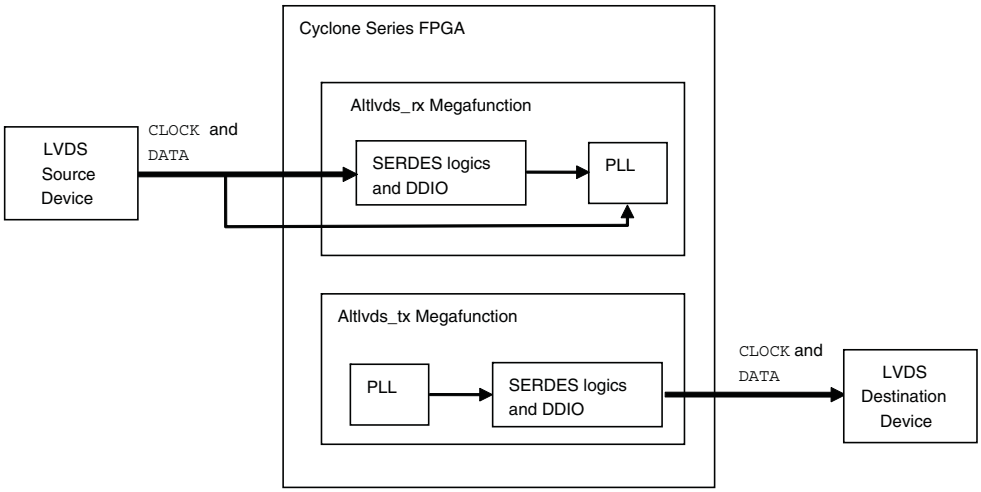
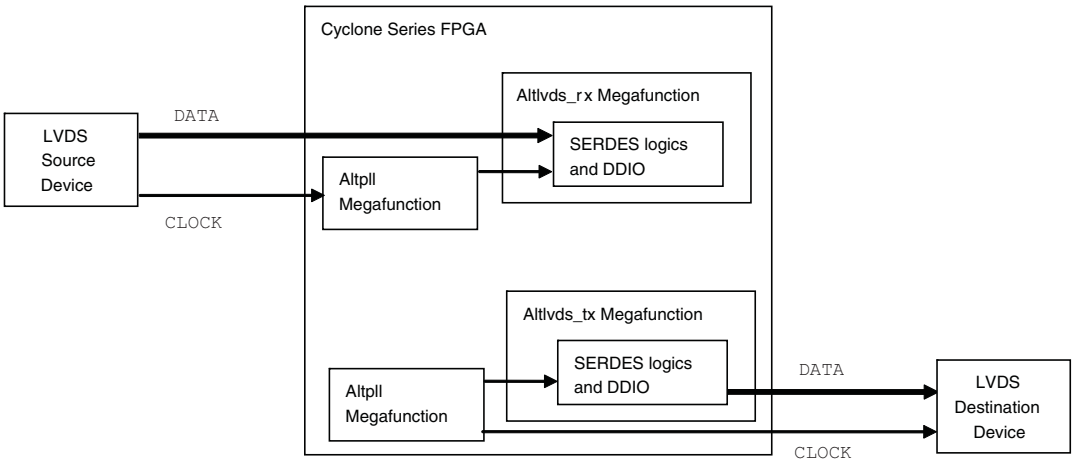


Figure 6 shows the modules of a typical LVDS interface design using internal and external PLL respectively.

Figure 6. Typical LVDS Interface Modules with ALTLVDS Megafunction using External PLL Option (with ALTPLL Megafunction)



You must consider some of the requirements when creating the settings in the ALTLVDS MegaWizard Plug-In Manager. This section describes the following:

- "Clock Settings" on page 9
- "PLL Sharing" on page 10
- "Odd SERDES Factor" on page 11
- "Data Word Alignment" on page 11

For more information on performing the settings in the ALTLVDS MegaWizard Plug-In Manager, refer to the *SERDES Transmitter/Receiver (ALTLVDS) Megafunction User Guide*.

Clock Settings

The ALTLVDS megafunction requires a PLL for ×4 and higher SERDES factors to generate a fast clock with a frequency at half the data rate. Cyclone series devices use the double data rate input/output (DDIO) registers as part of the SERDES interface (except for ×1 SERDES factor), hence the frequency is at half the data rate. There are various phase-settings available in the MegaWizard Plug-In Manager for input and output clock-to-data alignment.

The external PLL option allows you to access all the PLL clocks for other functions. However, you must ensure that all the clocking sources to the megafunction ports are in the correct frequency and phase setting.

Clock-to-data phase alignment is one of the main clock settings in the source and destination device. The clock is often set to edge-align with the data. However, you can also use the center-aligned and random phase-setting, depending on your source or destination device and board routing conditions. If you choose to implement the ALTLVDS megafunction which also instantiates the PLL, you can make such phase settings under the **Transmitter** settings or **Receiver** settings tab in the MegaWizard Plug-In Manager. Otherwise, you have to manually make the phase settings in the ALTPLL megafunction.

The clock is often centered in the data valid window at the destination device to maximize timing margin for data capture. There are a few recommended settings that you can make in the ALTLVDS megafunction of the receiver based on different clock-to-data alignment. Each alignment below refers to the clock-and-serial-data alignment.

- Edge-aligned: Turn on the **Use source-synchronous mode of the PLL** and **Align clock to center of data window at capture point** option. This option sets the receiver PLL to source-synchronous mode PLL and maintains the clock and data phase relationship at the pins to the input register. It also sets the phase of fast clock to -90° or -180° for double-data rate or single-data rate operations respectively.

  For more information on the source-synchronous mode for Cyclone III family devices, refer to the *Clock Networks and PLLs in Cyclone III Devices* chapter in volume 1 of the *Cyclone III Device Handbook*.

  For more information on the source-synchronous mode for Cyclone III family devices, refer to the *Source-Synchronous Mode* section in the *PLLs in Cyclone II Devices* chapter in volume 1 of the *Cyclone II Device Handbook*.

- Center-aligned: Turn on the **Use source-synchronous mode of the PLL** option. Incoming clock is aligned to the center of the data valid window, hence clock centering in the destination device is not required.
- Random-aligned: Manually determine the required phase-shift setting for the incoming clock of the destination device to correctly capture the data.

  For a Cyclone device which is used as an LVDS receiver, follow the guidelines for the random-aligned clock-to-data scenario.

PLL Sharing

In applications in which an LVDS transmitter and receiver are required, two PLLs are required to implement each of the interfaces. The ALTLVDS megafunction allows PLL sharing between the transmitter and receiver, thus reducing the number of PLL to one. Perform the following guidelines to implement this feature:

- To share a PLL, several PLLs must have the same PLL settings, such as PLL feedback mode, clock frequency, and phase settings. The LVDS transmitters and receivers must use the same input clock frequency, deserialization factor, and data rates.
- For transmitters, turn on the **Align clock to center of data window** option if you this option is enabled for your receiver.

- Turn on the **Use shared PLL(s) for receivers and transmitters** option to allow the Quartus II compiler to share the same PLL.

For more information on other PLL signals for PLL sharing, refer to the ALTLVDS Setting Options description and comments in the *SERDES Transmitter/Receiver (ALTLVDS) Megafunction User Guide*.

Odd SERDES Factor

Display applications often use an odd SERDES factor such as ×7 in LVDS interfaces. When designing with an odd SERDES factor, the Quartus II software by default sets the `txcoreclock` frequency to `data rate` per (`SERDES factor ×2`). This setting improves the timing performance in the SERDES logics and allows the device to operate at the maximum data rate. The ALTLVDS megafunction implements the odd SERDES factor interface differently than the even SERDES factor due to the odd number of data width being fed to the DDIO registers.

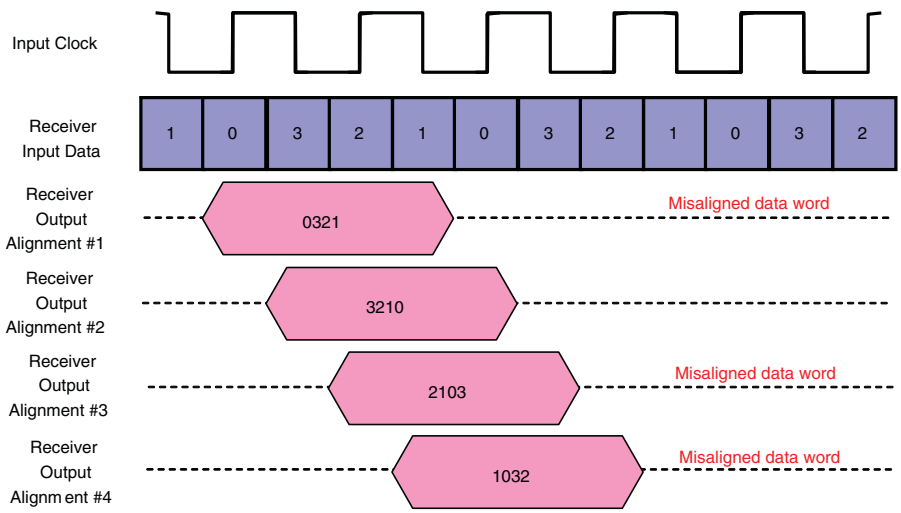
At certain lower data rates, you may choose a higher `txcoreclock` frequency (which is at `data rate per SERDES factor`). Higher `txcoreclock` frequency usage reduces logic element (LE) utilization by. Perform timing simulation and check the functionality of the interface to determine if your system can operate at a higher `txcoreclock` frequency to save resource utilization.

Data Word Alignment

In a parallel interface, each bit in a bus is transmitted over a separate pin simultaneously. If you want to transmit an 8-bit data bus over a parallel interface, the receiving device will receive the 8 bits of data in the same order that was transmitted; MSB to LSB in the correct data word boundary. In LVDS, the data bus is serialized and transmitted one bit at a time over one output (a pair of pins). With the same transmission of an 8-bit data bus but over LVDS interface, the receiving device will not be able to determine the correct data word boundary, not knowing the position of the MSB. The deserialized data may not be the same as the original 8-bit data that was transmitted earlier.

Figure 7 shows the possible data word alignment from a receiver for a 4-bit serial transmission. There is only a 25% possibility that the alignment is correct. Incorrect alignment corrupts the received data.

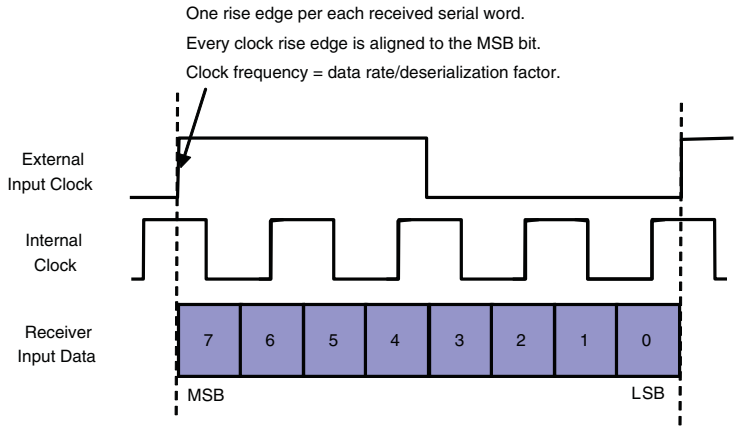
Figure 7. Four Possible Receiver Data Word Alignment Cases for a 4-bit Serial Transmission Interface



In Cyclone series devices, the PLL locks to the rising edge of the input clock. If you have one rising edge for each serial word received, the PLL locks to the same place within the serial word. Therefore, the easiest way for data word alignment in a Cyclone series device is to align the MSB bit to every rise edge of the clock during transmission from the source device. This is possible with the input clock frequency equal to the data bus frequency. In other words, if the input clock frequency is equal to the data rate per deserialization factor, data word alignment in a Cyclone series device used as a LVDS receiver is guaranteed without any additional setting or logic. In this setup, the word boundary at the Cyclone series receiver will be deterministic, regardless of power-up cycle or PLL reset event.

Figure 8 shows the input clock requirement and correct data word alignment in a Cyclone series device.

Figure 8. Input Clock and Serial Received Data Relation for Correct Data Word Alignment for an 8-bit 1 Channel LVDS Interface



If you interface a Cyclone series device to another Cyclone series device using LVDS, perform the following setting for the transmitter device using the ALTLVDS megafunction:

- Select deserialization factor value as the outclock divide factor (B) or ensure that the `outclock` frequency is equal to `coreclock` frequency.

In certain interfaces, you may meet the input clock frequency requirement (in which `clock=data rate/deserialization factor`) but the MSB bit is not aligned with the rising edge of the input clock. The data word boundary will be incorrect and may corrupt the received data. In such cases, use `RX_DATA_ALIGN` port (a bit-slipping feature) in the ALTLVDS megafunction to realign the data word boundary. The `RX_DATA_ALIGN` port is a user input port based on pulse response. Hence, you must determine the number of times to pulse it to achieve the correct data word boundary. Run timing simulation on the interface to determine how many time to pulse the `RX_DATA_ALIGN` port of the ALTLVDS megafunction each time you power up or reset the PLL to realign the word boundary.

In cases where the input clock frequency is different than the requirement, you need to have a data checking mechanism in place. Take a case of an 800 Mbps interface and a deserialization factor of 8. For example, if the clock rate is at 200 MHz, there are two rising clock edges for each of the serial word received. In that case, the PLL could lock to either of those rising edges, and thus you would have a 50% chance of having the same word alignment from one power-up cycle to another or after a PLL reset. Design a data checking mechanism with the `RX_DATA_ALIGN` port to determine the correct word boundary before establishing the data transmission. Typically, a training sequence is used to establish the correct word boundary as commonly used in serial protocols such as SPI 4.2.

For more information on ways to use the RX_DATA_ALIGN port in data-word alignment, refer to the *SERDES Transmitter/Receiver (ALTLVDS) Megafunction User Guide*.

Step 4: Add Constraints

The next step in the design process is to add all timing, location, and physical constraints related to the LVDS interface. This includes timing, pin locations, I/O standards, and pin loading assignments.

Differential Pin Placement

Use the Quartus II Pin Planner Package view to ease differential I/O assignment planning. On the View menu, click **Show Differential Pin Pair Connections** to highlight the differential pin pairing. The differential pin pairs will be connected with red lines. For differential pins, you only need to assign the signal to a positive pin. The negative pin is assigned automatically by the Quartus II software if the positive pin is assigned with a differential I/O standard.

Each differential pin pair has matched routing with minimal skew between the positive and the negative pins in Cyclone series devices. Internal routings are matched even for pins in a pair that are not next to each other on the package. To minimize skew and maximize performance, PCB designers must ensure the same trace lengths to the receiving device.

Logic Placement

The Quartus II software automatically places the SERDES logics at the best location to meet timing requirements. Therefore, you are not required to perform placement constraints on the ALTLVDS megafunction logics.

TCCS parameter is guaranteed per datasheet specification to the entire bank of differential I/O when the transmitter SERDES logic is placed within the logic array block (LAB) adjacent to the output pins. If you find that the TCCS performance is not satisfactory and does not meet the TCCS specification using the default Quartus II fitting, create LogicLock™ regions in the device floorplan for the transmitter SERDES logics. Constrain the transmitter SERDES logics to the LAB adjacent to the data output pins and clock output pins to improve the TCCS performance.

For step-by-step instructions on the methods to create a design floorplan with LogicLock location assignments, refer to *Design and Synthesis* chapter in volume 1 of the *Quartus II Handbook*.

Timing Constraint

For receiver designs that are using the SERDES in LEs, you must ensure proper timing constraints for the TimeQuest timing analyzer tool in the Quartus II software to indicate whether the SERDES captures the data as expected or otherwise.

For more information SERDES in LEs and ensuring proper timing constraints, refer to the *LVDS SERDES Transmitter/Receiver (ALTLVDS_TX and ALTLVDS_RX) Megafunctions User Guide*.

The Quartus II compiler automatically ensures the associated delay chain settings are set correctly for the data path at the LVDS receiver that uses the source-synchronous compensation mode of PLL operation. If the input clock and data are not edge-or center-aligned at the receiver, timing constraints may be necessary for the TimeQuest timing analyzer tool in the Quartus II software to indicate whether the SERDES captures the data as expected or otherwise.

Step 5: Compile Design and Verify Timing Closure

Compile the design in the Quartus II software after performing the proper constraints to the design. Check the timing report in classic timing analyzer or TimeQuest timing analyzer for any timing violation on clock setup and clock hold paths upon completion of the compilation.

☞ In the TimeQuest timing analyzer tool, the `report_tccs` and `report_rskm` commands are not available for Cyclone series devices. The commands are only available for dedicated transmitter and receiver with SERDES, as in Stratix® series devices.

For receiver designs which are not using the source-synchronous mode of the PLL operation, or in Cyclone devices, you must ensure proper timing constraints for the data to be captured correctly. Perform the following guidelines to ensure proper timing constraints:

- Check the t_{su} for serial data input pin.
- At the input register, ensure that the clock is -90° or -180° to the data to maximize the read capture margin respectively for double-data and single-data rate operation.
- If the clock is not centered to the data at the input register, follow "Step 6: Adjust Constraints" on page 15 to improve the timing.

☞ The RSKM reduces significantly if the clock is shifted away from the center of the data window (refer to Figure 4 on page 5). You must have a maximum RSKM margin when the clock is centered at the data window. You must determine if you have a sufficient RSKM margin to account for board-level skews.

Step 6: Adjust Constraints

You can make adjustments to the constraints if the timing result does not fulfill the requirement or the design needs to be fine-tuned to improve the margin.

In cases in which the clock is shifted from the center of the data window without using source-synchronous compensated PLL, adjust the PLL phase settings of the serial data `rxin` with respect to the rising edge of `rx_inclock`. Recompile the design and check if the clock is centered at the data window at the data input registers.

Step 7: Determine Board Design Constraints and Perform Board Level Simulations

After closing the timing for the design, examine the board design to determine how different factors can have an effect on the signal integrity, thus affecting overall timing seen at the receiving device of the LVDS interface. The time margin for the LVDS receiver (indicated by RSKM) is the timing budget allocation for board level effects, such as skew, jitter, and noise. Exceeding the timing budget causes the LVDS receiver megafunction to fail.

Board-level skew arises from board trace lengths, use of connectors, and parasitic circuits variations. Jitter effects are derived from electromagnetic interference (EMI) such as cross-talk. On board resources with imperfect power supplies and reference planes may also contribute to the noise factor.

Altera recommends that you follow the board design considerations highlighted in the respective device handbook.

For more information on the board design considerations for Cyclone III family devices, refer to the *Design Guidelines* section of the *High-Speed Differential Interfaces in Cyclone III Devices* chapter in volume 1 of the Cyclone III Device Handbook.

For more information on the board design considerations for Cyclone II devices, refer to the *Design Guidelines* section of the *High-Speed Differential Interfaces in Cyclone II Devices* chapter in volume 1 of the *Cyclone II Device Handbook*.

For more information on the board design considerations for Cyclone devices, refer to the *Design Guidelines* section of the *High Speed Differential Signaling in Cyclone Devices* chapter in volume 1 of the *Cyclone Device Handbook*.

For PCB layout guidelines, refer to *AN 224: High-Speed Board Layout Guidelines* and *AN 315: Guidelines for Designing High-Speed FPGA PCBs*.

After determining the system requirements for the board design and finalizing the right board constraints, perform board-level simulation to ensure that the setup is optimum. Determine if the data window meets the input specification (electrical and timing) of the LVDS receiver. The Cyclone III device family supports the programmable pre-emphasis feature on true LVDS output buffers to compensate the frequency-dependent attenuation of the transmission line. This feature helps to maximize the data-eye opening at the far-end receiver, especially on long transmission lines.

There are various EDA simulation tools available to perform board-level simulations. Perform simulations with IBIS or HSPICE models for the FPGA and the target LVDS interface device.

Step 8: Verify FPGA Functionality

You can obtain useful information about the LVDS interface performance with board-level verification using the FPGA prototype. While the focus here is to ensure the FPGA functionality in your end system, you can take additional steps to examine margins using oscilloscopes to verify the predicted size of the data-valid window, setup and hold margins at the I/O interface.

You can also use Altera's SignalTap® II Embedded Logic Analyzer to perform system level verification to correlate the system against your design targets.

For more information on using SignalTap II, refer to the *Design Debugging Using the SignalTap II Embedded Logic Analyzer* chapter in volume 3 of the *Quartus II Handbook*.

Conclusion

Cyclone series devices support high-speed LVDS interfaces at speeds up to 840 Mbps. The ALTLVDS megafunction in the Quartus II software provides smooth integration of the LVDS interfaces into Cyclone series devices in cost-sensitive applications. FPGA design flow and guidelines for the LVDS interface implementation provides a good out-of-the-box experience with the Cyclone series devices.

Referenced Documents

This application note references the following documents:

- *High-Speed Differential Interfaces in Cyclone III Devices* chapter in volume 1 of the *Cyclone III Device Handbook*
- *High-Speed Differential Interfaces in Cyclone II Devices* chapter in volume 1 of the *Cyclone II Device Handbook*
- *High-Speed Differential Signaling in Cyclone Devices* chapter in volume 1 of the *Cyclone Device Handbook*
- *AN 224: High-Speed Board Layout Guidelines*
- *AN 315: Guidelines for Designing High-Speed FPGA PCBs*
- *Design Debugging Using the SignalTap II Embedded Logic Analyzer* chapter in volume 3 of the *Quartus II Handbook*
- *Design and Synthesis* in volume 1 of the *Quartus II Handbook*
- Tutorial in the Quartus II software
- *LVDS SERDES Transmitter/Receiver (ALTLVDS_TX and ALTLVDS_RX) Megafunctions User Guide*

Document Revision History

Table 2 shows the revision history for this application note.

Table 2. Document Revision History

Date and Document Version	Changes Made	Summary of Changes
July 2013 v1.2	Updated "Timing Constraint" on page 14.	Minor update to clarify that proper timing constraint is needed for receiver designs that are using SERDES in LEs and added a link to the ALTLVDS megafunction user guide.
June 2009 v1.1	■ Removed Table 2 on page 4. ■ Updated "Step 2: Select Device" on page 4. ■ Updated "I/O Resources" on page 6. ■ Updated "Phase-Locked Loop (PLL) Resources" on page 6. ■ Updated "Clock Resources" on page 7. ■ Updated "Step 7: Determine Board Design Constraints and Perform Board Level Simulations" on page 16.	Updated to include Cyclone III LS information.
June 2008 v1.0	Initial release.	—

101 Innovation Drive
San Jose, CA 95134
www.altera.com
Technical Support
www.altera.com/support