Using Flexible-LVDS I/O Pins in APEX II Devices

August 2002, ver. 1.1

Application Note 167

Introduction

Recent expansion in the telecommunications market and growth in Internet use have created a demand to move more data faster than ever. To meet this demand, system designers are relying on solutions such as differential signaling and interface standards such as RapidIO, POS-PHY Level 4, and UTOPIA IV.

APEX™ II device high-speed interface I/O pins offer serialization and deserialization on a single chip to move data at high speeds. They also utilize a state-of-the-art CMOS process that consumes far less power than GaAs devices, the other alternative for high-speed devices.

Preliminary Information

The following documents provide information on APEX II device high-speed I/O standard features and functions. These documents also explain how system designers can take advantage of these standards to increase system efficiencies and bandwidth.

- *Application Note 157 (Using CDS in APEX II Devices)* describes the most common clock topologies, and how the unique clock-data synchronization (CDS) feature in APEX II devices is applied.
- *Application Note 166 (Using High-Speed I/O Standards in APEX II Devices)* provides information on APEX II device high-speed I/O standard features and functions. This document also explains how system designers can take advantage of these standards to increase system efficiencies and bandwidth.

Flexible-LVDS Differential Buffers

The APEX II high-speed interface offers four high-speed I/O banks. Each I/O bank is comprised of 18 channels, offering 36 differential input channels and 36 differential output channels. Every channel can transmit data at speeds of up to 1 gigabit per second (Gbps).

APEX II devices also offer 88 Flexible-LVDS™ pins that use internal phase-locked loops (PLLs) to transmit or receive data at 400 Megabits per second (Mbps). Flexible-LVDS pins are located in standard user I/O banks and only require a 100-Ω termination resistor at the input receiver pins. Flexible-LVDS pins support LVDS, LVPECL, and HyperTransport signaling on the receiver side and LVDS and HyperTransport signaling on the transmitter side. Because Flexible-LVDS I/O pins implement serialization/deserialization (SERDES) with minimal logic in APEX II devices, they do not require dedicated circuitry.

AN-167-1.1

Flexible-LVDS I/O Interface

Designers can use dedicated double data rate (DDR) circuitry to implement Flexible-LVDS I/O pins in APEX II devices. While single data rate (SDR) circuitry only samples data at the positive edge of the clock, DDR circuitry captures data on both the rising and falling edges, and is therefore capable of doubling the maximum SDR transfer rate.

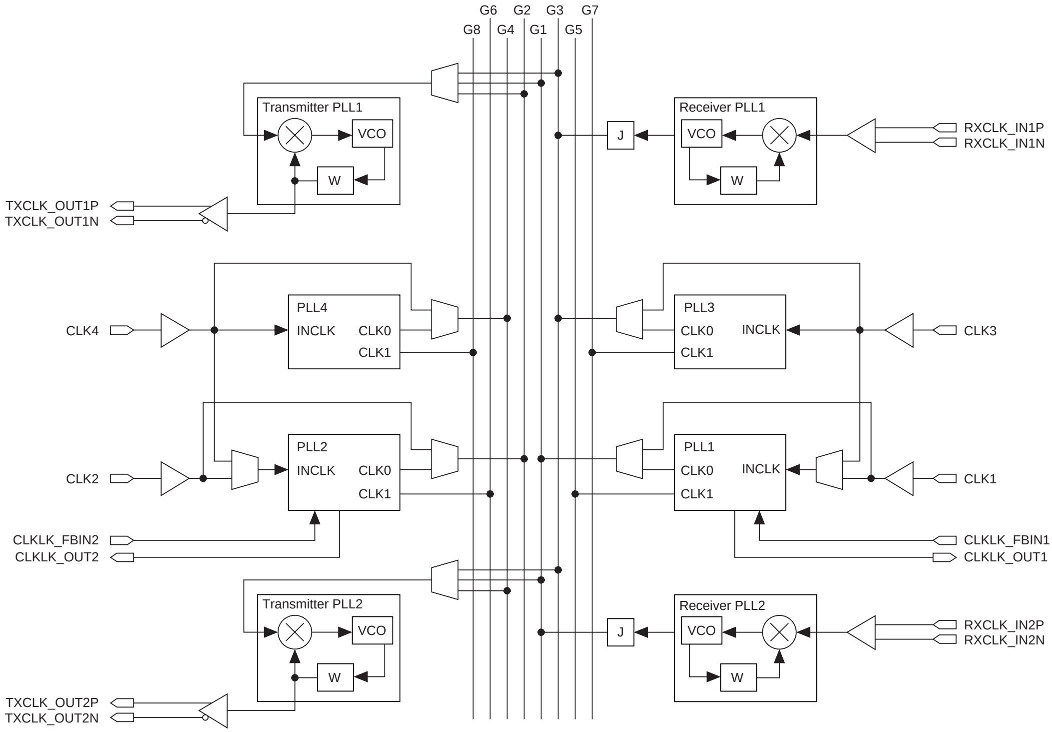
Designers can use APEX II device shift registers, internal global PLLs, and I/O cells to perform serial-to-parallel conversions on incoming data and parallel-to-serial conversions on outgoing data.

Clock Domains

Flexible-LVDS I/O pins use the many clock domains available in APEX II devices. These clock domains fall into four categories: eight global clock domains, two I/O element (IOE) clock domains from the peripheral control bus, four fast I/O clock domains, and unlimited, internally generated clock domains.

The four general-purpose PLLs generate the eight global clock domains. Each PLL features two taps that directly drive two unique global clock networks. A dedicated clock pin drives each of the four general-purpose PLLs. These eight clock lines are utilized when designing for speeds up to 400 Mbps. Figure 1 shows the PLL connections to the dedicated global clock lines. For more information on general purpose PLLs, see *Application Note 156 (Using General-Purpose PLLs with APEX II Devices)*.

Figure 1. APEX II PLL Clock Connections & Dedicated Global Clock Lines



Each APEX II device IOE selects clock, clear, clock enable, and output enable controls from the peripheral control bus, a network of I/O control signals. The peripheral control bus uses high-speed drivers to minimize signal skew across devices. In addition to the eight global clock signals, two of the twelve APEX II peripheral control bus signals can feed the IOE register's clock ports. Each one of the two clocks can be driven by any of the dedicated input pins or from any logic element (LE). Figure 2 shows the IOE configuration for DDR input. Figure 3 shows the IOE configuration for DDR output.

Figure 2. APEX II IOE in DDR Input I/O Configuration

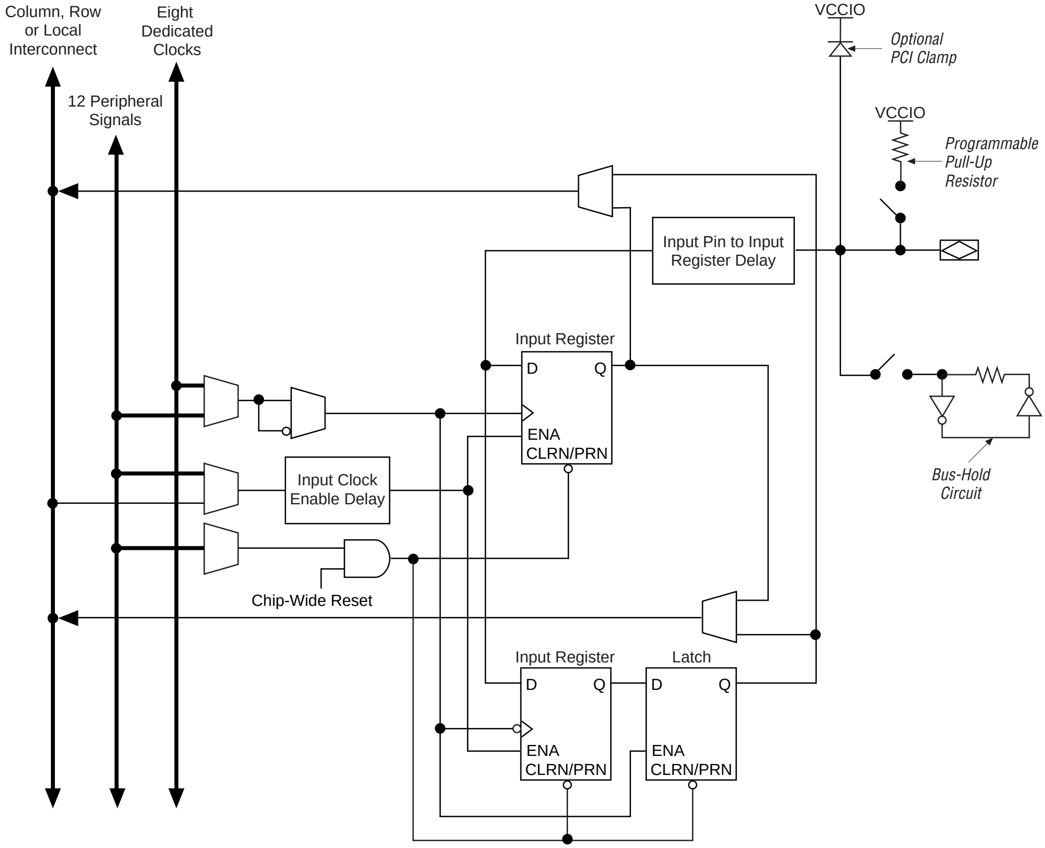
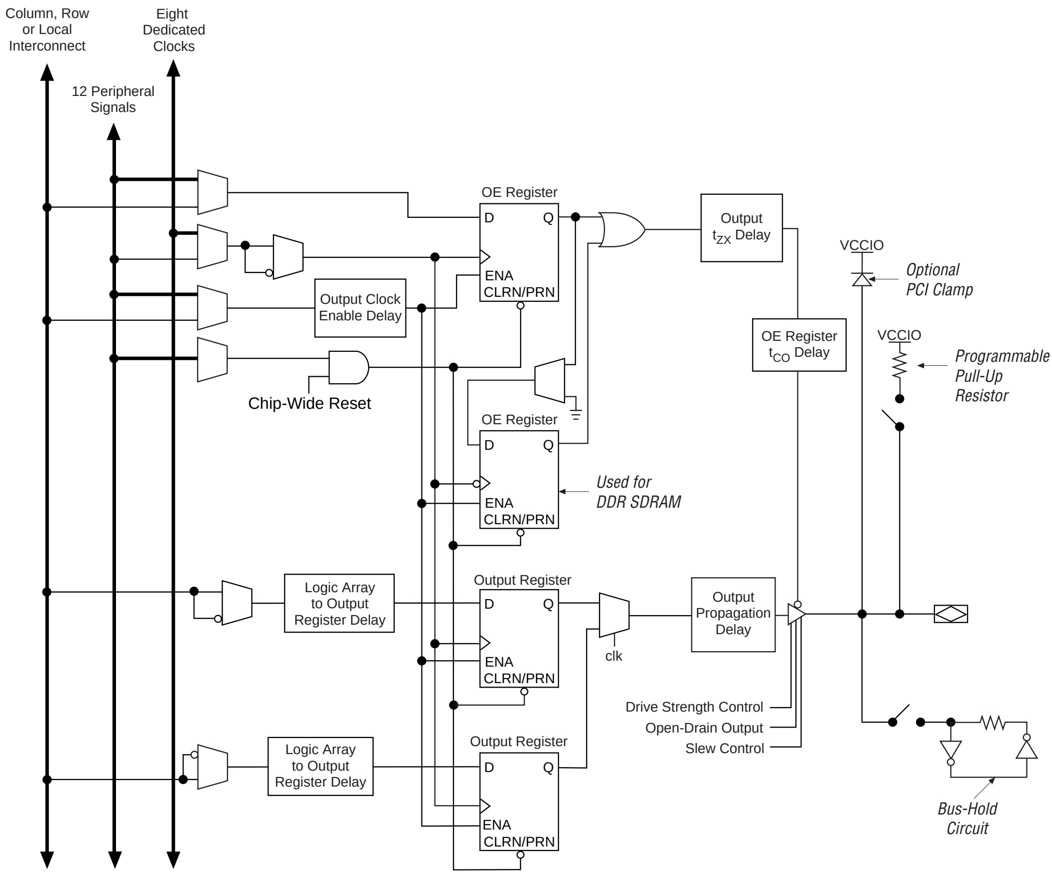


Figure 3. APEX II IOE in DDR Output I/O Configuration



The four dedicated fast I/O pins can also function as clocks. These fast I/O pins have a lower maximum speed than the global clocks. A fast I/O pin drives the IOE clock through the peripheral control bus.

Flexible-LVDS I/O Receiver Operation

The Flexible-LVDS I/O receiver uses the APEX II device's DDR input circuitry to receive high-speed serial data. The DDR input circuitry consists of a pair of registers used to capture the high-speed serial data and a latch. One register captures the data on the positive edge of the high-frequency clock (generated by PLL) and the other register captures the data on negative edge of the high-frequency clock. The data captured on the negative edge is delayed by one half of the high-speed clock cycle. Therefore, the data is latched before it interfaces with the system logic.

Figure 4 shows the DDR timing relation between the incoming serial data, and the high-frequency clock. The `inclock` signal is running at half the speed of the incoming data. Figure 5 shows the DDR input and the other modules used in a Flexible-LVDS receiver design to interface with the system logic.

Figure 4. DDR Timing Relationship between the Incoming Serial Data & Clock

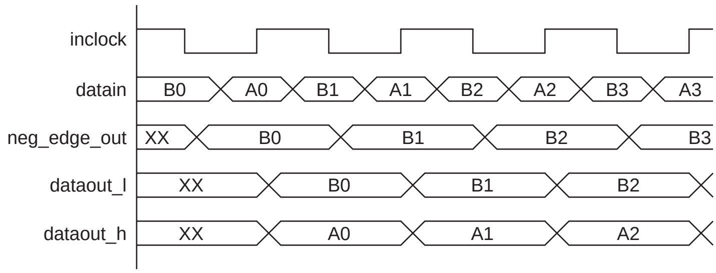
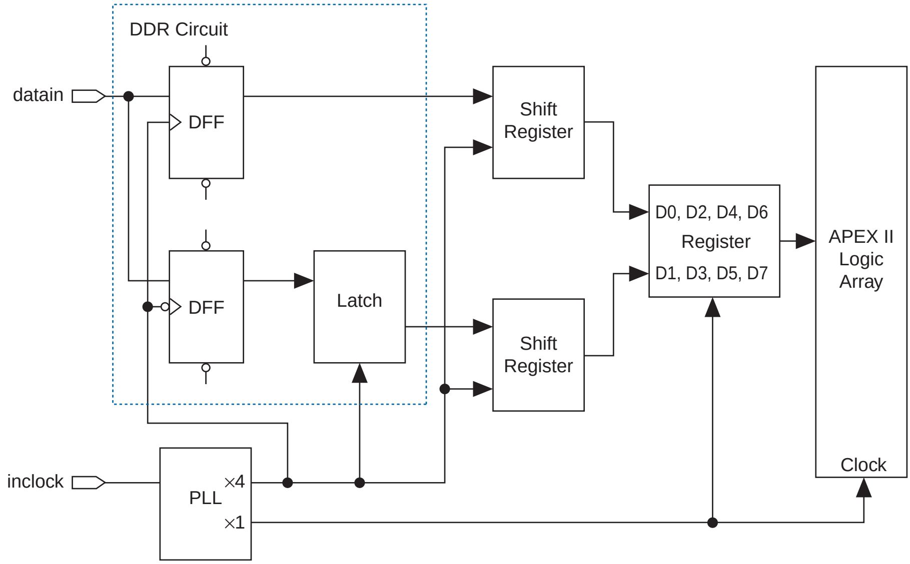


Figure 5. Flexible-LVDS Receiver Interface (×8 Mode)



Flexible-LVDS I/O Transmitter Operation

The Flexible-LVDS I/O transmitter uses the APEX II device's DDR output circuitry to transmit high-speed serial data. The DDR output circuitry consists of a pair of registers and a multiplexer. The transmitter has a pair of shift registers that capture and transfer data to the DDR output circuitry.

Figure 6 shows the DDR timing relation between the parallel data and the low-frequency clock. The `inclock` signal is running at half the speed of the data. Figure 7 shows the DDR output and the other modules used in a Flexible-LVDS transmitter design to interface with the system logic.

Figure 6. DDR Timing Relation between Parallel Data & Clock

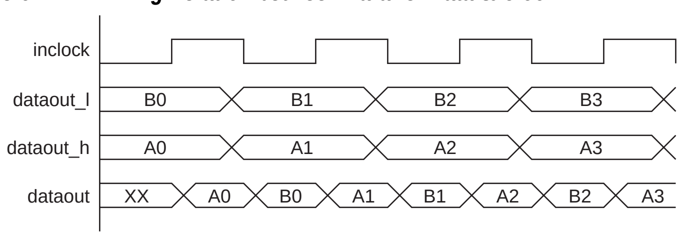
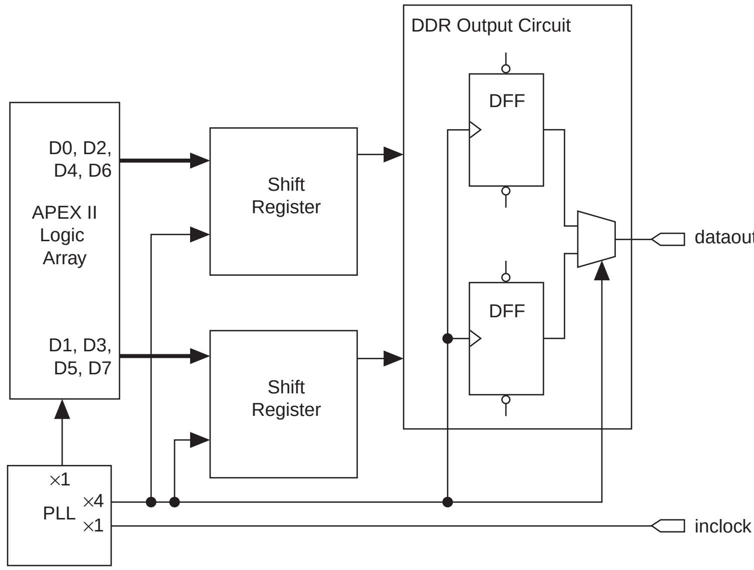


Figure 7. Flexible-LVDS I/O Transmitter Interface (×8 Mode)



Quartus II Software

Designing with Flexible-LVDS I/O buffers requires the use of the `ddio` megafunction in the Quartus® II software. Other functions such as serial shift registers and PLLs are also implemented to receive or transmit data at high speeds.

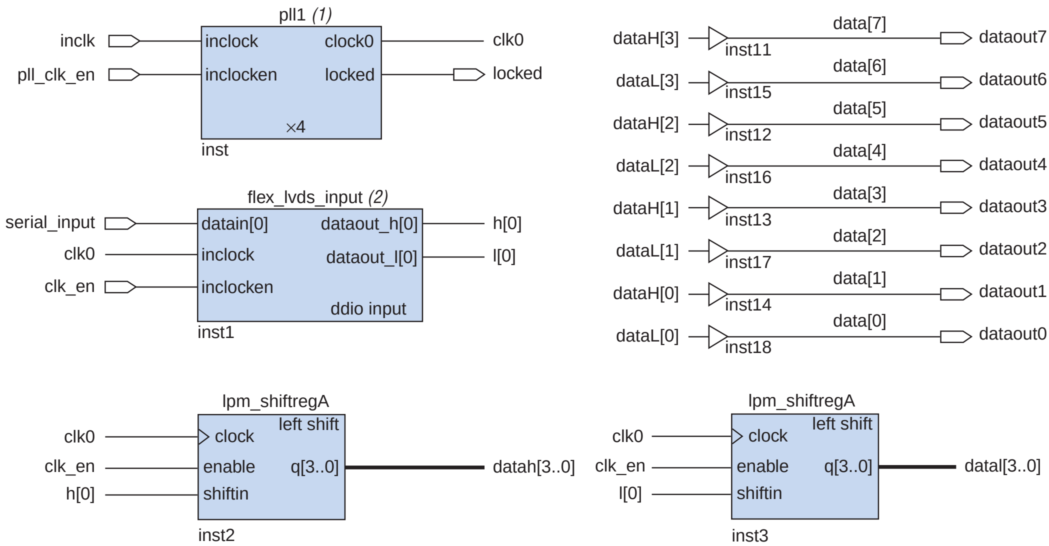
The following section discusses an example design that consists of both a VHDL receiver circuit and a transmitter circuit for I/O buffers. The design used in this section is also available on the Altera web site (**http://www.altera.com**). Although this example is for data transfers where the data rate is 8× the clock rate, the data transfer can be easily modified for other data/clock relationships.

Building an 8-Bit Flexible-LVDS Receiver

The DDR input register receives the data and separates it into odd bits and the even bits. The incoming data bits 0, 2, 4, and 6 are connected to the input of one shift register, and the data bits 1, 3, 5, and 7 are connected to the input of the other shift register. These two shift registers de-serialize the data. A third register, clocked by the low-frequency clock, drives the parallel data to the system design.

Figure 8 shows all the modules necessary for a single Flexible-LVDS buffer to receive serial data.

Figure 8. Complete Receiver Module



Notes to Figure 8:

(1) Input period = 25 ns; `clock0` frequency multiplication factor = 4.

(2) Power up low.

The `altddio_in` block captures the serial data on both clock edges and parses the data into two outputs. The bits captured on the negative edges are latched before they are driven to the shift register modules. This synchronizes the bits with the data captured on the rising edge of the clock.

The general-purpose PLL module (`PLL1`) generates the high-speed clock for the deserialization registers. The `inclk` signal is multiplied by a factor of four, generating the clock signal required by the shift registers for deserializing the data. The multiplication factor may be changed for different data-to-clock relationships.

The `clk0` PLL output clocks a pair of shift registers, which converts data from serial to parallel. The incoming data bits 0, 2, 4, and 6 are connected to the input of one shift register, and the data bits 1, 3, 5, and 7 are connected to the input of the other shift register. The shift registers deserialize the data, which then is driven to the system design.

Eight wires reconstruct the data bits and make the connection between the Flexible-LVDS circuitry and the system design.

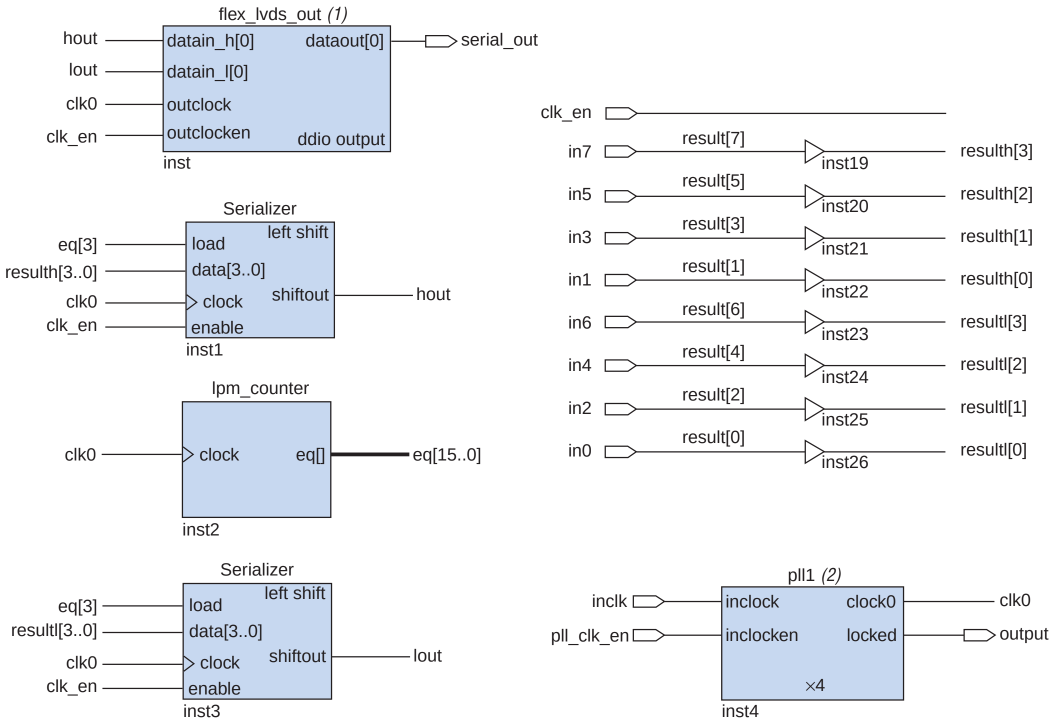
Building an 8-Bit Flexible-LVDS Transmitter

The data is received on two shift registers. Outgoing data bits 0, 2, 4, and 6 are connected to the input of one shift register and data bits 1, 3, 5, and 7 are connected to the input of the other shift register.

The DDR output module uses a high-speed clock generated by the general-purpose PLL module to transmit the serial data. A counter signals the shift register to receive data every fourth clock cycle.

Figure 9 shows all the modules necessary for a single Flexible-LVDS buffer to transmit serial data.

Figure 9. Complete Transmitter Module



Notes to Figure 9:

(1) Power up low.

(2) Input period = 25 ns; `clock0` frequency multiplication factor = 4.

A pair of shift registers is clocked by `clk0` (PLL output) and serialize the data. Data bits 0, 2, 4, and 6 are connected to the input of one shift register, and data bits 1, 3, 5, and 7 are connected to the input of the other shift register.

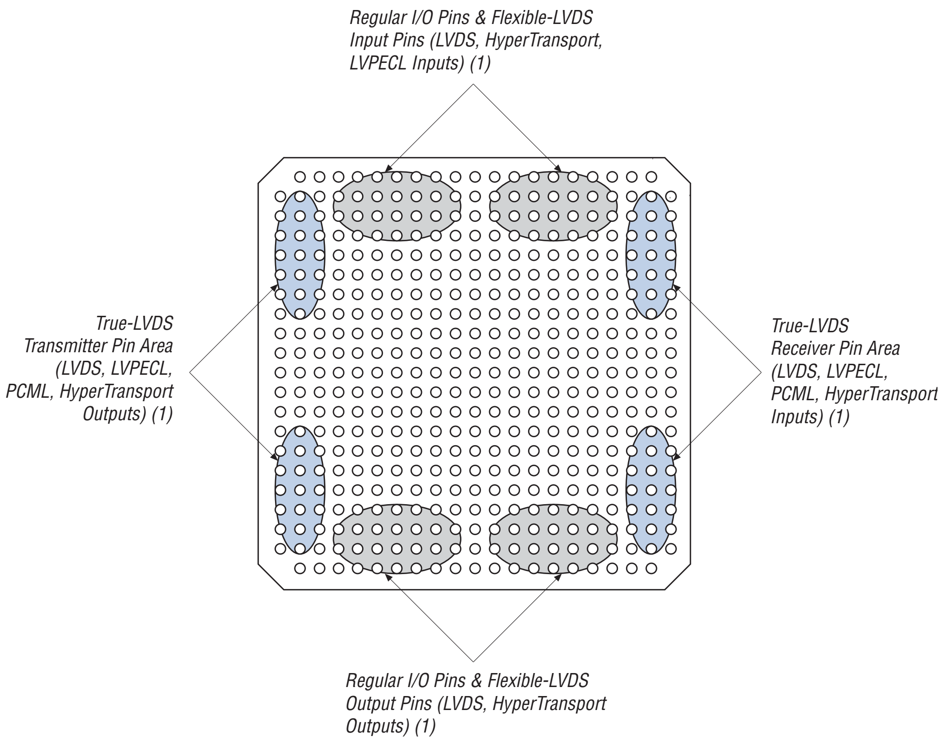
The general-purpose PLL module generates the high-speed clock. The `inclk` signal is multiplied by a factor of four, generating the clock signal the shift registers require to serialize the data. A counter signals the shift register to receive data on every fourth clock cycle.

The `altddio_out` megafunction block captures data on clock rising edges and parses the data to a single output. For both receiver and transmitter, the ×1 clock should be used to transmit or receive data to or from the system logic.

Flexible-LVDS I/O Pin Locations

APEX II Flexible-LVDS I/O pins are located at the edge of the package to reduce the possible mismatch between a pair of high-speed signals. Figure 10 shows the I/O blocks and their location relative to the package. Flexible-LVDS I/O pins are located on top and bottom of the device.

Figure 10. True-LVDS & Flexible-LVDS I/O Pins



Note to Figure 10:

(1) The shaded ovals show the approximate locations of the True-LVDS™ or Flexible-LVDS pins.

Summary

Flexible-LVDS I/O pins are dual-purpose user I/O pins that provide additional differential channel support in APEX II devices. The Flexible-LVDS solution supports up to 88 transceiver channels at a 400-Mbps data rate. It also supports applications that need more than 36 LVDS channels. External resistors are only needed for receivers, not on the transmitters. The function is easily implemented by instantiating Altera's library of parameterized modules (LPM) functions and the supplied reference design.

Revision History

The information contained in *AN 167: Using Flexible-LVDS I/O Pins in APEX II Devices* version 1.1 supersedes information published in previous versions.

Version 1.1

AN 167: Using Flexible-LVDS I/O Pins in APEX II Devices version 1.1 contains the following changes:

- Changed the value from 624 to 400 Mbps throughout the document.
- Updated notes of Figures 8 and 9 to read 25 ns instead of 16 ns.

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com
Applications Hotline:
(800) 800-EPLD
Customer Marketing:
(408) 544-7104
Literature Services:
lit_req@altera.com

Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. LeonardoSpectrum and Exemplar Logic are trademarks of Exemplar Logic. Mentor Graphics is a trademark of Mentor Graphics. All other product or service names are the property of their respective holders. All rights reserved. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services. All rights reserved.

Copyright © 2001 Altera Corporation. All rights reserved.

I.S. EN ISO 9001

Printed on Recycled Paper.