

---

# Using LVDS for Microsemi Axcelerator and RTAX-S Devices

---

## Table of Contents

---

Introduction . . . . .	1
LVDS Timing Requirements . . . . .	2
LVDS Board Layout Requirements . . . . .	7
Axcelerator/RTAX-S LVDS Interface . . . . .	10
Design Guidelines for LVDS in Axcelerator or RTAX-S Devices . . . . .	13
Conclusion . . . . .	23
List of Changes . . . . .	24
Related Documents . . . . .	25
References . . . . .	25

---

## Introduction

This application note describes the low voltage differential signaling (LVDS) I/O capabilities of Axcelerator<sup>®</sup> and RTAX-S device families by Microsemi. The application note begins by describing the LVDS signaling standard, then discusses detailed timing and board layout requirements, and finally provides a description of LVDS features unique to Axcelerator and RTAX-S.

### Overview

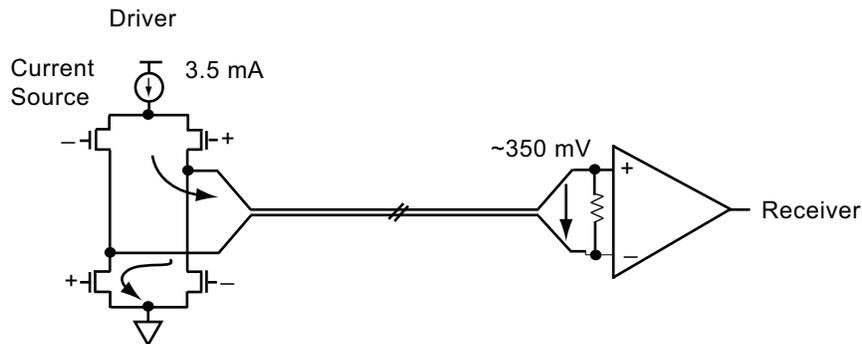
LVDS is an I/O specification originally proposed by National Semiconductor Corporation<sup>®</sup> and made into two standards: ANSI/TIA/EIA-644 and IEEE 1596.3. The original ANSI/TIA/EIA standard is primarily an electrical interface standard and provides requirements for an LVDS driver and receiver. The IEEE standard is more general and covers the implementation of LVDS drivers and receiver pairs for a variety of applications, refer to "[References](#)" section on page 25.

LVDS is a differential low voltage I/O standard that enables serial data transfer at very high rates with the following properties:

- High speed
- Low power (less than 2 mW)
- High common-mode noise rejection
- Simple termination (100  $\Omega$  resistor on driver/receiver pair)
- Low signal swing (differential voltage from 247 mV (min.) to 454 mV (max.))
- Low cost

## Using LVDS

LVDS works by switching current from a current source through a differential line pair. The current loop is completed by placing a  $100\ \Omega$  termination resistor across the differential line pair. This resistor creates a voltage swing across the LVDS receiver's inputs. Since the switching current through the differential line pair is very small (on the order of 3.5 mA), power dissipation across the termination resistor is also small (less than 2 mW, as listed above). [Figure 1](#) shows a schematic of an LVDS driver or receiver pair.



**Figure 1 • LVDS Driver/Receiver Pair**

## LVDS Timing Requirements

Since LVDS is intended for high-speed data transmission, understanding its timing requirements ensures the integrity and quality of both transmitted and received data. Important parameters to consider are the LVDS eye diagram, data sampling requirements, timing margin, skew between the LVDS channels, and the receiver input skew margin. Each of the requirements are discussed in detail below.

### BER

The bit-error rate (BER) is an important measurement of the performance of your LVDS channels. It is measured in [EQ 1](#):

$$\text{Bit Error Rate} = \frac{(\text{Number of Bit Errors})}{(\text{Total Number of Bits})}$$

*EQ 1*

Common requirements for BER are either  $\leq 1 \times 10^{-12}$  or  $\leq 1 \times 10^{-14}$ . These correspond to one bit error in one trillion and 100 trillion bits sent, respectively.

Microsemi recommends using bit-error rate test (BERT) equipment for measuring the BER of your LVDS channels under worst-case operating conditions. For example, using the longest cable lengths in the system, the highest operating frequency, and the noisiest environment. BERTs may take a long time depending on the transmission rate (in bits per second) of your channels. Refer to [National Semiconductor's AN-1040](#) application note for more information on how to test for BER.

Transmission line parameters such as skew and jitter, timing, and receiver skew margin can increase your BER and need careful attention. They are discussed in the ["Timing Margin" section on page 5](#).

### Eye Diagram

LVDS is an I/O signaling standard that supports different serial data transmission protocols driven over a transmission line. Several transmission line patterns are necessary to quantify the effect of noise, attenuation, and random data on the line while data is being transmitted or received. Each of these patterns affect the BER at the receiver. In particular, the following patterns should be monitored:

- VOH (Output High Voltage) and VOL (Output Low Voltage) of constant '1' and '0' bits, respectively
- 0-1 and 1-0 transitions rates
- 0-1-0 and 1-0-1 transitions

Figure 2 illustrates all these patterns. The left side of the figure shows the ideal behavior of a signal on the line. The right side shows its actual behavior seen at the receiver.

If these patterns are superimposed, they form an eye diagram (Figure 3 on page 4). The height of the eye diagram shows the maximum voltage swings seen on the transmission line. Its width represents the signaling rate.

For a transmission channel to be compliant with the LVDS signaling, the height of its eye diagram must meet or exceed the LVDS differential voltage specification.

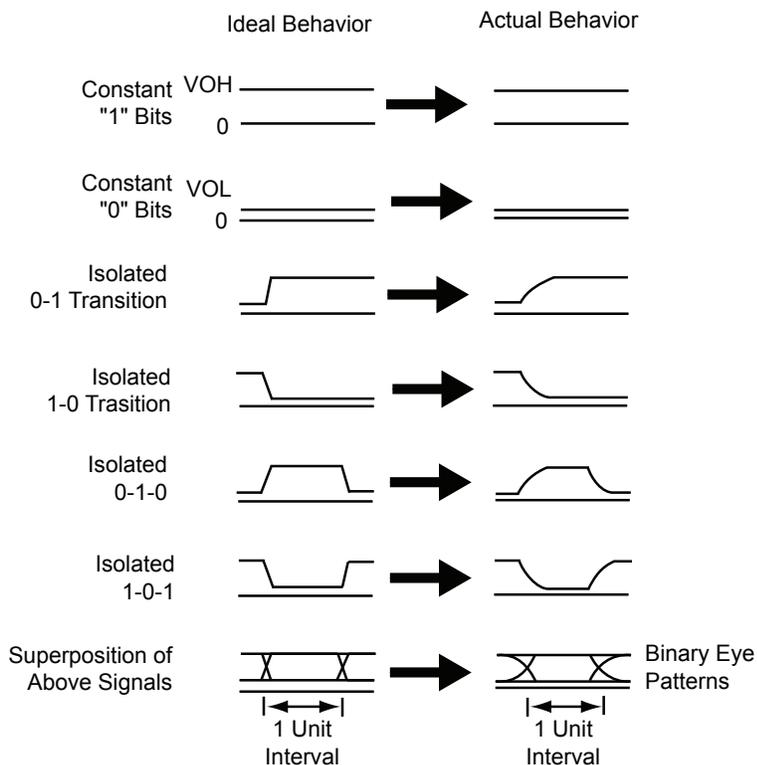
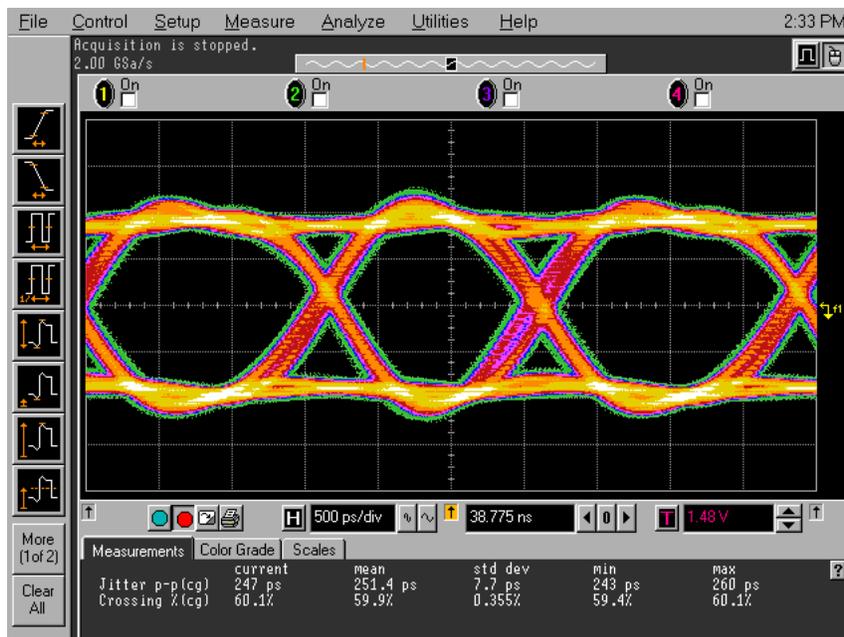


Figure 2 • Data Transmission Patterns

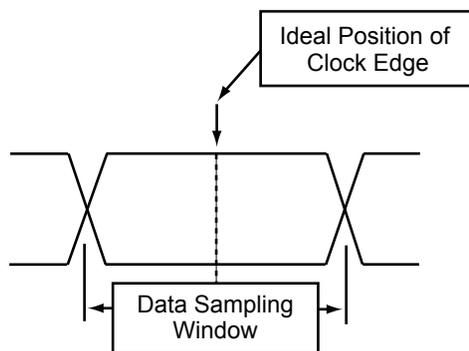


**Figure 3 • LVDS Receiver Eye Diagram Axcelerator Device (Vertical Scale: 100 mV/div, Horizontal Scale: 500 ps/div)**

## Data Sampling Window

Whenever data is transmitted using the LVDS I/O, a reference clock that is source synchronous with the LVDS data stream is either regenerated at the receiver using a phased locked loop (PLL) or transmitted with the data stream. Data may be latched on either the rising edge or both the rising and falling edges of the reference clock. In the latter case, the data is said to be transmitted in double data rate (DDR) mode. The relationship between the time each bit is valid (also called the bit width) and the edge of the reference clock is crucial to ensure that the receiver accurately samples each bit. This relationship is shown graphically as the data sampling window in [Figure 4](#).

In the ideal situation, the reference clock should be centered in the middle of the data sampling window to provide maximum setup and hold time margins at the receiver.



**Figure 4 • Ideal Clock Position**

## Timing Margin

The timing margin refers to the amount of extra delay available during which the receiver can accurately sample data. It is an important parameter for determining whether the transmission line has adequate time to transmit and receive data without error. In a real transmission line, the timing margin is determined by three factors:

- Receiver skew margin (RSKM)
- Transmission line skew (SKW)
- Clock and data jitter (JTTR)

Each of these factors, either working independently or together, narrow the data sampling window and reduce the timing margin on the transmission line. As the timing margin is reduced, the maximum transmission frequency of the LVDS data stream also decreases.

The timing margin of an LVDS transmission line is calculated using [EQ 2](#):

$$\text{Margin} = \text{RSKM} - \text{SKW} - \text{JTTR}$$

EQ 2

### **Transmission Line Skew**

There are three types of skew effects that occur on an LVDS transmission line: printed circuit board (PCB) trace, connector, and cable skew. Adding the delays associated with each of these skew effects determines the value of SKW described in the timing margin equation ([EQ 2](#)).

#### **Connector Skew**

If the LVDS cable connects to pins on different rows of the connector, they can introduce a small amount of skew (normally in picoseconds), which adds to the total skew of the design. Certain connectors are available, called Zero Skew connectors, that have no connector skew. These connectors allow LVDS data to connect to pins only on the same row. If a connector has skew, the connector manufacturer normally publishes a skew figure.

#### **Printed Circuit Board Trace Skew**

When the LVDS signals are routed from the PCB connector to receivers on the PCB with different trace lengths, they can introduce skew. The skew may occur either within the same channel or between channels. If not accounted for, this skew can be a significant problem when LVDS data needs to be transmitted at high speeds. This skew should be measured on each channel using TDR equipment.

The "[LVDS Board Layout Requirements](#)" section on [page 7](#) provides suggestions about how to manage and/or reduce this skew.

#### **Cable Skew**

The type of cable used, its length, and its frequency characteristics introduce a time delay (or cable skew) in the received data relative to the reference clock. If the LVDS receiver and/or transmitter pair are located in the PCB, the PCB trace interconnect length and connector capacitance add to the cable skew. Cable skew will narrow the data sampling window unless compensation is made for it. To determine the cable skew of the transmission line, Microsemi recommends you to measure the skew using time domain reflectometry (TDR) equipment.

## Clock/Data Jitter

If the reference clock for the LVDS data stream also shifts in time, it is called clock jitter. Reference clock jitter creates jitter on the LVDS data stream. Data is transmitted or received on the transmission line after being clocked out of or into registers. If the register's clock source is jittering, then its output will jitter.

Jitter on the LVDS data stream is also generated, if common mode noise couples on the amplitude of the reference clock signal. The clock source can pick up noise from several sources such as crosstalk between adjacent signals on the board, power supply noise, or inadequate termination of the clock signal. In most cases this noise modifies the edge rates of the clock slightly over time, thereby showing as jitter on LVDS data.

Short-term clock jitter can be reduced by feeding the original clock source into a PLL. However, most PLLs have an inherent jitter that propagates to the output clock source. The worst-case value of this jitter is defined and controlled. If you are using Axcelerator PLLs to generate an LVDS receive or transmit clock, refer to the *Axcelerator Family FPGAs* datasheet for specifications on the worst-case PLL jitter.

**Note:** The PLLs are not available for RTAX-S devices.

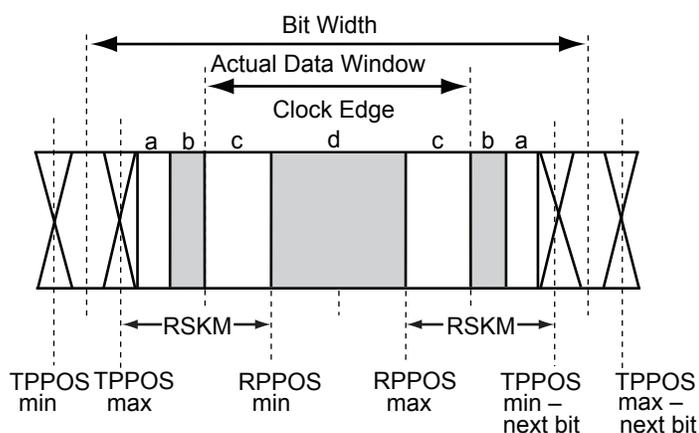
### Receiver Skew Margin

An LVDS transmitter takes a specific amount of time, typically known as its clock-to-output delay, before it can launch new data on the transmission line. Under typical operating conditions, this delay has a best-case (TPPOS<sub>min</sub> – Transmitter Pulse Position) and worst-case (TPPOS<sub>max</sub>) value. Alternately, the receiver has a minimum setup (RPPOS<sub>min</sub> – Receiver Pulse Position) and hold (RPPOS<sub>max</sub>) time requirement to latch the transmitted data. The RSKM defines the transmit time of the data, not including jitter or skew effects. Assuming that the clock edge is centered in the data sampling window, the RSKM can be calculated with EQ 3 and EQ 4.

$$\text{RSKM(Left Side)} = \frac{1}{2} (\text{Bit Width}) - \text{TPPOS}_{\text{max}} - \text{RPPOS}_{\text{min}} \quad \text{EQ 3}$$

$$\text{RSKM(Right Side)} = \frac{1}{2} (\text{Bit Width}) - \text{RPPOS}_{\text{max}} - \text{TPPOS}_{\text{min}} \quad \text{EQ 4}$$

Figure 5 illustrates the effect that RSKM and jitter have on narrowing the data sampling window. Initially, as shown in Figure 4 on page 4, the data sampling window is as wide as the bit width. The minimum size of the window required to correctly receive the data is shown in the region labeled **d**. This is the difference between RPPOS<sub>min</sub> and RPPOS<sub>max</sub>. Regions **a** and **b** show the total skew and clock/data jitter delays, respectively. The timing margin available to transmit data is labeled in region **c**. It is defined as the difference between RSKM and the sum of the skew and clock/data jitter delays. The final size of the data sampling window is shown as the union of regions **c** and **d**.



**Figure 5 • Receiver Timing Margin**

### Channel-to-Channel Skew

When LVDS data is being transmitted or received over multiple LVDS pairs, also called channels, the data on one channel may be delayed in relation to the other channels. This delay is called channel-to-channel skew.

The most common reasons for channel-to-channel skew are variances in the timing characteristics of the LVDS drivers for each channel, the board layout and trace length of signals in each channel, and variation in the termination resistance. If the data on all channels has to be latched on the same clock edge, channel-to-channel skew may cause incorrect data to be latched on some channels with correct data latched on other channels. Channel-to-channel skew effects become more important as data transmission frequencies increase.

While channel-to-channel skew cannot be eliminated, it can be managed by following both the PC board layout guidelines discussed in "LVDS Board Layout Requirements" and using other methods (beyond the scope of this application note) for deskewing each data channel.

## LVDS Board Layout Requirements

Careful PCB layout is essential to preserve both the integrity and transmission quality of the LVDS signals being received from and transmitted to the board. As signal transmission speeds increase, correct PCB layout becomes even more crucial. This section provides layout guidelines to maximize the LVDS signal integrity when transmitting or receiving high-speed data.

### Electrical Termination

The LVDS receivers require parallel termination for proper operation. A termination resistor between the positive and negative input of the receiver completes a current loop formed between the transmitter and receiver (Figure 1 on page 2). It also provides a voltage drop for the receiver's differential input comparator. Only one termination resistor is required; however, a differential termination scheme may also be used (Figure 6). The termination resistor must have a value of 90-130  $\Omega$  and should closely match the impedance of the transmission line within 1 to 2%.

The greater the mismatch between the termination resistor and the transmission line source impedance, the greater the chance common mode reflections could occur. These reflections degrade the signal integrity.

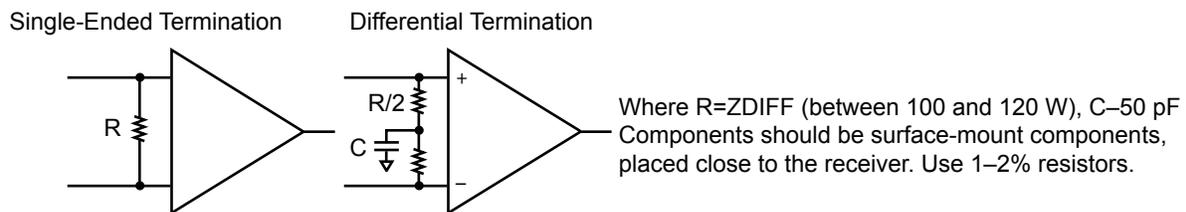


Figure 6 • Termination Requirements

### PCB Layout Guidelines

Some important layout guidelines to follow when designing with the LVDS receivers and transmitters are:

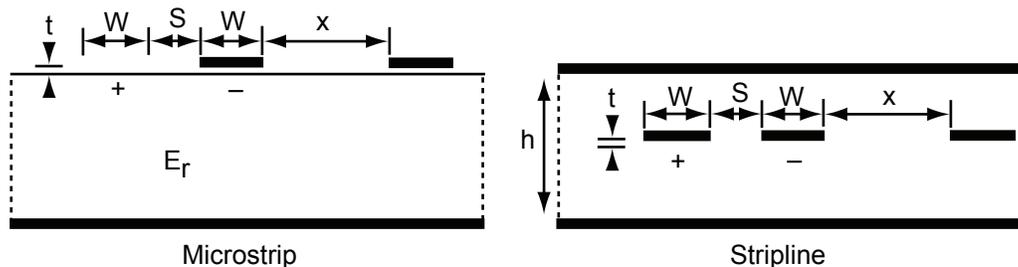
- Controlling trace length and width
- Signal placement
- Reducing electromagnetic interference (EMI) noise
- Fail-safe I/O biasing

#### Controlling Trace Length and Width

The LVDS signals are transmitted differentially. The advantage of this approach is that common-mode noise between the signal pair is eliminated. However, if the trace length between these signal pairs is not the same, they will have a phase difference (skew) with respect to each other. This means that the positive signal may arrive first at the receiver or vice-versa. If this occurs, then each signal will reflect off the receiver's termination at different times and create a common-mode noise source on the transmission line. Common-mode noise degrades the receiver's eye diagram, reduces signal integrity, and creates crosstalk between neighboring high-speed signals on the board. To minimize reflections due to unmatched trace lengths, consider the following guidelines:

- Match the length of each LVDS signal within the signal pair to within 100 mills.
- Minimize turns and vias (feed-throughs) – route as straight as possible from point-to-point. Do not use 90 degree turns when routing the LVDS signals. Use 45 degree bevels or rounded curves instead.

- Minimize vias on or near LVDS trace lines; they may create additional impedance discontinuities that will increase reflections at the receiver. If you need to use vias, place them as close to the receiver as possible.
- Use controlled impedance PCB traces. That is, control trace spacing, width, and thickness using Stripline or Microstrip layout techniques. [Figure 7](#) provides recommendations for trace length spacing. Note that controlled spacing also helps to reduce EMI emissions. This is discussed in the "Reducing EMI Noise" section on page 8.



For good coupling, make  $S < 2W$ ,  $S < h$ , and  $x \leq 2W$  and  $2S$ , where  $S$  = the trace spacing and  $W$  = the PCB trace width.

**Figure 7 • LVDS Signal Spacing**

### Signal Placement

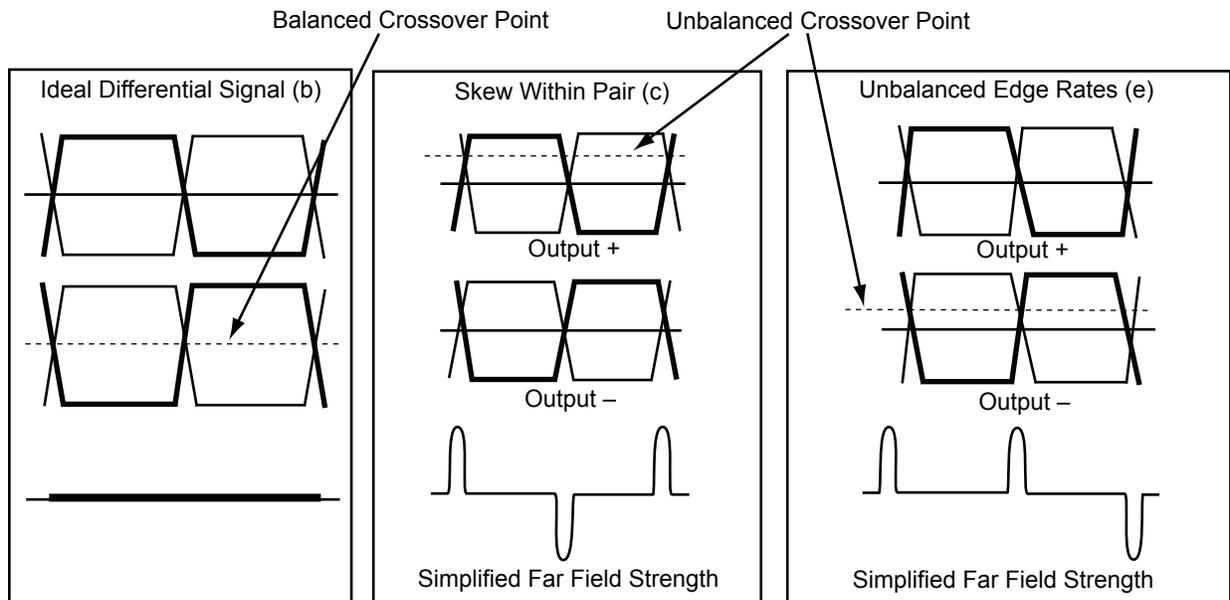
Placement of the LVDS signal pairs on the board is crucial for preserving the common-mode noise canceling characteristics of the LVDS differential signal pair. Improper or incorrect placement can induce common-mode noise on the signal pairs which reduces signal integrity and increases EMI noise emissions. Use the following guidelines for placing the LVDS signals:

- Place the LVDS receivers and transmitters as close as possible to their respective LVDS board connectors and receivers.
- Place the LVDS signals on their own signal layer separate from other signal layers. The ground plane should be the next layer below the LVDS signals, followed by the power plane. For example, an acceptable layer stack-up would be LVDS, ground, power, other signals. Ask your PCB manufacturer for more recommendations about this.
- If the preceding guideline is not practical, then place single-ended signals, such as TTL and CMOS, as far away from the LVDS signal traces as possible. This minimizes the crosstalk between them and the LVDS signals.
- If a discontinuity (for example, a via, component, PCB trace, or Power/GND plane) must be placed near an LVDS signal trace, then introduce the discontinuity equally on both members of the signal pair.
- Power and ground planes are essential for reducing noise and EMI radiation for designs using the LVDS devices.
- Use bypass capacitors on LVDS receiver and transmitter voltage supply pins. If the LVDS receiver or transmitter is located internally within a chip (for example, Accelerator), then use the capacitor bypass rules of the I/O voltage pins of the device.

### Reducing EMI Noise

The skew between LVDS complementary outputs can generate EMI noise on the board. The following are some guidelines to minimize EMI noise emission due to LVDS signals:

- The edge rates and signal skew of the LVDS signal pairs should be matched as closely as possible. Different signal rise and fall times and skew between signal pairs create common-mode noise, which, in turn, creates EMI noise. Refer to [Figure 8](#) for an example. Notice that the crossover point is not centered in the middle of the signal swing. Adjusting trace lengths of one LVDS signal in the complementary pair will align the crossover point to compensate for different signal rise and fall times along with compensating for any inherent skew in the signal pair ([Figure 8](#)).
- Maintain a spacing between the LVDS signal pairs that is less than twice the PCB trace width ( $S < 2W$ ), where S is the trace spacing and W is the PCB trace width.



**Figure 8 • Signal Edge Rate and Skew Effect**

### **Fail-Safe I/O Biasing**

Requirements for handling shorted, unused, and terminated LVDS I/Os on the board are as follows:

#### **Unused LVDS Inputs**

Float (leave open) unused LVDS inputs and place them as close to the connector pin as possible. If the LVDS receiver does not include a fail-safe resistor to keep the output high, external fail-safe resistors are required. Refer to the National Semiconductor *LVDS Owner Manual*, Chapter 4, for more information about how to specify fail-safe resistors values.

#### **Unused LVDS and TTL Outputs**

Unused LVDS and TTL outputs should be left floating to conserve power. Do not ground them.

#### **Shorted Inputs**

If the LVDS inputs to a receiver are shorted together, the receiver output must remain at stable High (positive voltage).

#### **Unconnected Receiver Inputs**

If an LVDS transmitter is driving a cable that is unused at the receiver end, it must be terminated. Alternately, if the receiver input is terminated but left unconnected to a cable, noise can be coupled into the receiver and cause its output to switch or oscillate in response to the noise source. Use external fail-safe resistors to resolve this issue.

### **Cabling Requirements**

The quality and type of cable used for LVDS signals is crucial for reducing noise on the LVDS transmission line. The following cabling guidelines are recommended:

- Each LVDS signal should be transmitted on its own co-axial connector with a ground ring. The ground ring must be connected to the ground plane on the PCB on both the transmitter and receiver sides.
- If the preceding guideline is not practical, use a twisted-pair cable or twin-axial connector with a ground shield between each connector pair. Each ground shield must be connected to the PCB's ground plane on both the transmitter and receiver sides.
- LVDS transmission is suitable for short distances (<10 meters). Keep cable lengths between transmitter and receiver as short as possible.
- Avoid using ribbon connectors or flexible circuits as LVDS connectors. If you must use them, use only very short (<12 inch) connections.
- Ground unused connector pins.
- Use low-skew and controlled-impedance connectors. For the connector, impedance should be matched to your PCB's LVDS trace impedance as closely as possible.

## Axcelerator/RTAX-S LVDS Interface

Both the Axcelerator and RadTolerant Axcelerator (RTAX-S) device families support LVDS I/Os. LVDS is available in a variety of modes such as point-to-point, bidirectional (BUS-LVDS), and multi-drop. Axcelerator and RTAX-S support the point-to-point mode.

### LVDS/Inputs and Receivers

Axcelerator and RTAX-S devices have LVDS receivers embedded in the I/O cell. Two I/O pins are reserved for each receiver. One pin is used for the positive channel input and the other for the negative channel. LVDS does not require a reference voltage. Axcelerator LVDS receivers amplify the voltage difference between their positive and negative I/O inputs and convert it to a single-ended output, which may be registered within the I/O cell or routed to logic within the core.

Axcelerator LVDS receivers, which are not current-mode receivers, conform to all the electrical specifications defined in the ANSI/TIA/EIA-644 specification. LVDS I/Os cannot be specified within the designer environment. You must configure I/Os to be LVDS inputs (LVDS receivers) by instantiating special I/O primitives called macros at the design level. A list of these macros is defined in the "I/O Requirements" section on page 11.

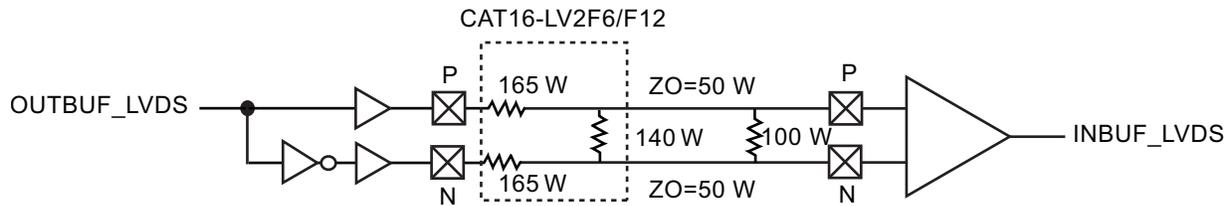
### LVDS/Output and Transmitters

Axcelerator and RTAX-S devices do not have embedded current-mode LVDS transmitters in the I/O cells. Instead, LVDS data is transmitted on two single-ended I/Os through an external resistor termination network that reproduces the required LVDS differential current and voltage. This termination network is described in the "LVDS Receiver/Transmitter Termination" section on page 10. No voltage reference setting is required for LVDS outputs. LVDS transmitters are configured by instantiating special I/O primitives in a similar manner as when configuring LVDS receivers.

### LVDS Receiver/Transmitter Termination

An LVDS point-to-point connection with Axcelerator and RTAX-S families requires termination resistors on both the transmitter and receiver sides. The LVDS receiver requires one termination resistor across the receiver's inputs. The termination resistor creates a differential voltage across the receiver and acts as a matched termination for the LVDS transmission line. The transmitter requires a resistor termination network to develop the correct voltage swing and source impedance on the transmission line. [Figure 9](#) shows a point-to-point LVDS connection as implemented in Axcelerator and RTAX-S devices.

All resistors in the connection must be matched to within 1% tolerance across the operating temperature of the device. The transmitter termination network can be implemented with the Bourns part number listed in [Figure 9 on page 11](#).



**Figure 9 • LVDS Point-to-Point Circuit in Axcelerator**

The transmitter termination network acts as a voltage divider on the difference between the positive and negative channel I/O voltages. The difference creates a voltage across the receiver's input terminals within the LVDS differential required output voltage of 250 to 450 mV.  $Z_0$  in Figure 9 shows the transmission line's characteristic impedance for both the positive and negative channels. The actual impedance of your transmission line could be different, so it is important to measure it using a TDR or other instrumentation. The receiver's termination resistance must be closely matched to be twice characteristic impedance ( $Z_0$ ) to prevent reflections on the line. As your transmission speeds increase, this impedance matching becomes very important. If you do not adjust the LVDS receiver's termination resistance, the voltage seen across the receiver's input terminals will change slightly but will still be within the LVDS differential voltage range. The termination resistor on the receiver side should not be adjusted outside the recommended 90 to 132  $\Omega$  range as defined in the ANSI/TIA-EIA-644 specification.

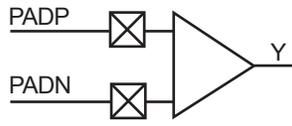
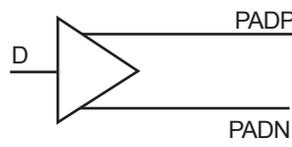
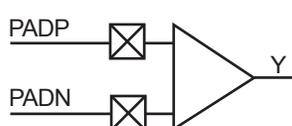
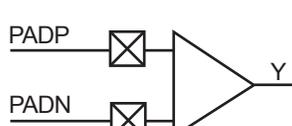
### Fail-Safe I/O Biasing

Microsemi recommends that you use external fail-safe pull-up resistors on the LVDS I/O pins of Axcelerator and RTAX-S devices to protect their LVDS receivers from shorted or floating inputs. Refer to the National Semiconductor *LVDS Owners Manual* for more details (refer to the "References" section on page 25).

### I/O Requirements

Instantiating and connecting special primitives called macros to your design logic allows you to specify LVDS I/Os. If you want an LVDS input to drive a global clock network on the device, such as RCLK or HCLK, then special macros are required. Table 1 on page 12 lists all the macros available for LVDS I/Os, including global clock inputs. Refer to the *Microsemi Antifuse Macro Library Guide* for more information.

**Table 1 • LVDS I/O Macro Names**

I/O Type	Macro Name	Macro Schematic	Descriptions
Input	INBUF_LVDS		LVDS Receiver – two I/O pins are used to receive differential data.
Output	OUTBUF_LVDS		LVDS Transmitter – two I/O pins are used to transmit single-ended data. Requires an external termination network.
RCLK	CLKBUF_LVDS		Special LVDS receiver and input buffer – two I/O pins are used to receive differential data. Receiver output is buffered on a special buffer type to be promoted to the global RCLK network in the device.
HCLK	HCLKBUF_LVDS		Special LVDS receiver and input buffer – two I/O pins are used to receive differential data. Receiver output is buffered on a special buffer type to be promoted to the global HCLK network in the device.

A simple way to instantiate any of these I/O macros automatically is to use SmartGen. Refer to the "Design Guidelines for LVDS in Axcelerator or RTAX-S Devices" section on page 13.

## LVDS I/O Channels

Devices in the Axcelerator and RTAX-S families support different numbers of LVDS I/Os. All Axcelerator and RTAX-S devices have eight I/O banks. Refer to the pin tables in the appropriate datasheets for the number of LVDS channels per I/O bank.

## DDR Support

LVDS data can be received or transmitted in Axcelerator and RTAX-S devices in DDR mode. In DDR mode, data is latched on each edge of the clock. All input cells in Axcelerator and RTAX-S devices have special DDR registers to receive DDR data. The LVDS DDR receiver can be instantiated using SmartGen. LVDS data can be transmitted out of Axcelerator or RTAX-S devices in DDR mode, but requires a special transmitter circuit and LVDS output buffer created using core logic. Additional information about creating and placing LVDS transmitters and receivers is discussed in the "Receiving LVDS Data" and the "Transmitting LVDS Data – DDR Mode" section on page 17.

## Design Guidelines for LVDS in Axcelerator or RTAX-S Devices

This section discusses guidelines for implementing an LVDS design in an Axcelerator or RTAX-S device. Some of these guidelines reference requirements about using either PLLs or global clock networks available in both devices. Refer to the [Using Global Resources in Microsemi Axcelerator Family](#) and [Axcelerator Family PLLs and Clock Management](#) application notes for more details on these requirements.

### Receiving LVDS Data

To receive LVDS data, Axcelerator offers special macros. These macros are described in [Table 1 on page 12](#). If you plan to receive LVDS data in DDR mode, you must instantiate a DDR register. Both the LVDS receiver and the DDR register are embedded in the I/O pin logic. You can use either the RCLK or HCLK network to generate the clock input for the DDR register. SmartGen will automatically create the Verilog or VHDL code for one or several LVDS receiver modules, with or without a DDR register. [Figure 10](#) shows a schematic of an LVDS receiver configured in DDR mode.

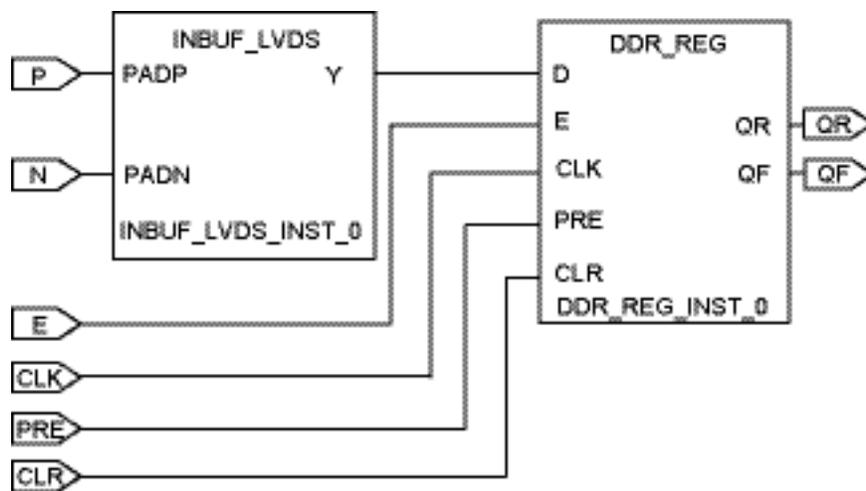
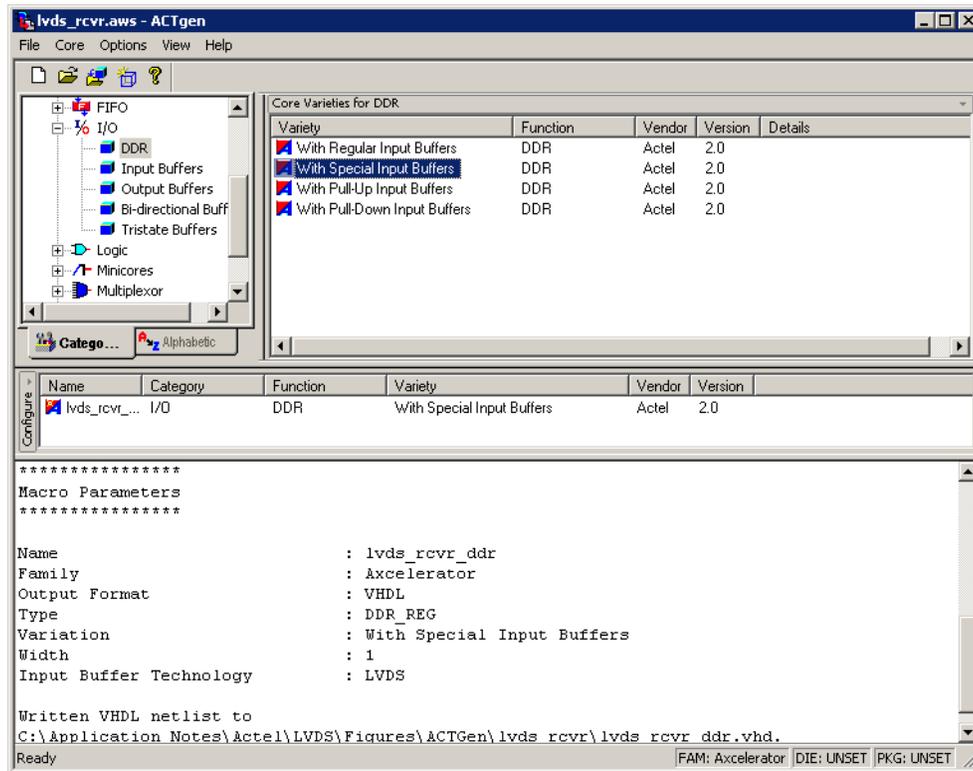


Figure 10 • DDR Input with LVDS

Figure 11 shows the SmartGen GUI settings that will create a single channel LVDS input with a DDR register for Axcelerator. The same settings can be used for RTAX-S.

The VHDL code created by SmartGen for the setting specified in Figure 11 is shown in Figure 12 on page 15. To use this code, you have to include it with your design files, instantiate the `lvds_input` module as a component in your top-level design, and connect your design's logic to the appropriate input and output ports of this module. If you do not plan to use the DDR mode, use SmartGen to specify an LVDS receiver and instantiate it in your top-level design. In this case, the SmartGen RTL will be equivalent to Figure 12 on page 15 except that the DDR Register would not be instantiated as a component.



**Figure 11 • LVDS Receiver with DDR Setting in SmartGen**

```
library ieee;
use ieee.std_logic_1164.all;
library Accelerator;

entity lvds_input is
  port(PADP, PADN, E, PRE, CLR, CLK, : in std_logic; QR, QF :
        out std_logic);end lvds_input;

architecture DEF_ARCH of lvds_input is

  component INBUF_LVDS
    port(PADP, PADN : in std_logic := 'U'; y : out std_logic
        );
  end component;
  component DDR_REG -- If non DDR Mode. This Component is Omitted.
    port(D, E, CLK, CLR, PRE : in std_logic := 'U'; QR, QF :
        out std_logic); -- Omitted in non DDR Mode
  end component ; -- Omit in non DDR Mode

  signal Y : std_logic ;
begin

  INBUF_LVDS_0_inst :INBUF_LVDS
  port map(PADP => PADP, PADN => PADN, Y => Y);
  DDR_REG_0_inst : DDR_REG -- Omitted for non DDR Mode
  port map(D => Y, E => E, CLK => CLK, CLR => CLR, PRE => PRE,
    QR => QR, QF => QF); -- Omitted for non DDR Mode
```

**Figure 12 • SmartGen RTL for LVDS Receiver with DDR Register**

## Transmitting LVDS Data – Non-DDR Mode

If LVDS data is being transmitted in non-DDR mode, use SmartGen to create an LVDS transmitter module, include the SmartGen RTL in your design, and instantiate the transmitter as a component in your top-level design. [Figure 13 on page 16](#) shows the SmartGen settings to implement a single channel LVDS transmitter. A schematic of this module is also shown as the OUTBUF\_LVDS macro in [Table 1 on page 12](#).

Place your design logic as closely as possible to the D-input of the LVDS output macro. This is necessary to meet the timing requirement for high-speed (> 300 MHz) data transmission. If a register drives the LVDS transmitter, place it within an LVDS output pin. This provides the shortest clock-to-output timing for the LVDS output and, if used for several outputs, helps minimize channel skew between them.

Use Designer's I/O Attribute Editor to embed a register into an I/O pin. Select the **Use I/O Reg** check box for one LVDS output, as shown in [Figure 14 on page 16](#). The I/O register should be included for one output since the LVDS transmitter macro has one input but uses two output pins. Designer's place-and-route tool will analyze the connectivity of the D-input of the OUTBUF\_LVDS macro and automatically place the driving register within the I/O cell specified with the I/O Attribute Editor.

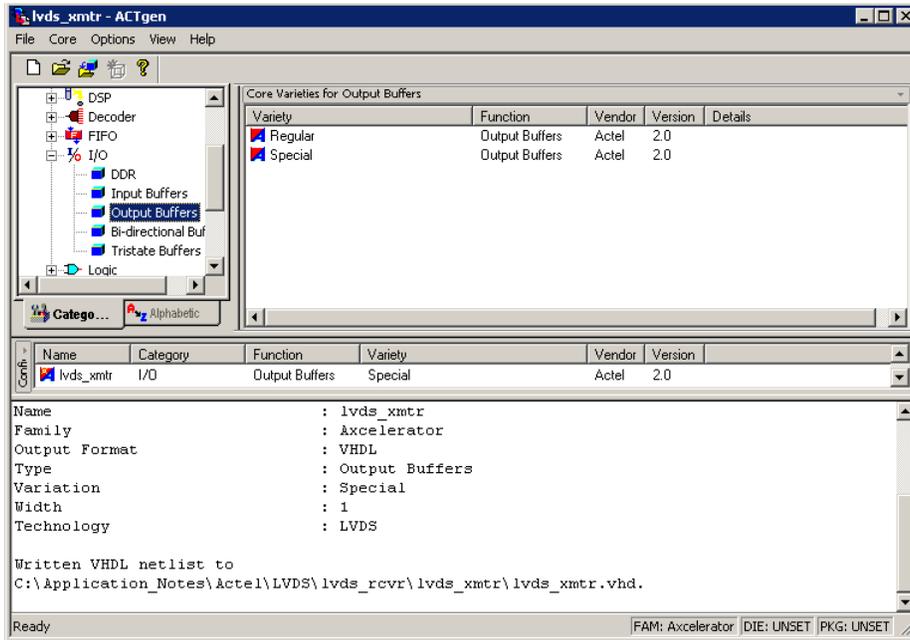


Figure 13 • SmartGen Settings for an LVDS Output Buffer

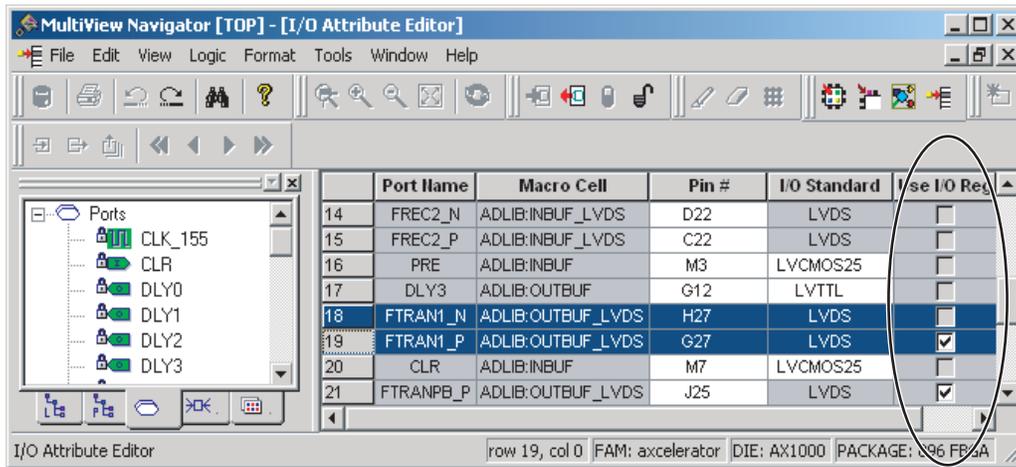


Figure 14 • Specifying I/O Register with an LVDS Transmitter Macro

## Transmitting LVDS Data – DDR Mode

If you plan to transmit LVDS data in DDR mode, a special transmitter circuit should be used to convert DDR output data (shown in Figure 15). Two variations of this circuit are proposed to fix a potential race condition. They are described in the "Pipelined DDR Inputs" section on page 17 and the "Registered DDR Output" section on page 20. Microsemi recommends using the Pipelined DDR Input version.

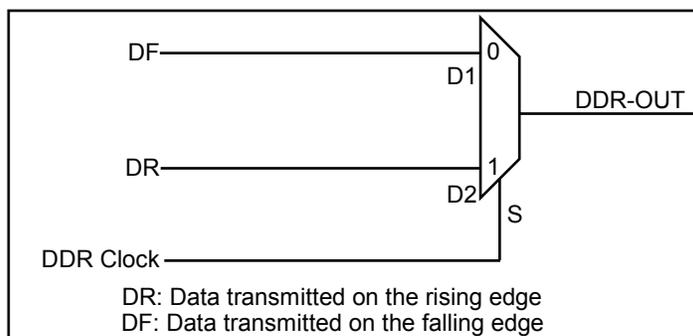


Figure 15 • DDR Multiplexer Circuit

### Basic Transmitter Circuit

To transmit DDR data, the DDR register needs to be connected to a multiplexer that selects either the falling edge (DF output) or rising edge (DR output) clock data. The clock used on the DDR register is also used for the multiplexer's select input.

The HCLK global clock network in Axcelerator and RTAX-S devices feeds only sequential (R-cells) in the device; therefore, the RCLK global clock network must be used for the DDR clock signal, since it connects to the multiplexer. Refer to the *Axcelerator Family FPGAs* and *RTAX-S Family FPGAs* datasheets or the *Using Global Clock Networks in Axcelerator Devices* application note for details about how to specify and use these global clock networks.

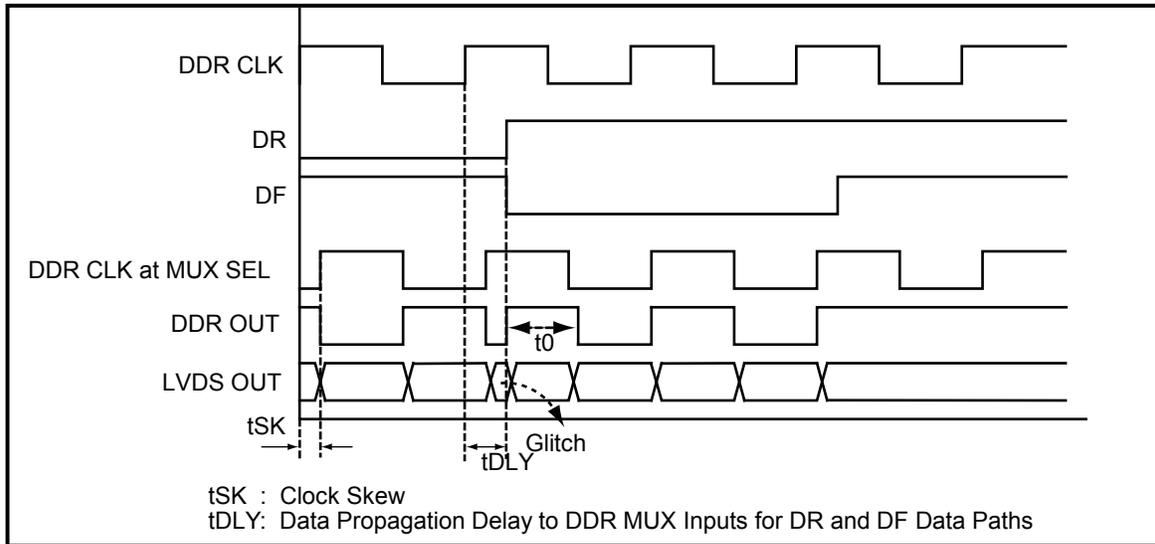
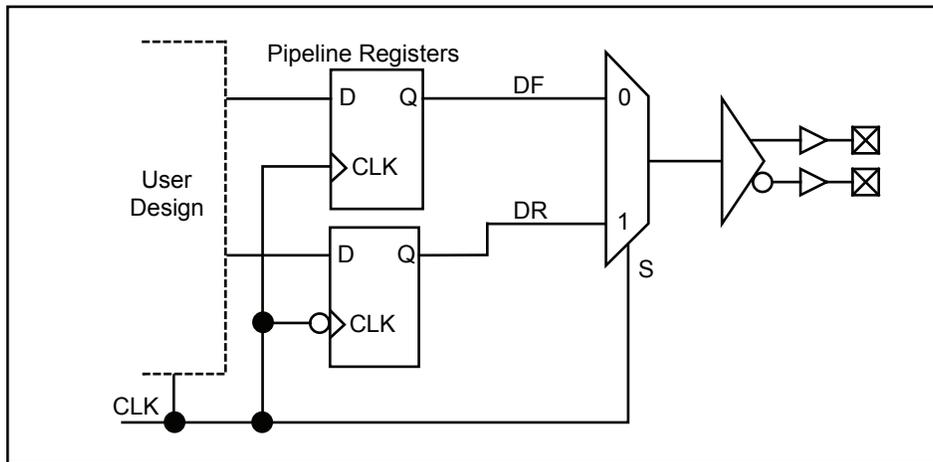
This circuit may generate glitches on the DDR-OUT signal because of a possible race condition between the DF and DR signals and the DDR clock input. Figure 16 on page 18 is a timing diagram that illustrates the race condition. If the data delay between the DDR clock pin and the DDR multiplexer select input is less than the propagation delay of either DR or DF to the multiplexer inputs, then the multiplexer may select data for the wrong input (DR is shown in Figure 16 on page 18). This will cause a glitch on the DDR OUT signal.

To create a glitch-free DDR output, either the DR and DF inputs can be registered before they feed the multiplexer, or the multiplexer output can be registered. These circuit modifications are the subject of the following two sections.

### Pipelined DDR Inputs

The circuit in Figure 17 on page 18 can be used, if the DF and DR multiplexer inputs are registered.

The pipeline registers labeled in Figure 17 on page 18 latch the multiplexer data one-half clock cycle before the data is selected by the multiplexer. For example, the DR data would be latched on the falling edge of the clock to be valid, and would be selected by the MUX on the next rising edge. The reverse is true for the DF data.


**Figure 16 • DDR Glitch Timing Diagram**

**Figure 17 • Pipelined DDR Circuit**

### Timing Requirements

To prevent a glitch on the DDR output, the slack for the paths connecting the pipeline registers and the multiplexer input must be greater than or equal to zero. This slack is defined by EQ 5:

$$\text{Slack} = (\text{DDR Clock Period}/2) - (\text{Propagation Delay between Pipeline Register Q Output and MUX Input})$$

EQ 5

You can use Timer to calculate this propagation delay. Figure 18 shows a path between one pipeline register and the multiplexer.

The propagation delay from the clock input of the Pipeline register in this path to the MUX input is 0.92 ns. Therefore, for positive slack, the DDR clock period must be greater than 1.84 ns.

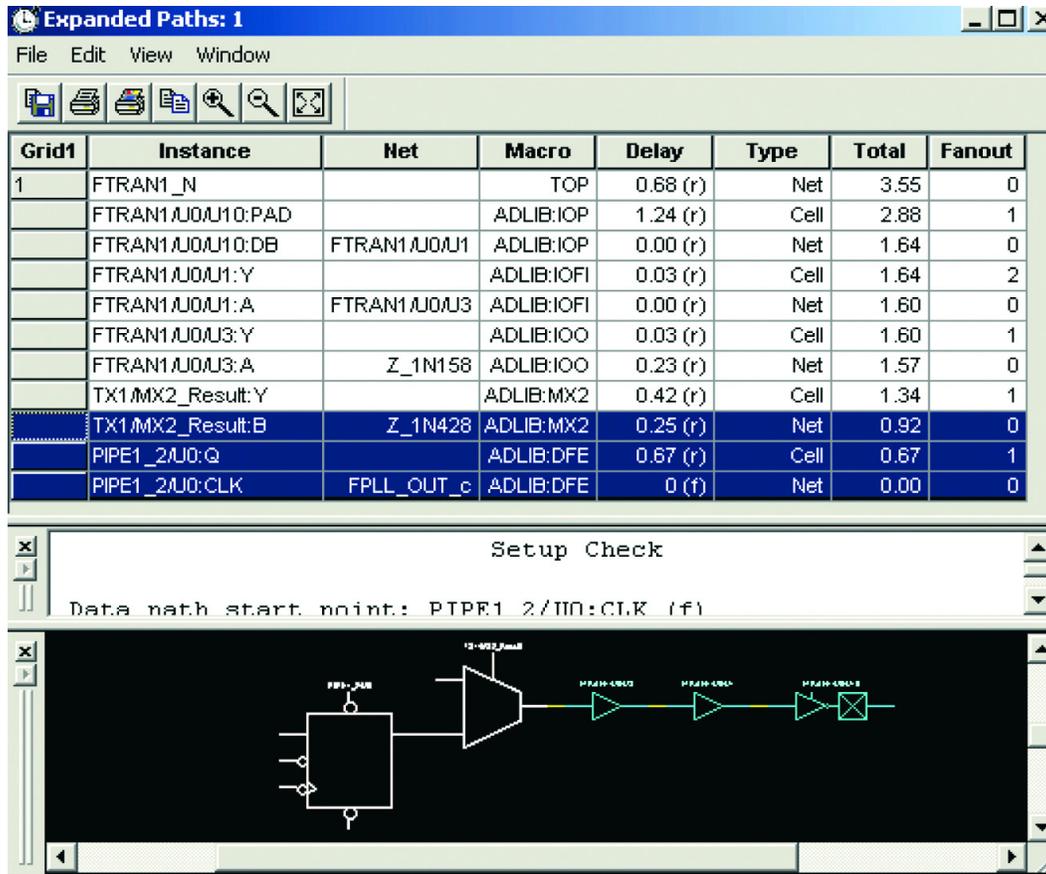


Figure 18 • Propagation Delay Path in Timer

### Placement Guidelines

If you are transmitting DDR data on several LVDS channels, the amount of channel skew is important and should be minimized. To reduce channel skew, the DDR multiplexer for each channel must be placed as close as possible to its LVDS output. Figure 19 shows an example of the recommended placement.

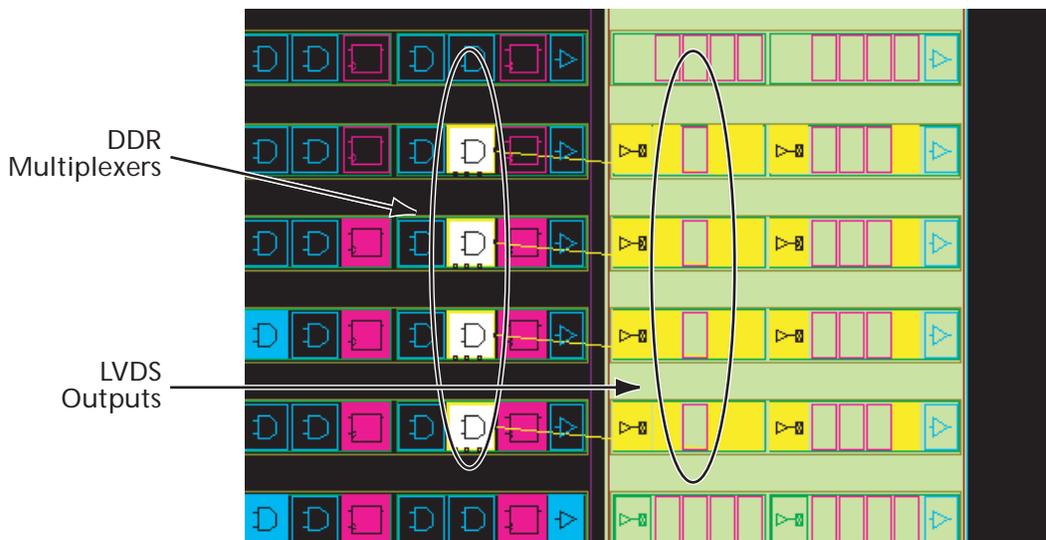


Figure 19 • Pipeline DDR Placement

### Registered DDR Output

Alternately, you can register the output of the DDR MUX to prevent the glitch from occurring. This circuit is shown in Figure 20.

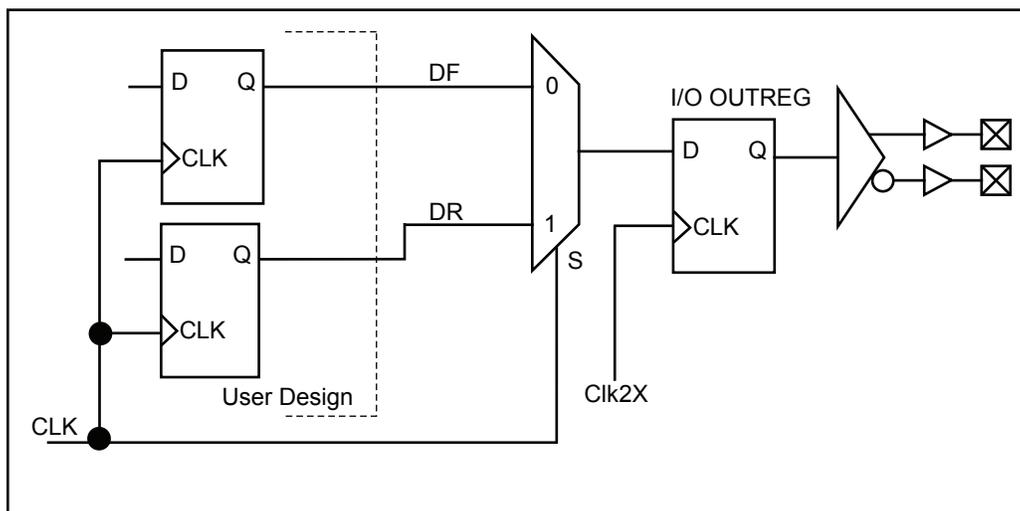


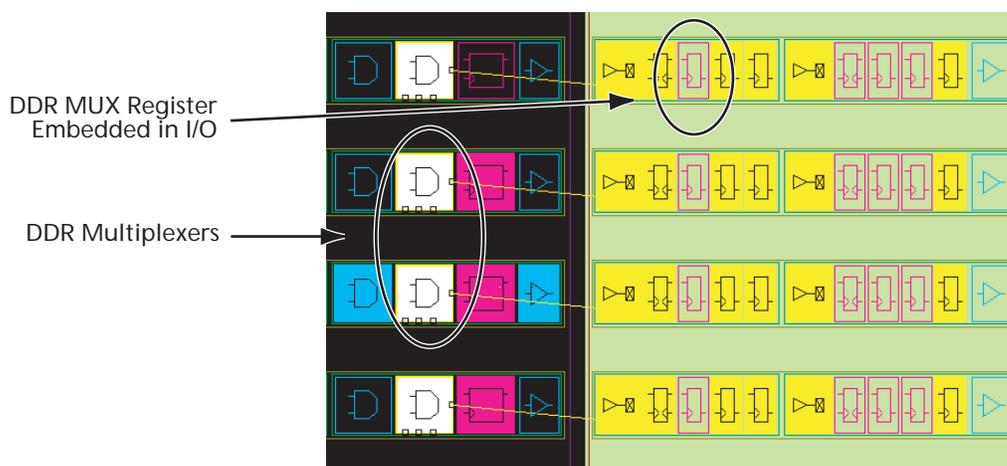
Figure 20 • Registered DDR Output

## Placement Guidelines

In this circuit, two clocks are needed: one for the MUX select input, labeled Clk, and one for the output register Clk2x. Clk2x must be twice the frequency of the Clk signal. Some placement guidelines are listed below:

- The Clk signal must be placed on the RCLK network since it feeds a combinatorial cell.
- The Pipeline register, labeled I/O OUTREG in [Figure 20 on page 20](#), must be placed within the LVDS I/O pin. Follow the recommendation given in the "Transmitting LVDS Data – Non-DDR Mode" section on [page 15](#). This helps to reduce channel skew between the LVDS outputs since all outputs will have the same clock-to-output delay. [Figure 21](#) shows an example of this placement.

The Clk2x signal can be connected to either the HCLK or RCLK network. However, the HCLK network is preferred. If you are using an Axcelerator device, use SmartGen to create this clock. [Figure 22 on page 22](#) shows the SmartGen PLL settings to create the clock. If the Clock Out parameter is set to **Hardwired**, the PLL output will drive the HCLK network; otherwise, the RCLK network will be used.



**Figure 21 • Registered DDR Placement**

## Timing Guidelines

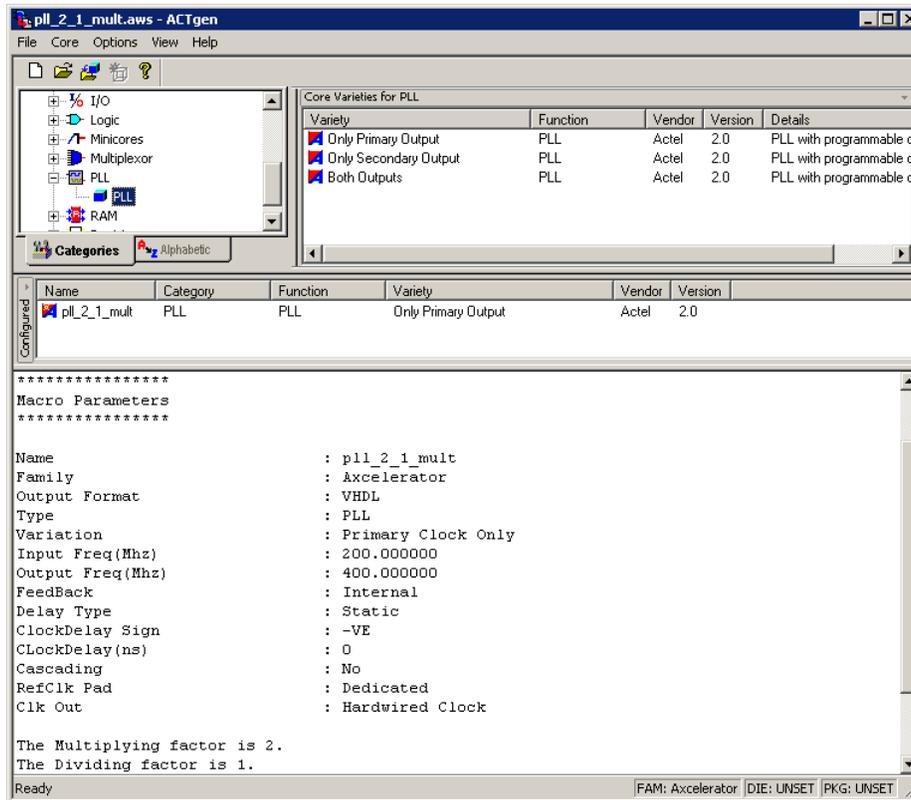
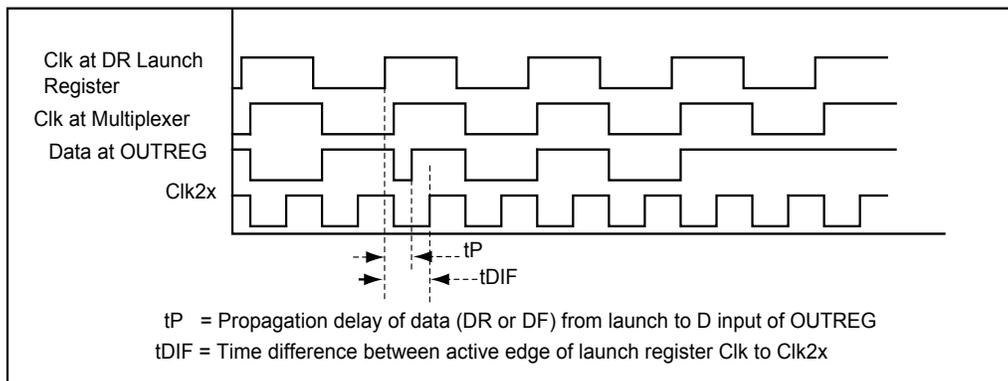
Timing requirements for the circuit are shown in [Figure 23 on page 22](#).

- To reach the maximum data rate, the I/O register should be placed within the LVDS output pin.
- To prevent a glitch on the MUX output, the Clk2x signal should transit within the time window marked as  $t_D$  in [Figure 16 on page 18](#), minus the setup time ( $t_{SU}$ ) of the register embedded within the I/O pin. To meet this requirement, specify the Clk2k period to meet the condition in [EQ 6](#):

$$\text{Clk2x period} \geq t_{DIF} > t_P + t_{SU}$$

EQ 6

Where  $t_{DIF}$  and  $t_P$  are shown in [Figure 23 on page 22](#).


**Figure 22 • SmartGen PLL Settings for Clk2x Signal**

**Figure 23 • Clk2x Period Timing Requirements**

If Clk2x is generated through a PLL, the programmable delay element in the PLL can be adjusted to delay the CLK2x signal. The Clk2x signal must be delayed by a value equal to or greater than  $t_p + t_{SU}$ , compared with the Clk signal. Use Timer to calculate the delay between the register that launches either the DF or DR data and the I/O\_OUTREG register, employing the same method as discussed in the "Timing Requirements" section on page 19.

The I/O\_OUTREG setup time,  $t_{SU}$ , can be obtained from the *Axcelerator Family FPGAs* datasheet. After you have calculated the total delay, use SmartGen to specify it as a PLL setting and instantiate the PLL in your design.

Make sure that you calculate the worst-case value for the delay by checking its value under worst-case operating conditions.

## PLL Placement for LVDS Transmit and Receive

The PLLs in Axcelerator drive different clock networks based on their location on the device. If the PLLs are located on the north side of the die, they can drive the HCLK network or a regular net. If the PLLs are located on the south side, they can drive the RCLK network or a regular net. This implies that if you plan to use a PLL to generate LVDS transmit or receive clocks, you should use one of the four PLLs located on the south side of the die.

## Conclusion

This application note provided an overview of the LVDS standard, focusing on its electrical and timing requirements and listing many PCB layout guidelines. Next, unique characteristics of LVDS implementation in Axcelerator and RTAX-S families were presented. It concluded with detailed recommendations for instantiating LVDS transmitter and receivers in your design, and provides guidelines for creating LVDS receivers and transmitters with DDR circuits. Many of the PCB recommendations listed in this application note are derived from the National Semiconductor *LVDS Owner Manual*.

## List of Changes

The following table lists critical changes that were made in the current version of the document.

Revision	Changes	Page
Revision 4 January 2014	Figure 10 is updated.	13
Revision 3 November 2012	The <a href="#">LVDS Advantages</a> and <a href="#">LVDS Electrical Specifications</a> sections were removed (SAR 41932).	NA
Revision 2 May 2012	The " <a href="#">Timing Guidelines</a> " section was revised (SAR 21620).	21
Revision 1 October 2004	The following statement was added to " <a href="#">LVDS/Inputs and Receivers</a> ": "LVDS does not require a reference voltage".	10
	The following statement was added to " <a href="#">LVDS I/O Channels</a> " Refer to the pin tables in the appropriate datasheets for the number of LVDS channels per I/O bank.	12
	Table 4: LVDS Channel Count was deleted. An accurate LVDS channel count can be found in the <a href="#">Axcelerator Family FPGAs</a> or <a href="#">RTAX-S RadTolerant FPGAs</a> datasheets.	N/A

## Related Documents

### Datasheets

*Axcelerator Family FPGAs*

[www.microsemi.com/soc/documents/AX\\_DS.pdf](http://www.microsemi.com/soc/documents/AX_DS.pdf)

*RTAX-S RadTolerant FPGAs*

[www.microsemi.com/soc/documents/RTAXS\\_DS.pdf](http://www.microsemi.com/soc/documents/RTAXS_DS.pdf)

### Application Notes

*Using Global Clock Networks in Axcelerator Devices*

[www.microsemi.com/soc/documents/AX\\_Global\\_Resources\\_AN.pdf](http://www.microsemi.com/soc/documents/AX_Global_Resources_AN.pdf)

*Axcelerator Family PLLs and Clock Management*

[www.microsemi.com/soc.com/documents/AX\\_PLL\\_AN.pdf](http://www.microsemi.com/soc.com/documents/AX_PLL_AN.pdf)

### User's Guides

*Antifuse Macro Library Guide*

[www.microsemi.com/soc/documents/genguide\\_UG.pdf](http://www.microsemi.com/soc/documents/genguide_UG.pdf)

*ChipEditor User's Guide*

[www.microsemi.com/soc/documents/chipeditor\\_UG.pdf](http://www.microsemi.com/soc/documents/chipeditor_UG.pdf)

*Timer User's Guide*

[www.microsemi.com/soc/documents/timer\\_UG.pdf](http://www.microsemi.com/soc/documents/timer_UG.pdf)

## References

*LVDS Owner Manual* - National Semiconductor, Chapters 1, 4, and 5

Chapter 1: [http://www.national.com/appinfo/lvds/files/lvds\\_ch1.pdf](http://www.national.com/appinfo/lvds/files/lvds_ch1.pdf)

Chapter 4: [http://www.national.com/appinfo/lvds/files/lvds\\_ch4.pdf](http://www.national.com/appinfo/lvds/files/lvds_ch4.pdf)

Chapter 5: [http://www.national.com/appinfo/lvds/files/lvds\\_ch5.pdf](http://www.national.com/appinfo/lvds/files/lvds_ch5.pdf)

*AN-1059 High Speed Transmission with LVDS Link Devices* - National Semiconductor

[www.national.com/an/AN/AN-1059.pdf](http://www.national.com/an/AN/AN-1059.pdf)

*ANSI/TIA/EIA-644 LVDS Electrical Interface Standard*

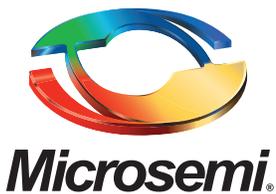
[www.tiaonline.org/standards/](http://www.tiaonline.org/standards/)

*AN-1040, LVDS Performance: Bit Error Rate (BER) Testing Test Report #2*, National Semiconductor

[www.national.com/an/AN/AN-1040.pdf](http://www.national.com/an/AN/AN-1040.pdf)

*IEEE 1596.3 LVDS Standard for Scalable Coherent Interface (SCI)*

[http://standards.ieee.org/reading/ieee/std\\_public/description/busarch/1596.3-1996\\_desc.html](http://standards.ieee.org/reading/ieee/std_public/description/busarch/1596.3-1996_desc.html)



**Microsemi Corporate Headquarters**  
One Enterprise, Aliso Viejo CA 92656 USA  
Within the USA: +1 (949) 380-6100  
Sales: +1 (949) 380-6136  
Fax: +1 (949) 215-4996

Microsemi Corporation (NASDAQ: MSCC) offers a comprehensive portfolio of semiconductor solutions for: aerospace, defense and security; enterprise and communications; and industrial and alternative energy markets. Products include high-performance, high-reliability analog and RF devices, mixed signal and RF integrated circuits, customizable SoCs, FPGAs, and complete subsystems. Microsemi is headquartered in Aliso Viejo, Calif. Learn more at [www.microsemi.com](http://www.microsemi.com).

© 2012 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.