Order this document by IND_DVI_AN

GigaSTaR®
Digital Display Link

Interfacing Between GigaSTaR DDL and DVI/LVDS

Revision 1.0

May 2004

Contents

1 Introduction

2 Interfacing LVDS / DVI to DDL Transmitter

3 Interfacing DDL Receiver to LVDS / DVI

4 Pixel Interface Mapping List

1 Introduction

The GigaST★R Digital Display Link (DDL) is a powerful, serial long-distance mutlimedia link. Video, audio and digital sideband data are serialized and can be transmitted via standard STP cables up to 50 m. Distances up to 1000 meters are achievable with fiber-optic cables.

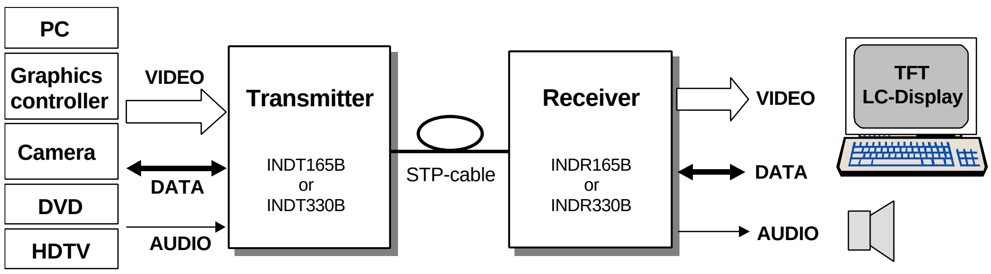

Figure 1.0 GigaSTaR Digital Display Link

Today's popular DVI- and LVDS Digital Graphic Links transmit graphics and clock signals through multiple cable pairs over distances up to 10 meters maximum.

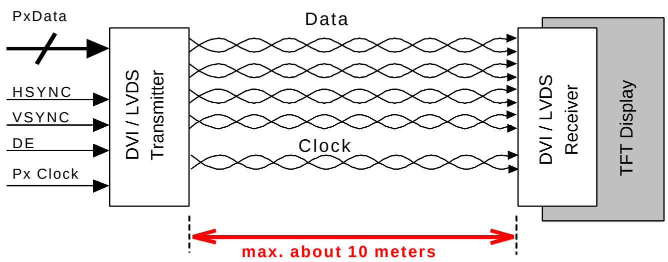

Figure 1.1 Transmitting Digital Video Data Using LVDS / DVI Standard Devices

The DDL transmits the embedded data and clock signal over two independant twisted cable pairs thus typically eliminates 4 or 8 twisted pair data cables. As a result long distances between transmitter and receiver can be established by standard twisted pair cables. In addition, the bitserial CML signals allow easy interfacing to optical transceivers. A built-in elastic buffer in the DDL transmitter allows the DDL link system to cope with almost any clock jitter of the graphics source (see **figure 1.2**).

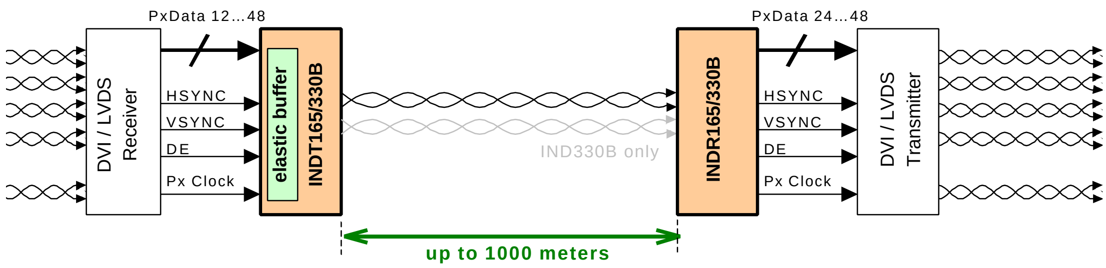

Figure 1.2 Extending LVDS / DVI using GigaSTaRÒ Digital Display Link Devices

This application note demonstrates how to use the GigaSTaR Digital Display Link in combination with popular DVI- and LVDS Display links to extend the distance in between a video source (graphics card) and the display or beamer (projector).

2 Interfacing LVDS / DVI to DDL Transmitter

The DDL transmitter devices can be configured to interface with 12, 18, 24, 36 or 48 bit pixel formats (half, full or double pixel single-ended). The pixel clock active edge can be set to rising, falling or both. The flexibility of the parallel interface allows an easy interfacing to popular LVDS/DVI-to-TTL translator devices as depicted **figure 2.1**.

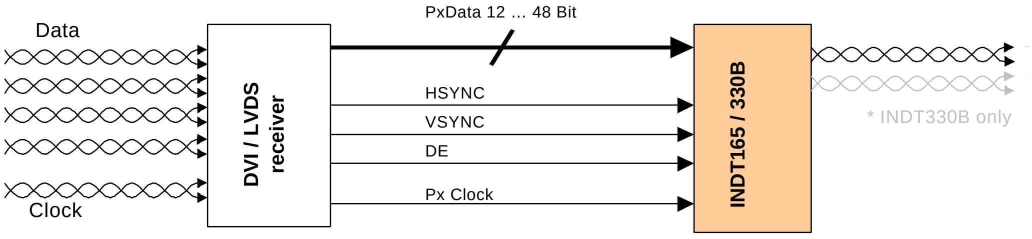

Figure 2.1 Interfacing LVDS or DVI to GigaSTaR® Digital Display Link Transmitter

For correct functionality the DDL transmitter requires active graphic control signals (HSYNC, VSYNC and DE) with positive or negative polarity on HSYNC and VSYNC and positive polarity on DE. The timing has to maintain a valid horizontal and vertical back porch timing (e.g. **figure 2.2**).

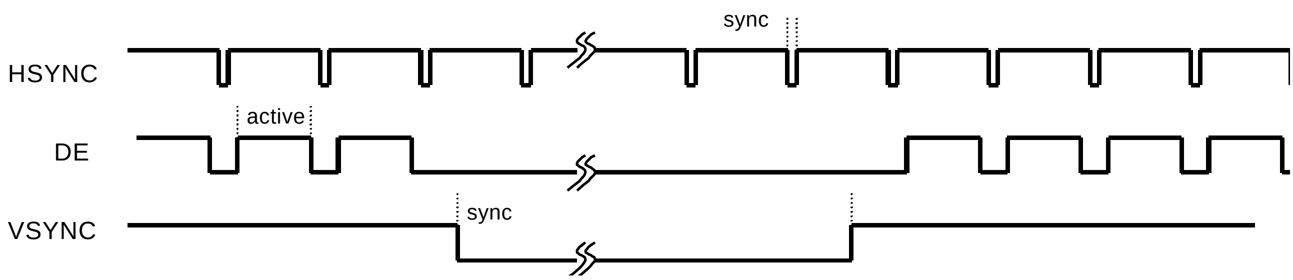

Figure 2.2 GigaSTaR® Digital Display Link Transmitter Timing Requirements

Note :

To use the IND165B for VESA XGA resolution (1024x768 pixel, 24 Bit @ 60Hz refresh) the reference clock frequency must be 66.67 MHz.

3 Interfacing DDL Receiver to LVDS / DVI

At lower clock frequencies and resolution, DDL's flexible application interface can be configured to match all popular parallel video interfaces. Basically the DDL receiver devices can be directly connected to all LVDS or DVI transmitter devices (**figure 3.1**).

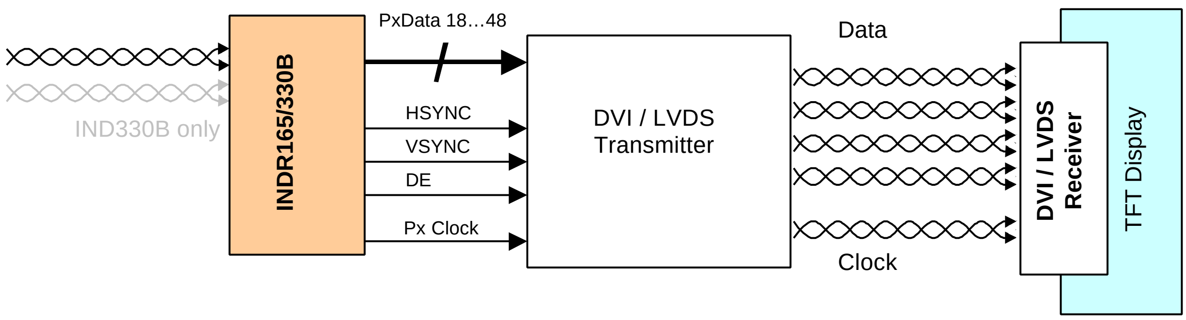

Figure 3.1 Interfacing DDL Receiver to LVDS / DVI Transmitter

DVI and LVDS translator chips include PLL circuits to provide a solid internal clock. These PLL circuits usually have an integrating behaviour to smooth the incoming pixel clock and to create a stable clock for the serial data transmission (**figure 3.2**).

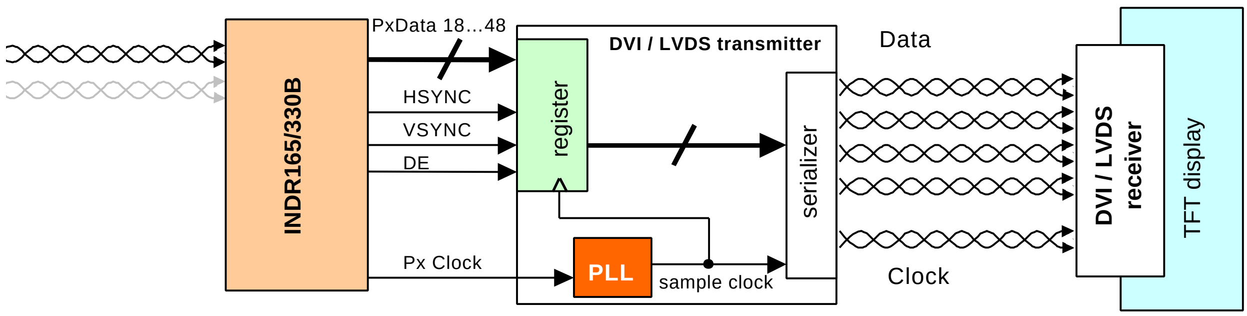

Figure 3.2 Internal Operation of LVDS / DVI Devices

In some cases, especially at higher clock frequencies and resolutions, this PLL-characteristic may conflict with the clock generation of the DDL in a way that loss of pixel data may occur.

Technical Background

In order to eliminate the impact of the pixel clock's quality to the link performance, the DDL utilizes a clock system which is independent of the pixel clock (**figure 3.3**).

Therefore the DDL receiver features a fractional clock synthesis where each pixel frequency can be generated as an average of two discrete frequencies $f_{pixRXout1}$ and $f_{pixRXout2}$. The minimum step between the two frequencies is 1/150ps - the minimum switching period is 256 clocks. The pixel clock synthesizer is directly controlled by the transmitter device.

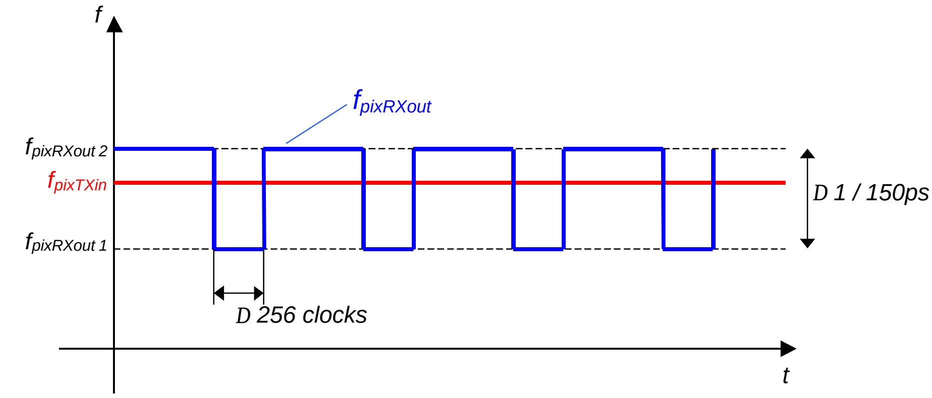

Figure 3.3 Pixel Clock Frequency Synthesis at INDR165/330B Receiver

This synthesized Rx DDL clock is used to drive the DVI / LVDS PLL (**figure 3.4**). The PLL circuitry in the DVI / LVDS Tx device will follow the frequency of the synthesized DDL pixel clock.

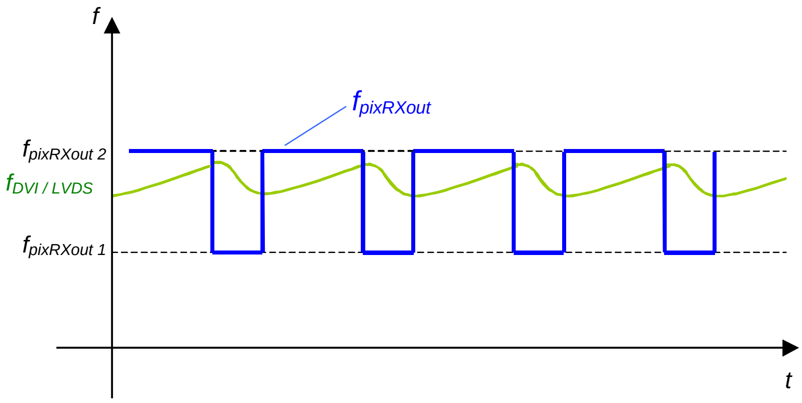

Figure 3.4 PLL Circuits in DVI / LVDS Devices Follows Pixel Clock Frequency

Because of the PLL characterictics of the DVI / LVDS PLLs, the clock generated within the DVI/LVDS devices may vary over time from the pixel clock (and pixel data) provided by the DDL receiver. As LVDS and DVI devices do not provide elastic buffer at their inputs to compensate potential timing mismatch between their external and internal clock system, data losses or bit errors can occur. The effect becomes more evident with increasing pixel clock frequencies. At low frequencies this effect decreases the available setup- and hold-time budgets at the pixel interface (**figure 3.5**).

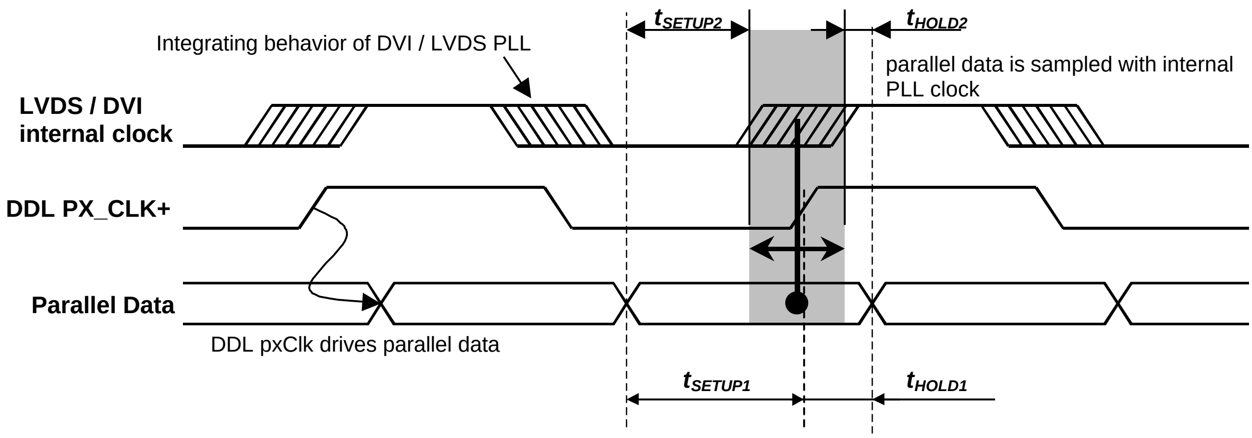

Figure 3.5 Decreased Setup and Hold Times by Mismatched Clock Frequencies

Pixel clocks above SVGA resolution (> 40 MHz) may cause data loss (**Figure 3.6**), therefore an elastic buffer shall be implemented between the DDL's receiver and DVI or LVDS transmitter. Also, if signal repeating to multiple displays (daisy-chaining) is required, elastic buffering is mandatory.

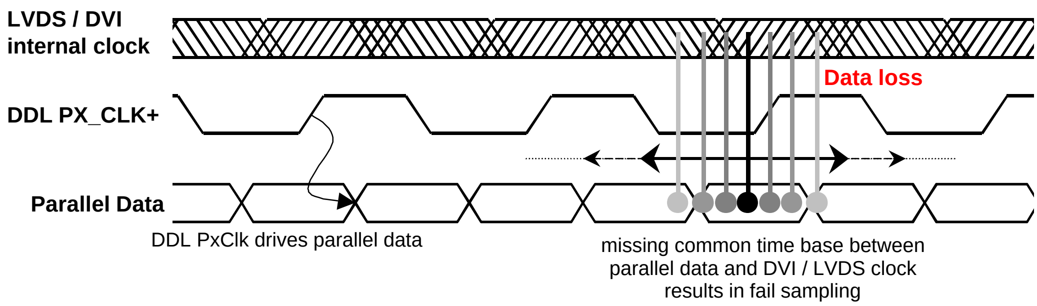

Figure 3.6 Fail Sampling by Mismatched Clock Frequencies

For pixel clock frequencies above SVGA/XGA-resolutions an elastic buffer with pixel clock PLL between the DDL receiver and the DVI- or LVDS transmitter provides a clock system suitable for DVI/LVDS devices.

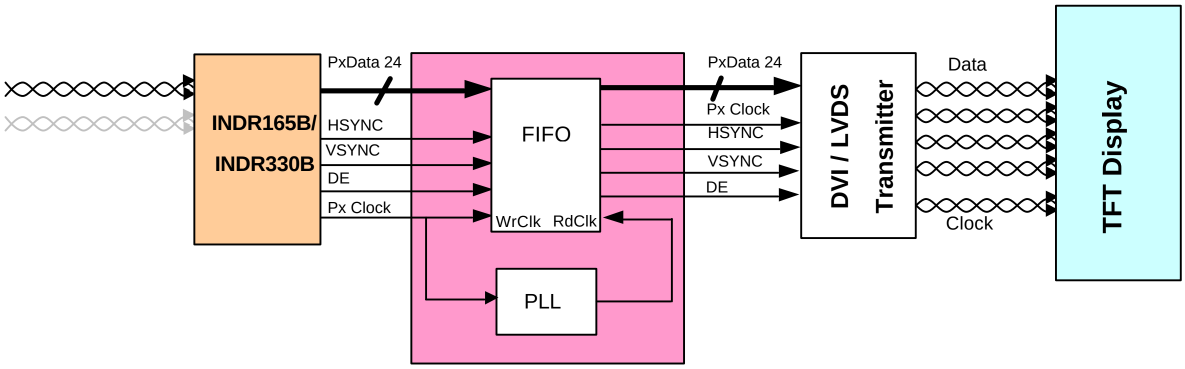

Figure 3.7 FIFO/PLL Concept

The parallel pixel data and clock (Write Clock) of the DDL receiver are feed to a FIFO. The discrete pixel clock fractions are averaged with a PLL.The smoothed pixel clock of the PLL reads the FIFO. Pixel data together with the belonging Pixel Clock are provided at the FIFO interface (output).

Implementation of Elastic Buffer:

Figure 3.8 shows a possible realization of an elastic buffer. A CPLD may be used to implement the FIFO functionality including the clock divider required for the PLL.

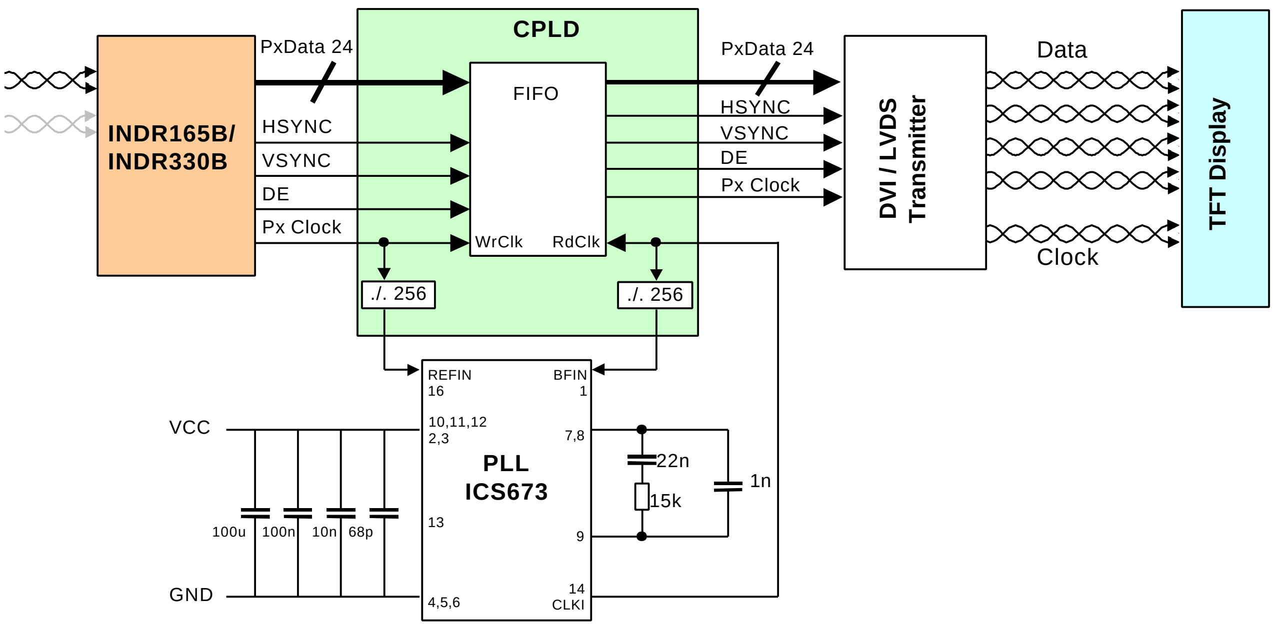

Figure 3.8 Interfacing DDL Receiver to LVDS or DVI Using Elastic Buffering

4 Pixel Interface Mapping List

The following table provides information how to connect DDL IND165/330B to various DVI- and LVDS-translator devices running at 1 pixel/clock mode. The IND165/330B data mapping for less than 24 Bit per pixel interfaces is MSB justified. For more information please refer to the IND165/330B data sheet.

DC90CF386 output pin 18 bpp	DC90CF386 output pin 24 bpp	SIL151/161 output pin 18 bpp	SIL151/161 output pin 24 bpp	INDT165/330B input pin	INDT165/330B output pin	SIL150/160 input pin 24 bpp	SIL150/160 input pin 18 bpp	DC90C385 input pin 24 bpp	DC90C385 input pin 18 bpp
-	RxOUT0	-	QE0	PX_D0	PX_D0	DIE0	-	TxIN0	-
-	RxOUT1	-	QE1	PX_D1	PX_D1	DIE1	-	TxIN1	-
RxOUT0	RxOUT2	QE2	QE2	PX_D2	PX_D2	DIE2	DIE2	TxIN2	TxIN0
RxOUT1	RxOUT3	QE3	QE3	PX_D3	PX_D3	DIE3	DIE3	TxIN3	TxIN1
RxOUT2	RxOUT4	QE4	QE4	PX_D4	PX_D4	DIE4	DIE4	TxIN4	TxIN2
RxOUT3	RxOUT6	QE5	QE5	PX_D5	PX_D5	DIE5	DIE5	TxIN6	TxIN3
RxOUT4	RxOUT27	QE6	QE6	PX_D6	PX_D6	DIE6	DIE6	TxIN27	TxIN4
RxOUT6	RxOUT5	QE7	QE7	PX_D7	PX_D7	DIE7	DIE7	TxIN5	TxIN6
-	RxOUT7	-	QE8	PX_D8	PX_D8	DIE8	-	TxIN7	-
-	RxOUT8	-	QE9	PX_D9	PX_D9	DIE9	-	TxIN8	-
RxOUT7	RxOUT9	QE10	QE10	PX_D10	PX_D10	DIE10	DIE10	TxIN9	TxIN7
RxOUT8	RxOUT12	QE11	QE11	PX_D11	PX_D11	DIE11	DIE11	TxIN12	TxIN8
RxOUT9	RxOUT13	QE12	QE12	PX_D12	PX_D12	DIE12	DIE12	TxIN13	TxIN9
RxOUT12	RxOUT14	QE13	QE13	PX_D13	PX_D13	DIE13	DIE13	TxIN14	TxIN12
RxOUT13	RxOUT10	QE14	QE14	PX_D14	PX_D14	DIE14	DIE14	TxIN10	TxIN13
RxOUT14	RxOUT11	QE15	QE15	PX_D15	PX_D15	DIE15	DIE15	TxIN11	TxIN14
-	RxOUT15	-	QE16	PX_D16	PX_D16	DIE16	-	TxIN15	-
-	RxOUT18	-	QE17	PX_D17	PX_D17	DIE17	-	TxIN18	-
RxOUT15	RxOUT19	QE18	QE18	PX_D18	PX_D18	DIE18	DIE18	TxIN19	TxIN15
RxOUT18	RxOUT20	QE19	QE19	PX_D19	PX_D19	DIE19	DIE19	TxIN20	TxIN18
RxOUT19	RxOUT21	QE20	QE20	PX_D20	PX_D20	DIE20	DIE20	TxIN21	TxIN19
RxOUT20	RxOUT22	QE21	QE21	PX_D21	PX_D21	DIE21	DIE21	TxIN22	TxIN20
RxOUT21	RxOUT16	QE22	QE22	PX_D22	PX_D22	DIE22	DIE22	TxIN16	TxIN21
RxOUT22	RxOUT17	QE23	QE23	PX_D23	PX_D23	DIE23	DIE23	TxIN17	TxIN22
RxOUT24	RxOUT24	HSYNC	HSYNC	PX_HSYNC	PX_HSYNC	HSYNC	HSYNC	TxIN24	TxIN25
RxOUT25	RxOUT25	VSYNC	VSYNC	PX_VSYNC	PX_VSYNC	VSYNC	VSYNC	TxIN25	TxIN26
RxOUT26	RxOUT26	DE	DE	PX_DE	PX_DE	DE	DE	TxIN26	TxIN27
RxCLKOUT	RxCLKOUT	ODCLK	ODCLK	PX_CLK+	PX_CLK+	IDCLK	IDCLK	TxCLKIN	TxCLKIN

Table 4.1 Connecting IND165/330B Devices to Standard DVI / LVDS devices for 24-/18-bit @ 1-pixel/clock

Inova Semiconductors GmbH

Grafinger Str. 26

D-81671 Munich, Germany

Phone: +49 (0)89 / 45 74 75 - 60

Fax: +49 (0)89 / 45 74 75 - 88

E-mail: info@inova-semiconductors.de

URL: http://www.inova-semiconductors.com