



Application Note

Parallel Interface

Revision 1.0

November 2000

Table of Contents:

INTRODUCTION.....3

BASIC INTERFACE FUNCTIONALITY4

Signal Definition..... 4
 Transmitter – INGT165B 4
 Receiver – INGR165B..... 5

Terms and Definition 5

Block Diagram 6
 Transmitter – INGT165B 6
 Receiver – INGR165B 6

Transmission Modes 7
 Burst 7
 Single Word 7
 Sideband Signaling..... 7
 Plesynchronous Transmission..... 8

INTERFACE APPLICATION EXAMPLES 9

FIFO Interface 9

Video Interface 10

Repeater for Display Data Distribution 11

Scaling GigaSTAR™ Channels..... 13

List of Illustrations:

Figure 1 Transmitter and Receiver Parallel Interface..... 6

Figure 2 Functional Burst Timing 7

Figure 3 Functional Single Word Timing 7

Figure 4 Functional FlagI / FlagO Timing 7

Figure 5 Plesynchronous Timing 8

Figure 6 RDCLK and WRCLK Timing at Ple-Synchronous Operation 8

Figure 7 Fifo Interface 9

Figure 8 Video Interface 10

Figure 9 Repeater for Display Data Distribution..... 11

Figure 10 Repeater System Setup 11

Figure 11 Cascaded Link Timing..... 12

Figure 12 Dual GigaSTAR™ Link Set-up 13

Introduction

Inova Semiconductors GigaSTAR™ devices are building blocks for a robust high-speed low-latency serial transmission link. The purpose of the link is to provide a universal data transmission platform for all types of applications.

Here are some examples: all types of pixel data transmission in industrial system such as high speed printers or copy machines, the whole spectrum of digital video data distribution networks for info terminals, daylight display walls or remote panel applications, also all types of sensor and actuator data distribution for industrial systems, bus coupling and high performance CPU interconnects.

As this list can easily be expanded, it is obvious that the variety of different applications demands a flexible and universal data interface, which can easily be adapted to all types of applications.

To support a wide spectrum of different applications, the GigaSTAR™ features:

- **no external coding** omits external high speed coding and decoding logic
- **36 bit parallel TTL data interface** enables direct bus interfacing
- **parallel interface speeds of maximum 33 MHz** simplify board layouts
- **low latency data transfer initiation** for maximum single word transfer performance
- **unlimited burst support** enables continuous data stream transfers
- **data rate adaptation** to support lower data rates requiring no external logic
- **transmission error detection** to handle physical layer errors
- **multi channel synchronization** with shared reference clock system

A prerequisite for any high performance serial data transmission is a jitter free and high quality clock system. The GigaSTAR™ clock system is derived from a high quality clock source. Each GigaSTAR™ link operates with a local clock system.

As a GigaSTAR™ link operates with its own clock system, a clock domain crossing may be necessary on the transmitter and receiver application.

Achieving extreme low latency requires limiting data buffers to a minimum. If data buffers are required to support a clock domain crossing, these have to be implemented in the application.

Providing background information about the parallel interface as well as examples of implementations of different interface applications are the subject of this application note.

Basic Interface Functionality

Signal Definition

Transmitter – INGT165B

System and Control Pins:

RESET# is an asynchronous active low input. Asserting reset forces the transmitter to a defined state.

PARGEN activates the internal parity generation. If PARGEN is driven high the external PARITY is ignored. The parity is calculated internally instead.

Data Pins:

PDATA[35..0] is the parallel transmit data, which is registered with the rising edge of RDCLK.

PARITY is the input pin for the externally provided parity signal.

FLAGI is an edge sensitive input, which is intended to mark a specific word in burst transmission. A rising edge places a flag on the data word currently in transmission. The receiver decodes the flag out of the serial bit-stream and toggles the level of the FLAGO output.

Interface Control Pins:

VALID is an asynchronous input signal. With the assertion of VALID the data transfer is initiated. After VALID has been asserted, the RDCLK output starts to toggle as long as VALID remains asserted. The possibility to assert VALID asynchronously allows synchronizing the data transfer start points to an external clock system. A periodic assertion / deassertion forces a continuous re-synchronization of the data transfer to an external clock system.

RDCLK parallel data is registered at the rising edge of RDCLK. RDCLK is derived from the 66 MHz reference clock. RDCLK is suitable to clock data out of a dual port memory device.

Status and Error Reporting Pins:

PERR is an active high output. If PERR has been asserted it indicates that the externally provided parity bit does not match the internally generated parity bit.

LOCK is an active high output. If LOCK is high the transmitter PLL is properly locked to the reference clock. If the PLL is unlocked no data is transmitted.

Receiver – INGR165B

System and Control Pins:

- RESET#** is an asynchronous active low input. Asserting reset forces the receiver to a defined state.
- EQSEL** activates the internal equalizer when set to high.

Data Pins:

- PDATA[35..0]** is the parallel receive data, which is provided with the rising edge of WRCLK.
- PARITY** is the received parity. If it mismatches the current data parity PERR# is asserted and the receiver starts the frame synchronization process (LSYNC# deassertion).
- FLAGO** is reset to low when RESET# is asserted. Each time a data word marked with a flag is received, the FLAGO signal is toggled.

Interface Control Pins:

- WRCLK** is the parallel data clock. When data are received the WRCLK is provided in parallel. The falling edge of WRCLK is centered with the data valid window.

Status and Error Reporting Pins:

- LOCK** is an active high output. If LOCK is high the receiver PLL is properly locked to the reference clock and the incoming serial data stream. If the PLL is unlocked no serial data is processed.
- LSYNC#** indicates whether the receiver has properly detected the frame structure of the incoming bit stream. In the case of a parity error or the reception of an invalid frame structure LSYNC# is deasserted.
- PERR#** is asserted if a mismatch of data and parity is detected.

Terms and Definition

- chip clock** is the internal clock of the transmitter's and receiver's logic block. This clock has a frequency of 330 MHz. It is derived from the serial line clock of 1320MHz generated by the PLL.
- data cycle** means the posting of a parallel 36 bit data word into the transmitter or the output of 36 bit of parallel data at the receiver.

Block Diagram

The block diagram below shows a simplified functional model of the transmitter and receiver parallel data interface and how they are linked together.

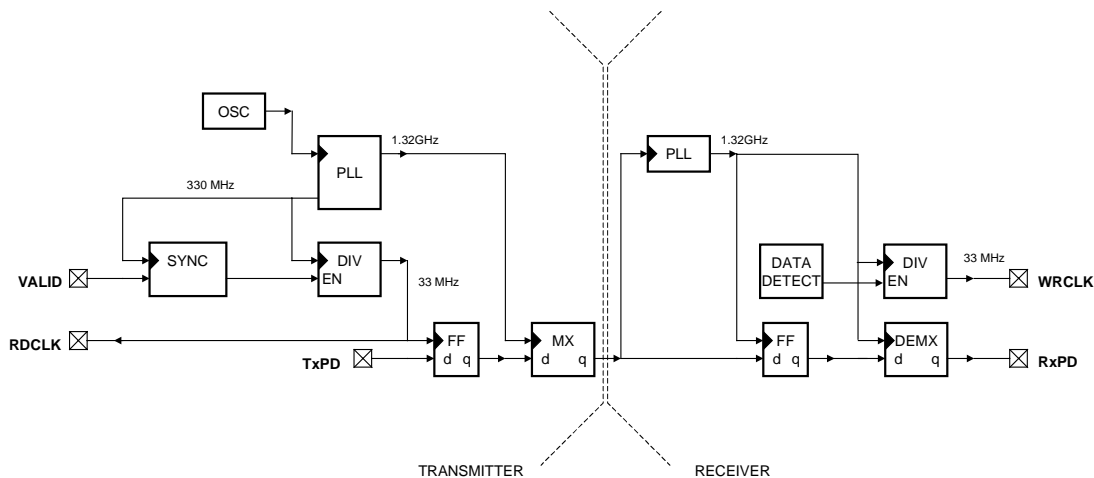


Figure 1: Transmitter and Receiver Parallel Interface

Transmitter – INGT165B

The GigaSTAR™ transmitter has a parallel data interface. The data transfer is initiated asynchronously with the assertion of VALID. The asynchronous VALID signal is sampled at the transmitter device with the chip clock (330 MHz).

As long as VALID remains asserted RDCLK pulses are generated.

VALID can also be de-asserted asynchronously. A data transfer which has been initiated always terminates correctly or, in other words, a full RDCLK pulse is always generated.

Due to the synchronization of the asynchronous VALID signal to the chip clock data transfers can be started every 3 ns. Therefore the time between VALID assertion to the rising edge of RDCLK has a uncertainty of 3 ns.

Receiver – INGR165B

The receiver PLL is locked on the incoming serial bit stream. The serial bit stream is generated with the transmitter RF clock, which is derived from the reference clock. Consequently, the receiver PLL is locked on the transmitter reference clock.

The incoming serial data stream is re-timed to the receivers clock system. The serial data stream may contain any sequence of idle pattern and data frames. Data is extracted and provided in a parallel format together with a WRCLK pulse. Each time new data is presented WRCLK goes high for 5 chip clock cycles.

Transmission Modes

Burst

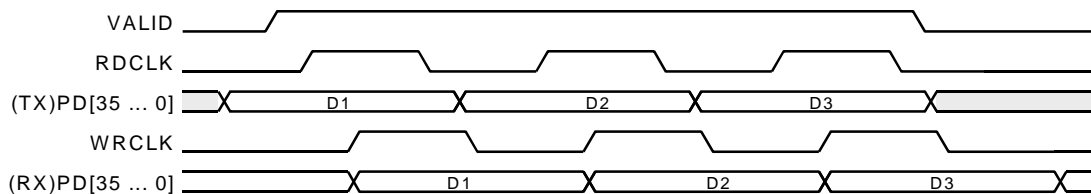


Figure 2: Functional Burst Timing

The burst timing enables a continuous data stream to be transmitted with the maximum data rate. Data has to be provided synchronously to the RDCLK at the transmitter. At the receiver data has to be accepted synchronously to WRCLK.

Single Word

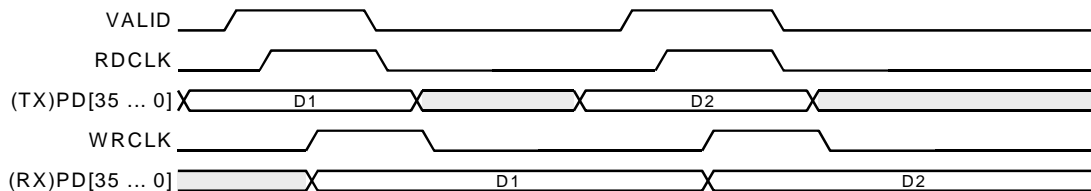


Figure 3: Functional Single Word Timing

The single word timing is intended to support lower data rates and it provides a mechanism to synchronize the transmit-data stream to an external clock system (see plesynchronous transmission).

The basic principle is that if data is available, VALID has to be asserted. If no further data is available VALID has to be deasserted, for example with the rising edge of RDCLK. The transmit-data becomes registered with the rising edge of RDCLK.

At the receiver data is provided with a WRCLK pulse.

Sideband Signaling

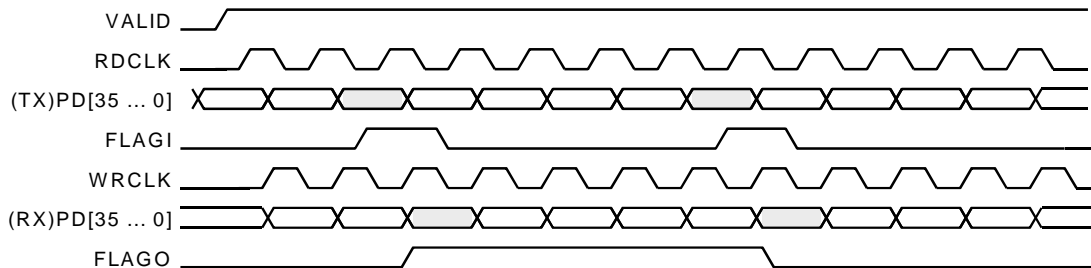


Figure 4: Functional FlagI / FlagO Timing

If single data words within a burst transmission should be marked without interrupting the burst, this data can be flagged. A rising edge at FLAGI flags the actual data. At the receiver the FLAGO is toggled each time a flagged data is received. Note that by the insertion of a flag the RDCLK and WRCLK cycle times are prolonged by 3 ns.

Plesynchronous Transmission

A specific variant of the single word timing is to drive the VALID signal directly with a data clock. The frequency of the data clock can be close to the maximum interface clock frequency of 33 MHz but must never exceed the maximum interface clock (Reference Clock / 2). The result is a plesynchronous interfacing to the application. This mode of operation is of special interest for display data distribution applications.

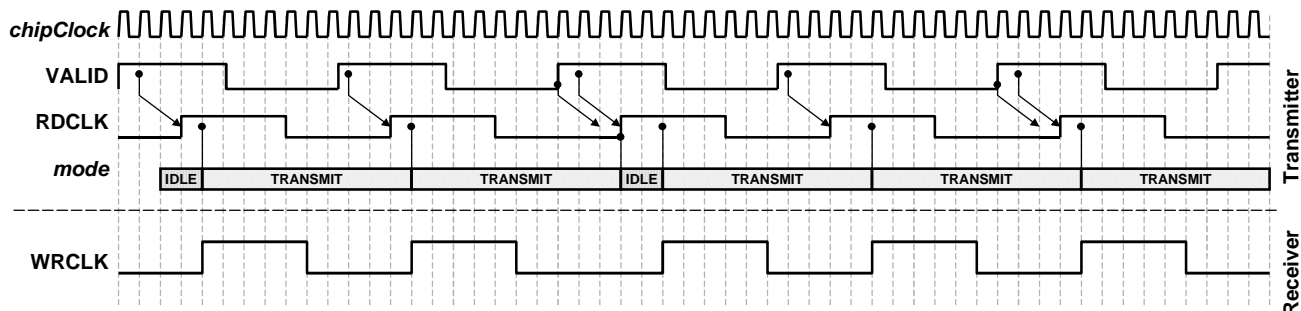


Figure 5: Plesynchronous Timing

At the GigaSTAR™ transmitter the external data clock is connected to the VALID input. The asynchronous VALID signal is sampled and synchronized with the chip (internal) clock with a frequency of 330 MHz. When VALID is high a RDCLK pulse is generated. The RDCLK is high for 5 chip clocks and low for 5 chip clocks. At the end of an RDCLK pulse the VALID signal is checked. If high, a further RDCLK pulse is initiated else RDCLK stays at a low level until VALID goes to high.

Each time RDCLK has a rising edge parallel data is registered and the serialization process starts. The complete serialization of one parallel data word takes 10 chip clocks. After serialization is finished the RDCLK is checked again for the availability of new data. If RDCLK is high the serialization continues with the new data otherwise an idle pattern of 6 ns is transmitted. After transmission of the idle pattern RDCLK is checked again for new data.

At the GigaSTAR™ receiver, each time a data word has been completely converted from serial to parallel format a WRCLK pulse is generated. WRCLK remains high for 5 chip clock periods.

A result of the process described above is that the external clock controls, with its rising edge, the initiation of a data transfer. The typical characteristic of such a data transfer is a sequence of short bursts interrupted by an idle pattern (stepping). Each rising edge of the external clock realigns the starting point of the data transfer to the external clock system. This realignment is performed with the accuracy of a chip clock period of 3 ns. As a consequence, the RDCLK low phase varies in 3 ns steps. The conditional insertion of idle pattern causes the WRCLK low period to vary by 6 ns.

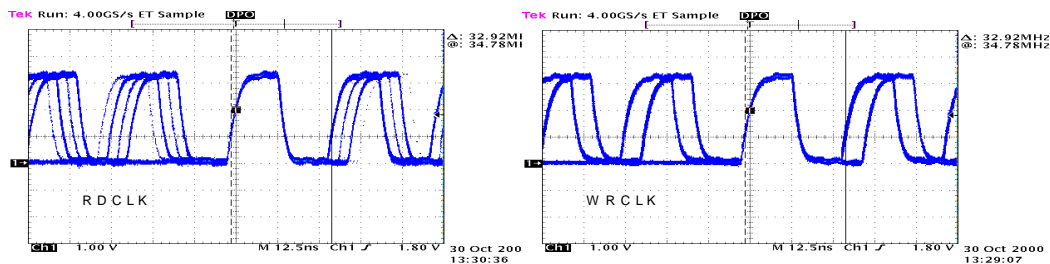


Figure 6: RDCLK and WRCLK Timing at Ple-Synchronous Operation

Note: All timings in this section are functional timings. Synchronization and propagation delays are not included. For specific timing values, please refer to the data sheet.

Interface Application Examples

FIFO Interface

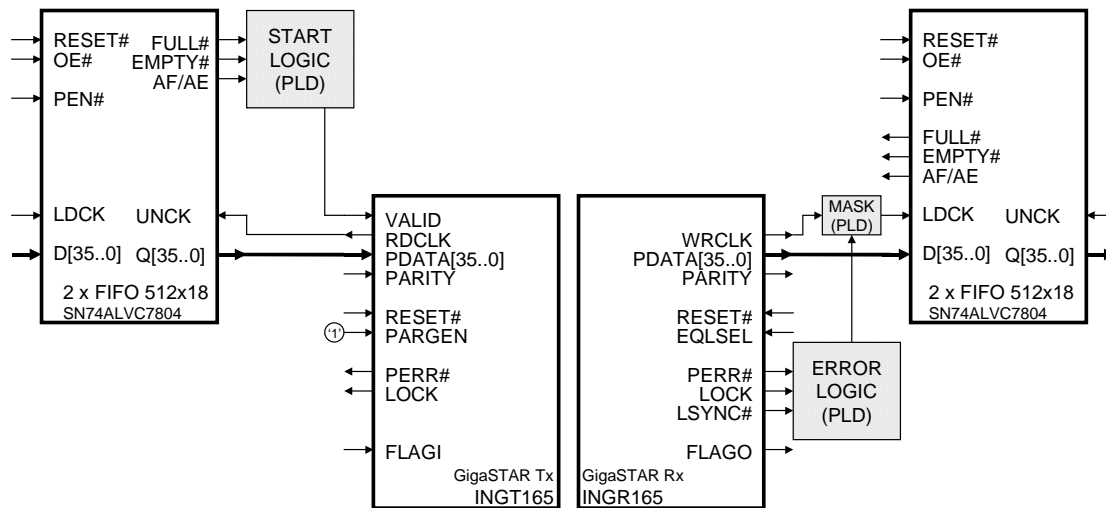


Figure 7: Fifo Interface

The GigaSTAR™ can directly interface to many asynchronous FIFOs.

The GigaSTAR™ transmitter's RDCLK is connected to the unload clock UNCK of the FIFO. According to the filling status flags like FULL, EMPTY or AF/AE the GigaSTAR™ VALID signal is generated. In many cases one of the filling flags can be directly connected to the VALID signal.

Transmit data is posted in the FIFO. The filling-flags status causes the assertion of VALID. A burst transmission is initiated. The RDCLK clocks data out of the FIFO until the FIFO empties. The burst transmission stops when VALID becomes deasserted due to an empty status of the FIFO.

The GigaSTAR™ receiver's WRCLK very often can be directly connected to the load clock of an asynchronous FIFO. In some cases it might be useful to mask the WRCLK if an error condition is signaled by the receiver, just to avoid that erroneous data is stored in the FIFO.

Video Interface

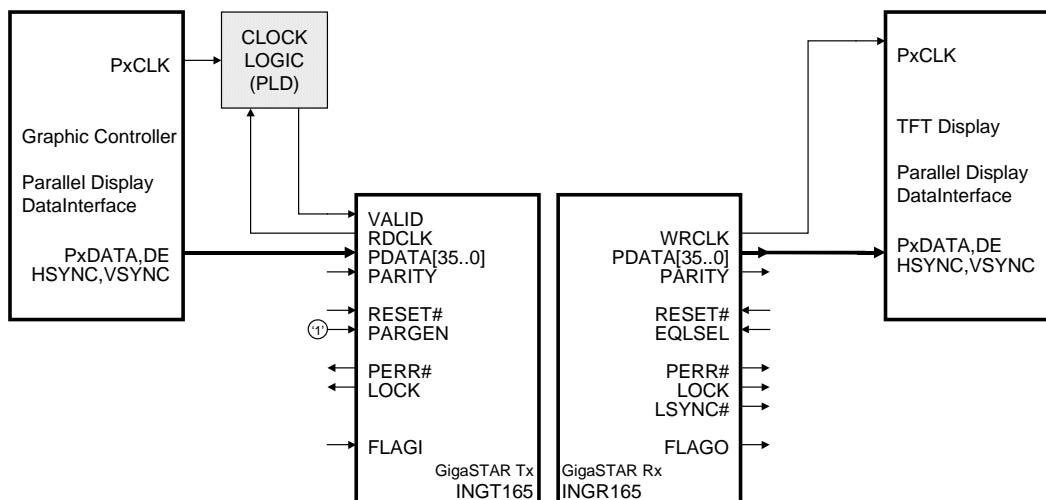


Figure 8: Video Interface

Parallel display data interfaces for flat panel displays provide the pixel and control data together with a pixel clock. A variety of pixel clock speeds and pixel data interface sizes exist.

At the receiver the pixel clock, together with pixel data and control signals, has to be provided in the same manner as it was provided at the transmitter interface.

In this application the link operates in a plesynchronous mode, as outlined on page 8.

The pixel clock frequency must not exceed the maximum RDCLK frequency of 33 MHz, otherwise transmit data will be lost. The pixel clock frequency of an 2 pixel/clock XGA interface with 6 bit color resolution and 60Hz refresh rate is 32.5 MHz if VESA compliant.

For lower pixel clock frequencies (24 MHz for example) the pixel clock cannot be connected directly to the VALID input. The high phase timing would cause a pixel data to be transmitted twice or more. To transmit a pixel data per clock edge external logic is required generating an appropriate VALID signal. The principle of such a logic would be setting VALID high with a rising edge on the pixel clock and resetting VALID to low with a rising edge on RDCLK.

Repeater for Display Data Distribution

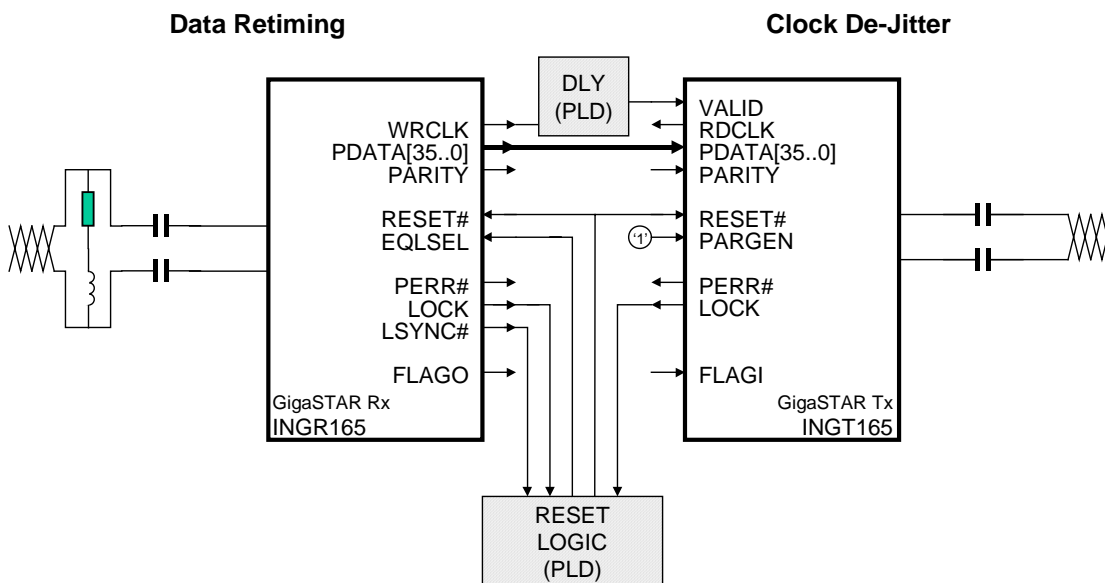


Figure 9: Repeater for Display Data Distribution

The concept of a GigaSTAR™ repeater for digital display data distribution is to connect a GigaSTAR™ receiver and a GigaSTAR™ transmitter back-to-back. The WRCLK of the receiver drives the VALID input of the transmitter. Only data is repeated. Sideband signaling is not repeated.

The GigaSTAR™ receiver (B) located on the repeater is connected to a GigaSTAR™ transmitter (A). GigaSTAR™ transmitter (A) and GigaSTAR™ receiver (B) implement the first GigaSTAR™ link (L1). The GigaSTAR™ transmitter (C) located on the repeater is connected to a receiver (D). GigaSTAR™ transmitter (C) and GigaSTAR™ receiver (D) implement the second link (L2).

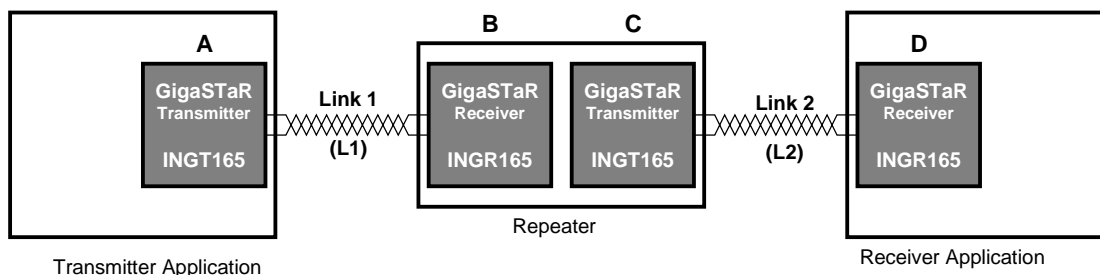


Figure 10: Repeater System Setup

A clock domain crossing is performed within the repeater, which completely decouples the two links L1 and L2.

Due to the plesynchronous operation mode each GigaSTAR™ link, consisting of a transmitter and a receiver, operates with its own jitter-free clock system. The clock system of each GigaSTAR™ link is fully decoupled from each other. A jitter accumulation over multiple links can not occur. This allows for multiple repeater cascading.

As outlined on page 8 the plesynchronous mode requires the maximum frequency at VALID to be below half of the reference clock (66 MHz). As each link operates with its own local reference clock oscillator, the maximum interface frequency at each link may vary depending on the accuracy of the crystal oscillators. With a maximum interface frequency of 32.5 MHz, enough margins are provided to compensate for differences at the reference clock oscillators.

The following timing shows the parallel interface timing of two cascaded links. The application's data clock is connected to the VALID input. The data clock (VALID) frequency at the first transmitter is assumed to be slightly below the maximum frequency.

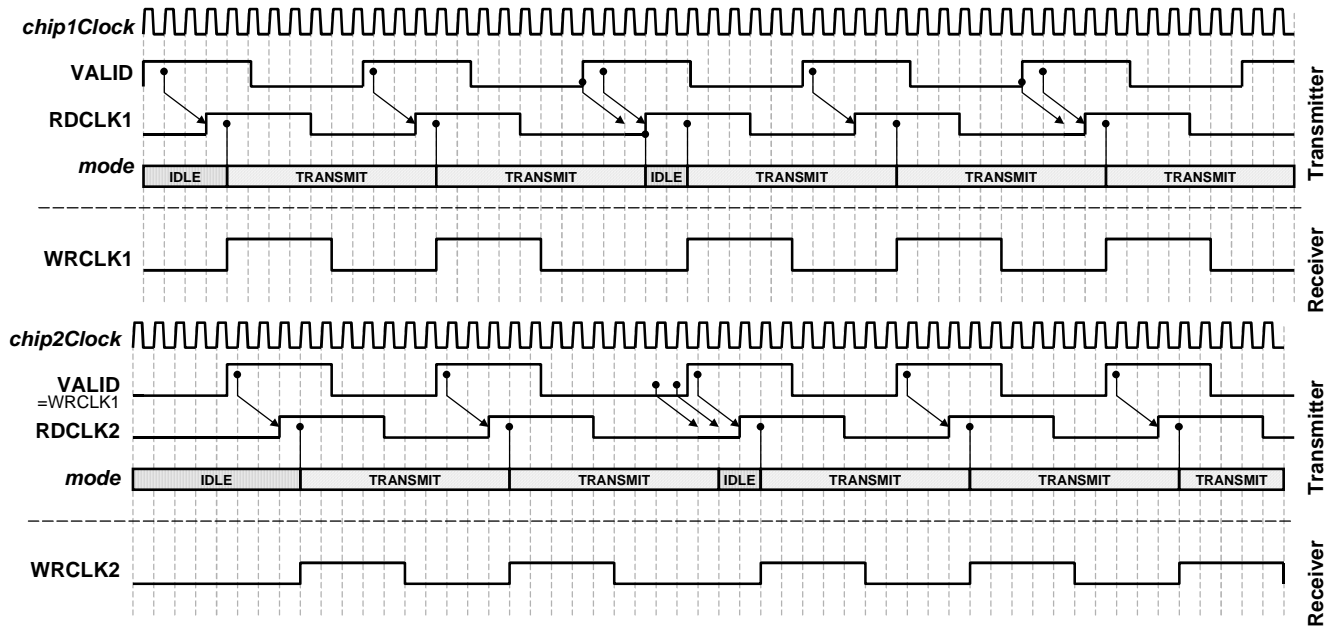


Figure 11: Cascaded Link Timing

At the top the plesynchronous timing as outlined on page 8 can be seen. The data clock, which is connected to VALID at the first transmitter, is recovered at the receiver and provided at the WRCLK output. The average WRCLK frequency equals the data clock frequency. Inserting an idle pattern after a burst sequence performs the adaptation.

The WRCLK output is connected to the VALID input of the following transmitter and parallel data is passed through.

It can be seen that the WRCLK (data clock) at the different receiver stages are identical except a transport delay.

Scaling GigaSTAR™ Channels

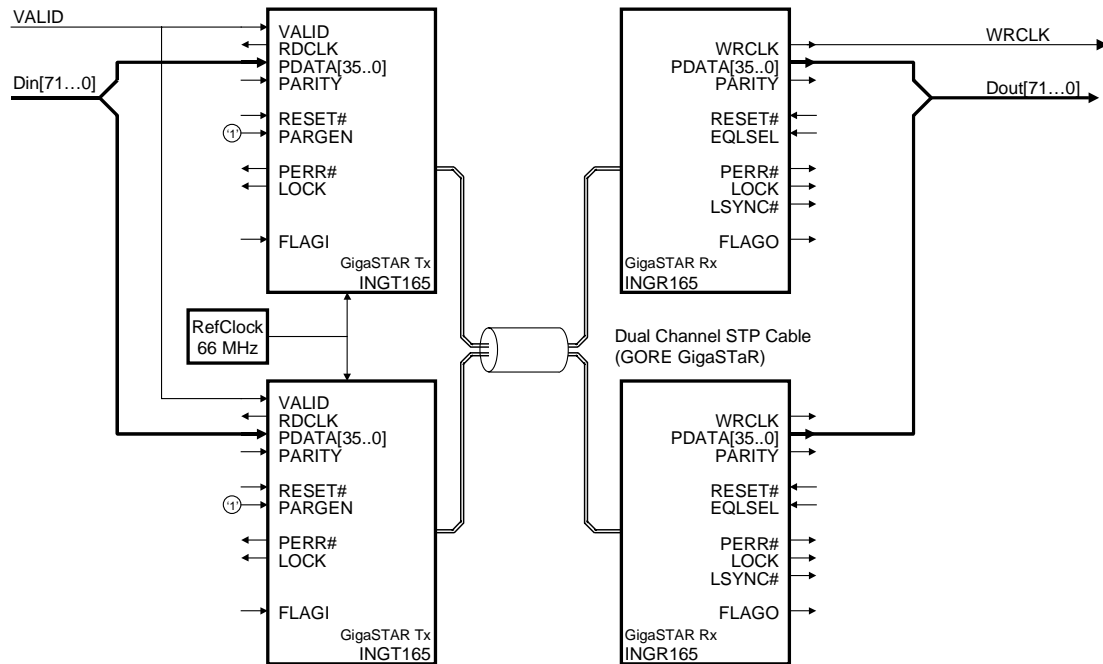


Figure 12: Dual GigaSTAR™ Link Set-up

The fundamental harmonic of the serial data is 660 MHz. This is still very convenient to handle, as the impedance of transmission lines on PCBs based on FR4 substrate does not vary too much. Moreover, a wide range of connectors and cable assemblies are available for this frequency range.

If the application requires more bandwidth, it might be wise not to double the transmission frequency but to increase the bandwidth by scaling several GigaSTAR™ channels.

As the clock system of GigaSTAR™ is based on the reference clock, GigaSTAR™ transmitters can easily be synchronized just by operating with the same reference clock. The transmitter's chip clocks are locked to the reference clock. Providing all GigaSTAR™ transmitters with the same signal at VALID will initiate the data transfer synchronously. Due to the VALID sampling period of 3 ns, the RDCLKs of all transmitters and therefore also the data sampling point can have a skew of 3 ns maximum. Most interface timings have enough margins to allow for this.

As the chip clocks of all transmitters are synchronous the serial data is also synchronous, except a maximum skew of 3 ns as mentioned. The cabling might add further skew but with typical propagation delays of approximately 4 ns/m the cabling skew between the single channels can be kept easily below 1 ns.

The GigaSTAR™ receivers are locked to the serial data. At burst or single word transfers parallel data output and WRCLK have a maximum skew of 3ns plus the cabling skew. As a result the WRCLK of any receiver can be used to sample the data of all receivers.

At plesynchronous data transfers with a data rate close to the maximum the insertion of idle pattern for data rate adaptation is a function of the sampling time of VALID. The idle pattern insertion might vary from link to link. This adds further 6 ns skew between the WRCLKs at the receivers. In total a maximum skew of 9 ns has to be compensated.

Inova Semiconductors GmbH

Ismaninger Str. 3

D-81675 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>



inova[®] Semiconductors is a registered trademark of Inova Holding GmbH.



GigaSTAR[®] is a registered trademark of Inova Semiconductors GmbH.

All other trademarks or registered trademarks are the property of their respective holders.

Inova Semiconductors GmbH does not assume any liability arising from the application or use of the product described herein; nor does it convey any license under its patents, copyright or any rights of others.

Inova Semiconductors products are not designed, intended or authorized for use as components in systems to support or sustain life, or for any other application in which the failure of the product could create a situation where personal injury or death may occur.

The information contained in this document is believed to be current and accurate as of the publication date. Inova Semiconductors GmbH reserves the right to make changes at any time in order to improve reliability, function or performance to supply the best product possible.

Inova Semiconductors GmbH assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction if such be made.

© Inova Semiconductors GmbH 2000. All rights reserved.