Virtex-5 RocketIO GTP Transceiver

Objectives

- **Understand the Xilinx Virtex-5 RocketIO GTP transceiver**
- **Provide an overview of the GTP design and verification tools**
- **Provide a demo to better understand the operation of the Virtex-5 RocketIO GTP transceiver**

Agenda

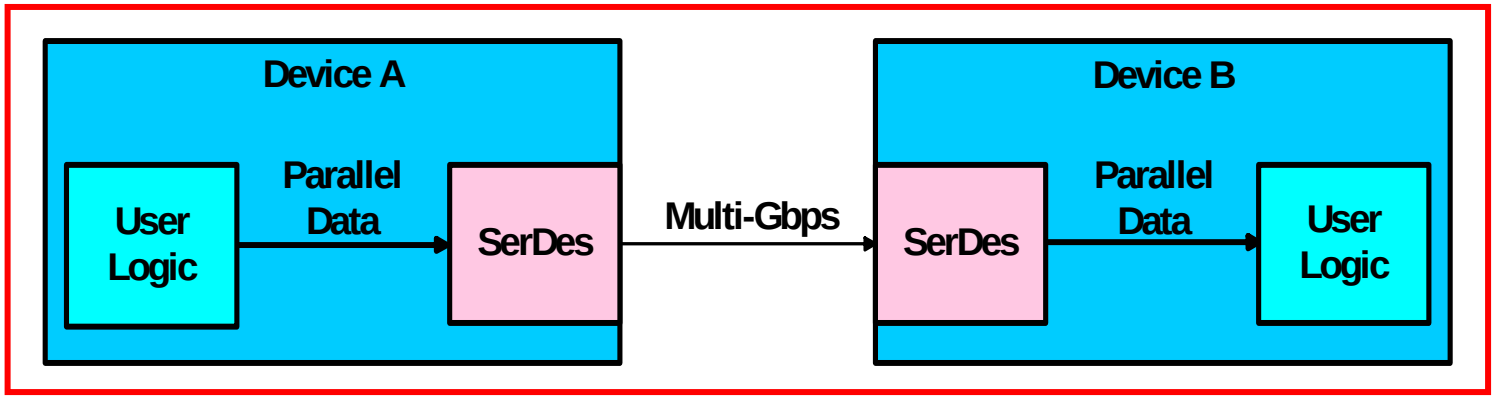
- **Introduction to Virtex-5 GTP Transceiver**
- **Virtex-5 GTP Transceiver Shared Resources**
- **GTP Transmitter**
- **GTP Receiver**
- **GTP Loopback and CRC**
- **GTP Design and Verification Tools**
- **GTP Demo**

Agenda

- **Introduction to Virtex-5 GTP Transceiver**
- Virtex-5 GTP Transceiver Shared Resources
- GTP Transmitter
- GTP Receiver
- GTP Loopback and CRC
- GTP Design and Verification Tools
- GTP Demo

Serial Signaling

- **Serial signaling is the preferred choice in all markets such as telecom, datacom, computing, and storage**
  - Supports very high multi-gigabit data rates
  - Avoids clock/data skew by using embedded clock
  - Reduces EMI & power consumption
  - It is the only IO technology that meets today's high-speed requirements
  - It is faster than parallel signaling
    - Parallel I/O data rates are inherently limited due to unavoidable skew between clock lines and multiple data lines
    - High pin and via counts in parallel signaling create complex routing and skew challenges



- **The Virtex-5 RocketIO GTP transceiver is a power-efficient transceiver, highly configurable and tightly integrated with the programmable logic resources of the FPGA**
  - Available in Virtex-5 LXT and SXT families
  - Current Mode Logic (CML) serial drivers/buffers with configurable termination, voltage swing, and coupling
  - Programmable TX pre-emphasis and RX equalization for optimized signal integrity
  - Line rates from 500Mbps to 3.2Gbps, with optional 5x digital over-sampling for rates between 100Mbps and 500Mbps
  - Optional built-in PCS features, such as 8B/10B encoding, comma alignment, channel bonding, and clock correction
  - Out of band signaling, including COM signal support for PCI Express and SATA
  - PRBS generation and checker

Virtex-5 LXT/SXT Device and Package Combinations

- **Depending on the device, a Virtex-5 LXT/SXT FPGA has between 8 and 24 transceiver modules**
- **V5 LXT and SXT families offer device migration options both within and between families**

Virtex-5 LXT Device Family

	Package		
	FF665	FF1136	FF1738
Device	GTPs	GTPs	GTPs
XC5VLX30T	8		
XC5VLX50T	8	12	
XC5VLX85T		12	
XC5VLX110T		16	16
XC5VLX220T			16
XC5VLX330T			24

Virtex-5 SXT Device Family

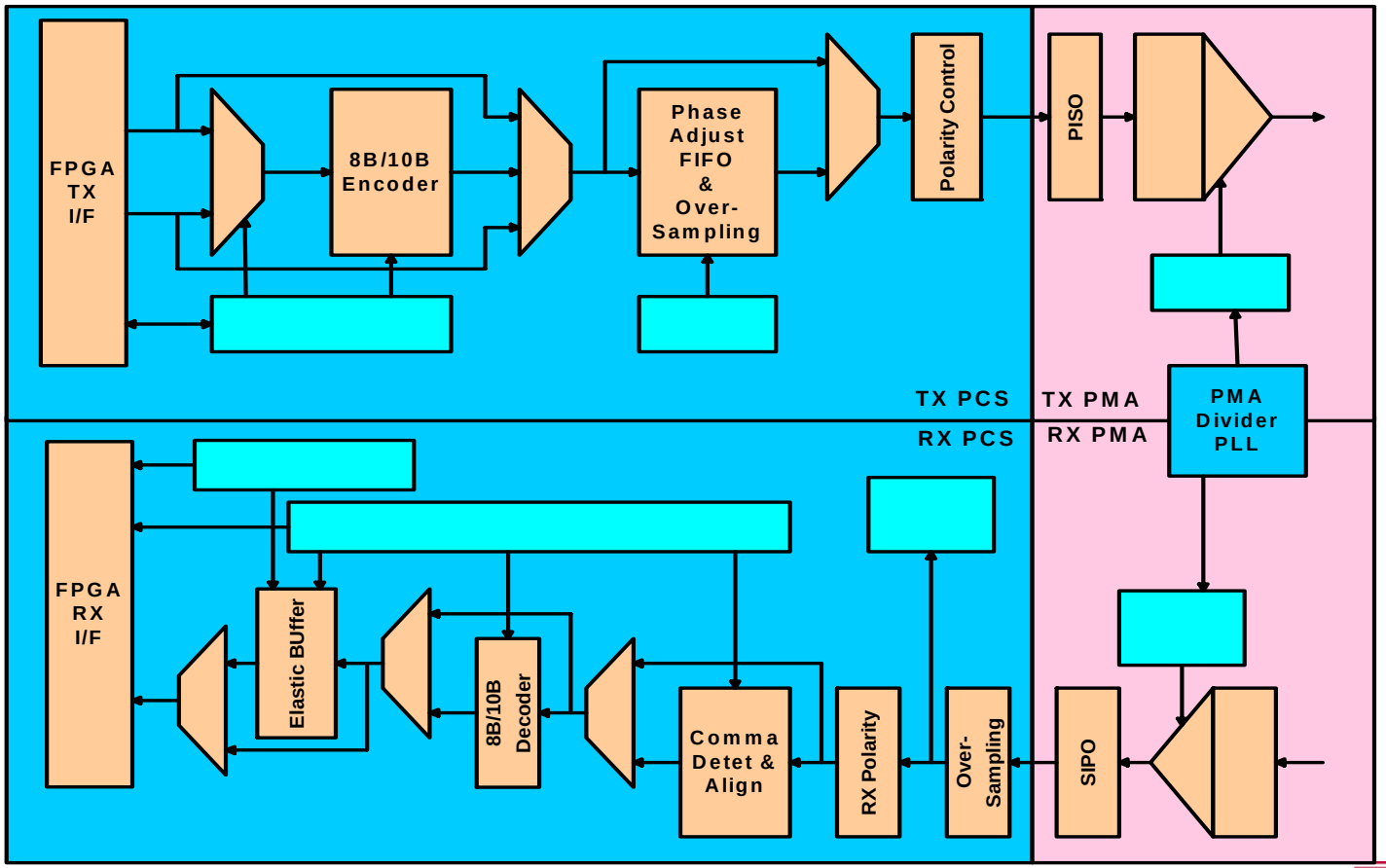
	Package		
	FF665	FF1136	FF1738
Device	GTPs	GTPs	GTPs
XC5VSX35T	8		
XC5VSX50T	8	12	
XC5VSX95T		16	24

Communications Standards Supported by the GTP

Protocols Supported	Protocol Data Rates Supported
PCI Express Rev 1.0a PCI Express Rev 1.1	2.5 Gb/s
XAUI 802.3ae D5p0	3.125 Gb/s
OC-12/48	622.08/2488.32 Mb/s
FC-1 Rev 4.0	1.0625 Gb/s
FC-2 Rev 4.0	2.125 Gb/s
10GFC	3.1875 Gb/s
SDI HD-SDI DVB-ASI	143/176/270/360 Mb/s 1.485/1.4835 Gb/s 270 Mb/s
Aurora	100 Mb/s – 3.2 Gb/s
Gigabit Ethernet (1000BASE-CX 802.3z/D5.0)	1.25 Gb/s
SATA Gen 1/II (Rev 1.0a) SATA Gen. 2 (Rev 1.0a)	1.5 Gb/s 3.0 Gb/s
Serial RapidIO	1.25/2.5/3.125 Gb/s
Infiniband (Volume 2 Release 1.1)	2.5 Gb/s
10G Base-CX4 802.3ak/D4.0	3.125 Gb/s

Virtex-5 GTP High-level Block Diagram

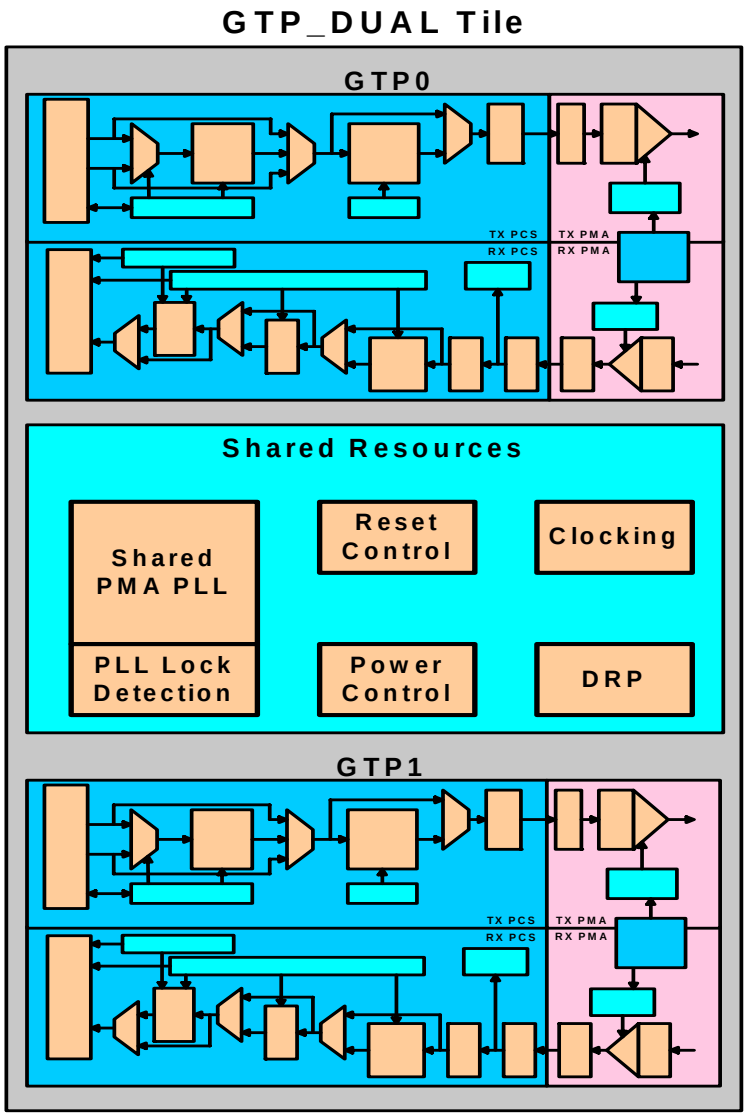
- **GTP transceiver consists of the transmit PCS/PMA and receive PCS/PMA blocks**
  - The transmit and receive blocks share a PMA divider PLL
  - The TX and RX internal data paths are 8 or 10 bits wide



Virtex-5 GTP Tile

- **GTP transceivers are placed as dual transceiver GTP_DUAL tiles in Virtex-5 LXT/SXT platform devices**

- **To minimize power consumption and area, many important GTP functions are shared between two transceivers**
  - These functions include the generation of a high-speed serial clock, resets, power control, and dynamic reconfiguration
  - This configuration allows the TX and RX functions of both transceivers to share a single PLL, reducing size and power consumption



Virtex-5 GTP Configuration Overview

- **There are two ways to configure a Virtex-5 GTP**
  - *Static configuration* - The transceiver is configured using a combination of GTP port tie-offs and attribute settings at design time to support a specific protocol
  - *Dynamic configuration* - The transceiver is configured by driving the GTP ports and using the Dynamic Reconfiguration Port (DRP) to modify the run-time configuration of the GTP

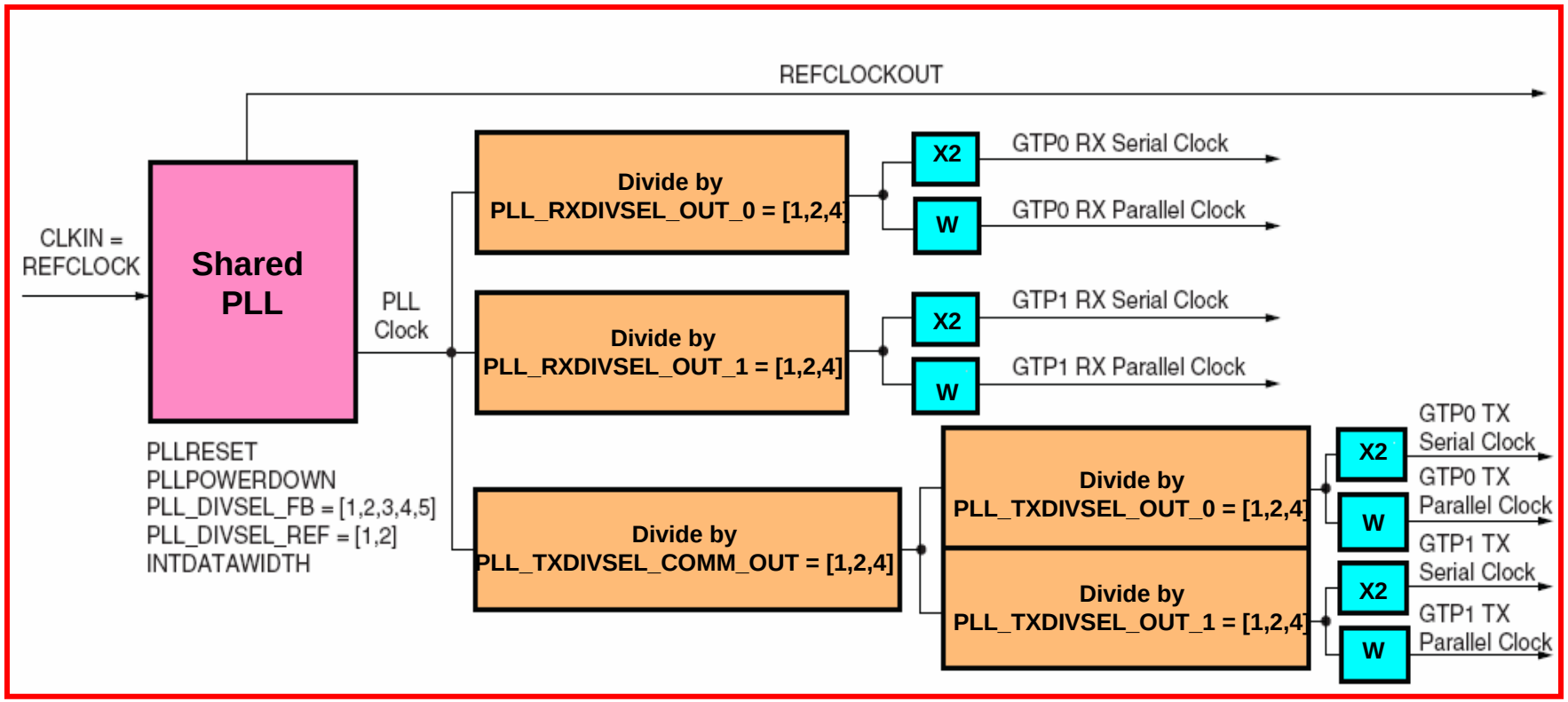
- **GTP configuration can be complex because of the large number of possible settings**
  - Xilinx provides a RocketIO wizard to help manage the configuration process of the Virtex-5 GTP
  - The wizard is highly recommended for any design that utilizes the Virtex-5 RocketIO GTP

Agenda

- Introduction to Virtex-5 GTP Transceiver
- **Virtex-5 GTP Transceiver Shared Resources**
- GTP Transmitter
- GTP Receiver
- GTP Loopback and CRC
- GTP Design and Verification Tools
- GTP Demo

GTP Shared PMA PLL

- **Each GTP_DUAL tile includes one shared PMA PLL used to generate a high speed serial clock from a high-quality reference clock input (CLKIN)**
  - The high-speed clock from this block drives the TX and RX PMA blocks for both GTP transceivers in the tile



GTP Shared PLL Conceptual View

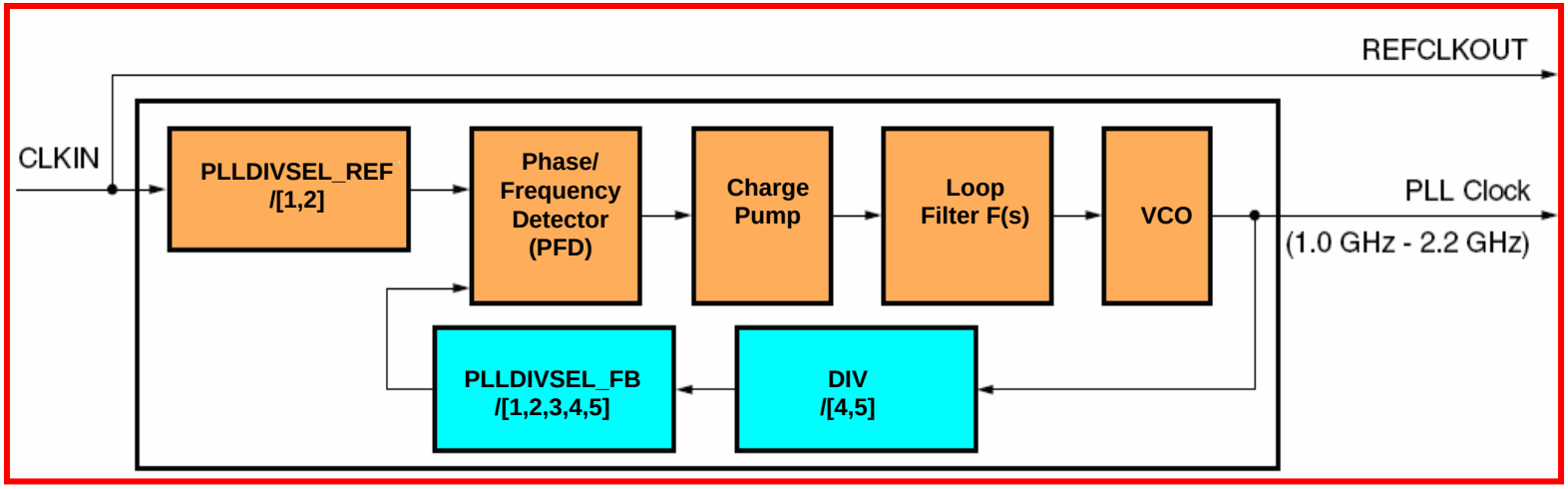
- **The first step in using the GTP_DUAL tile is to set the output of the shared PLL (PLL clock) to a rate that can be used by each GTP transceiver to generate an appropriate serial line rate and corresponding parallel clock rate**

  PLL Clock = CLKIN x PLLDIVSEL_FB x DIV / PLLDIVSEL_REF

  Where: DIV = 4 for 8-bit and 5 for 10-bit internal data path

  – Both GTP TX and RX blocks are equipped with an independent divider that can divide the PLL clock by a factor of 1, 2, or 4

  – The PLL clock rate must be set to one-half the required line rate before the independent dividers



Shared PLL Configuration Example - Gigabit Ethernet

Inputs (assumptions):

Line Rate = 1.25Gbps for TX and RX
Internal Data Path = 10 bits (Gigabit Ethernet uses 8B/10B encoding)
DIV = 5 (internal data path is 10, otherwise DIV will be set to 4)
Reference Clock = 125MHz (user selectable)

Outputs (calculated values):

1) Calculate the required PLL clock rate

- **Considering the RX side, because the SIPO block uses both edges of the clock to deserialize data, it must be fed a clock running at 1.25/2 = 0.625 GHz**
- **Because this RX rate of 0.625 GHz is below the PLL range (1.0 GHz – 2.2 GHz), the external divider (PLL_RXDIVSEL_OUT) must be 2 to allow the PLL to run twice as fast (1.25 GHz)**
- **The PLL clock rate is thus PLL_CLOCK = 0.625 x 2 = 1.25 GHz**

2) Calculate the required PLL divider ratio

- **Using the values CLKIN, DIV, and PLL_CLOCK determined above**

$$\frac{\text{PLLDIVSEL\_FB}}{\text{PLLDIVSEL\_REF}} = \frac{\text{PLL\_CLOCK}}{\text{CLKIN x DIV}} = \frac{1.25\text{GHz}}{125\text{MHz x 5}} = 2$$

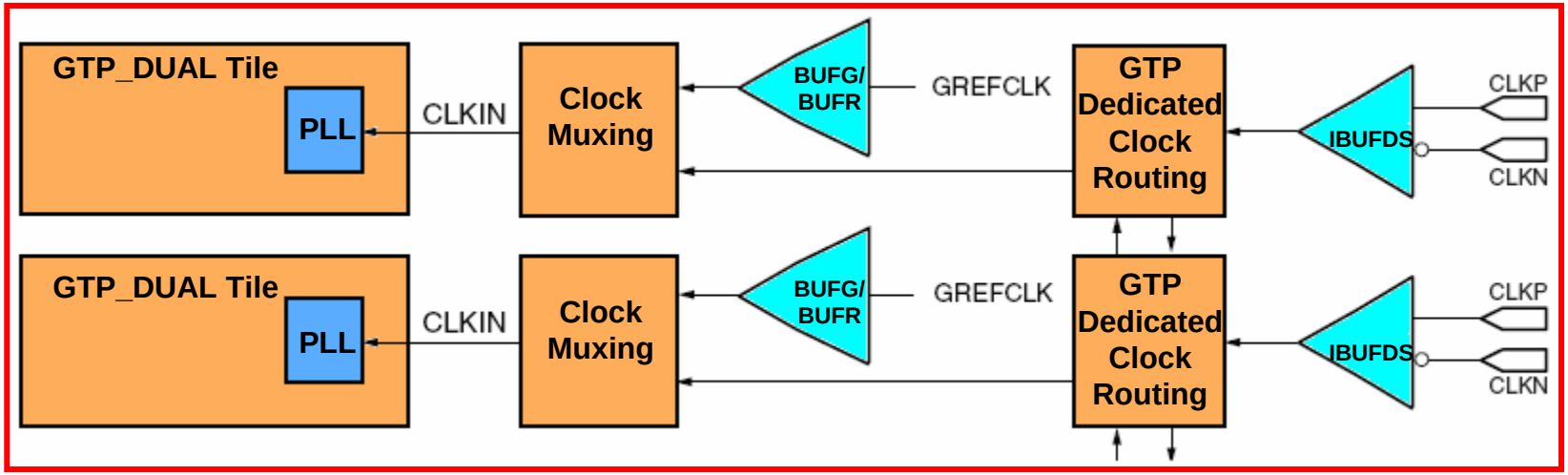
3) Select the PLL divider values

Select the smallest divider values that result in the required PLL divider ratio. In this case, using PLLCLKDIV_FB = 2 and PLLCLKDIV_PLL = 1 results in a ratio of two.

GTP Clocking

- **The GTP_DUAL reference clock is provided through the CLKIN port. There are three ways to drive the CLKIN port**
  - Using an external oscillator to drive GTP dedicated clock routing (CLKP and CLKN clock inputs)
  - Using a clock from a neighboring GTP_DUAL tile through GTP dedicated clock routing
  - Using a clock from inside the FPGA (GREFCLK)

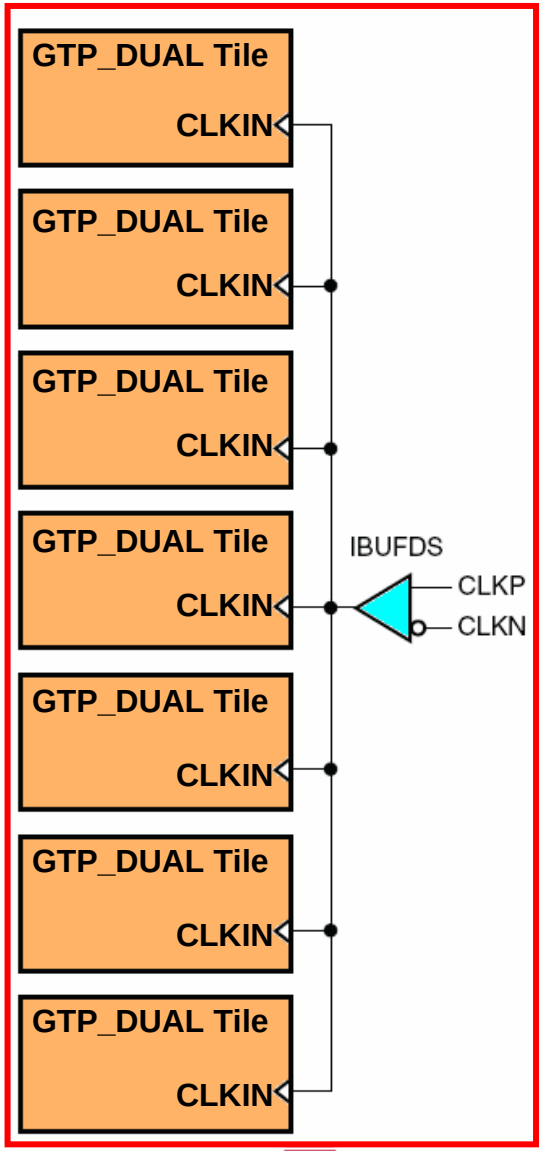
- **Using the dedicated clock routing provides the best possible clock to the GTP_DUAL tiles**



Multiple GTP_DUAL Tiles with Shared Reference Clock

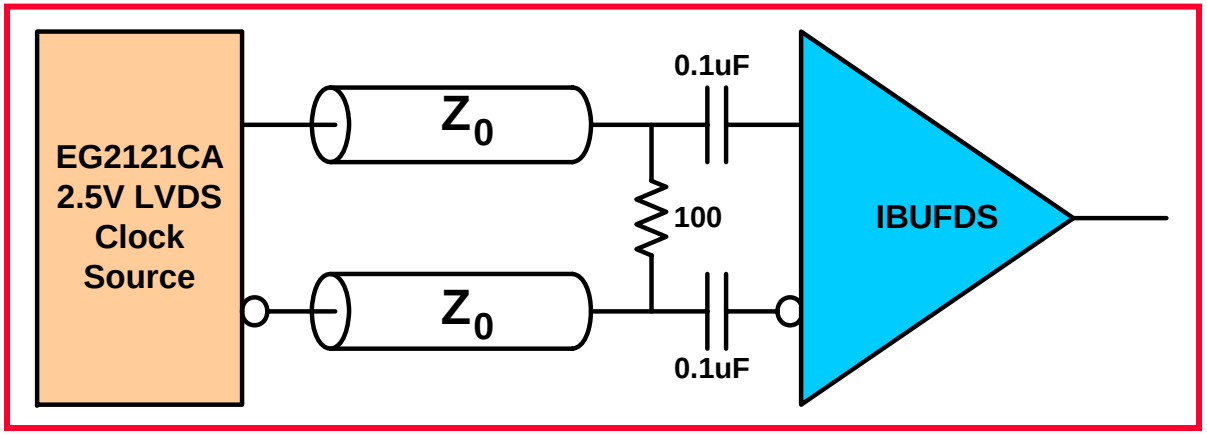
- **The external clock from one GTP_DUAL tile can be used to drive the CLKIN ports of neighboring tiles using the GTP Dedicated Clock Routing block**

- **To keep the jitter of this configuration within the required jitter margin for high-speed designs, the following rules must be observed**
  - The number of GTP_DUAL tiles *above* the sourcing GTP_DUAL tile must *not* exceed three
  - The number of GTP_DUAL tiles *below* the sourcing GTP_DUAL tile must *not* exceed three
  - The total number of GTP_DUAL tiles sourced by the external clock pin pair (CLKN/CLKP) must *not* exceed seven



Reference Clock Oscillator

- **A high-quality crystal oscillator is essential for good performance**
  - When using one of the recommended oscillators, the manufacturer's power supply design guide must be followed
  - Virtex-5 device characterization is based on the same recommended oscillators
- **When considering alternate clock sources, the alternate oscillators must meet or exceed the specifications of the recommended oscillators**
- **Depending on the application and its performance goals, it is possible to stray from the clock source specifications, but in that case the specified performance of the GTP is not guaranteed**



Review Questions

- **How many different ways can a user drive the CLKIN port of a GTP_DUAL tile?**
  - Using an external oscillator to drive GTP dedicated clock routing (CLKP and CLKN clock inputs)
  - Using a clock from a neighboring GTP_DUAL tile through GTP dedicated clock routing
  - Using a clock from inside the FPGA (GREFCLK)

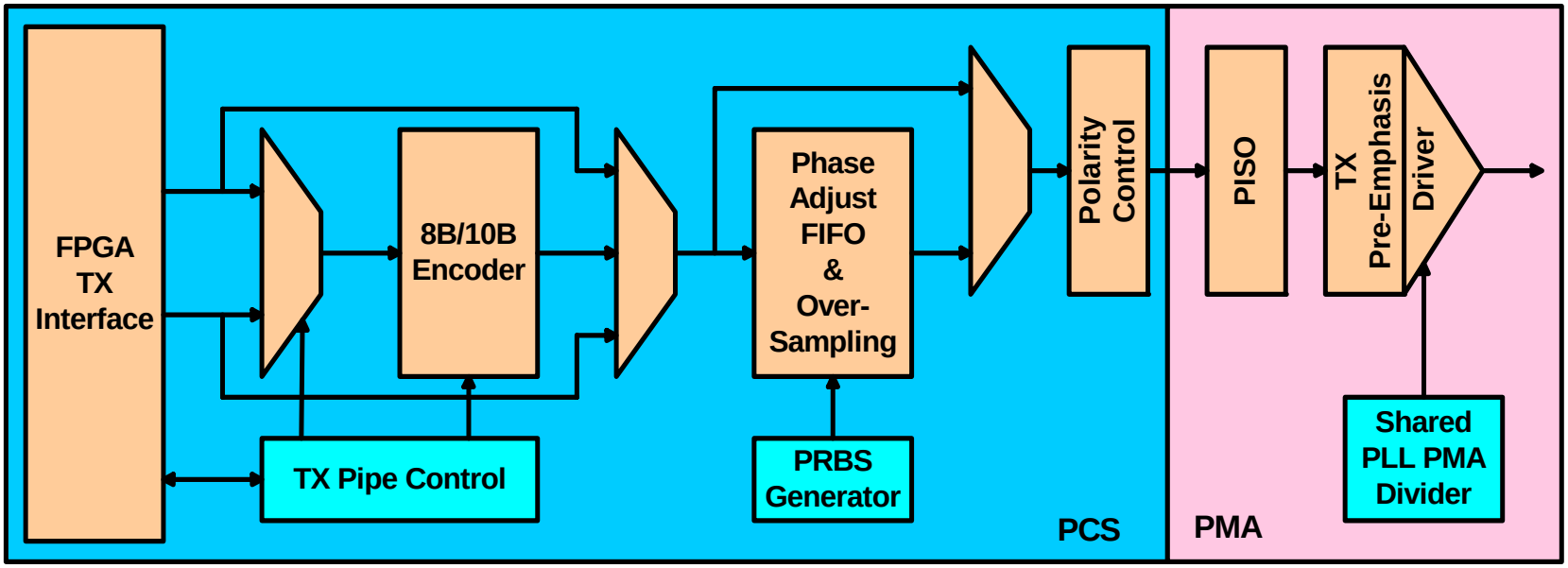
- **How many GTP_DUAL tiles can a differential clock input connected to the CLKP/CLKN pins drive?**
  - Seven, three tiles above and three tiles below

Agenda

- Introduction to Virtex-5 GTP Transceiver
- Virtex-5 GTP Transceiver Shared Resources
- **GTP Transmitter**
- GTP Receiver
- GTP Loopback and CRC
- GTP Design and Verification Tools
- GTP Demo

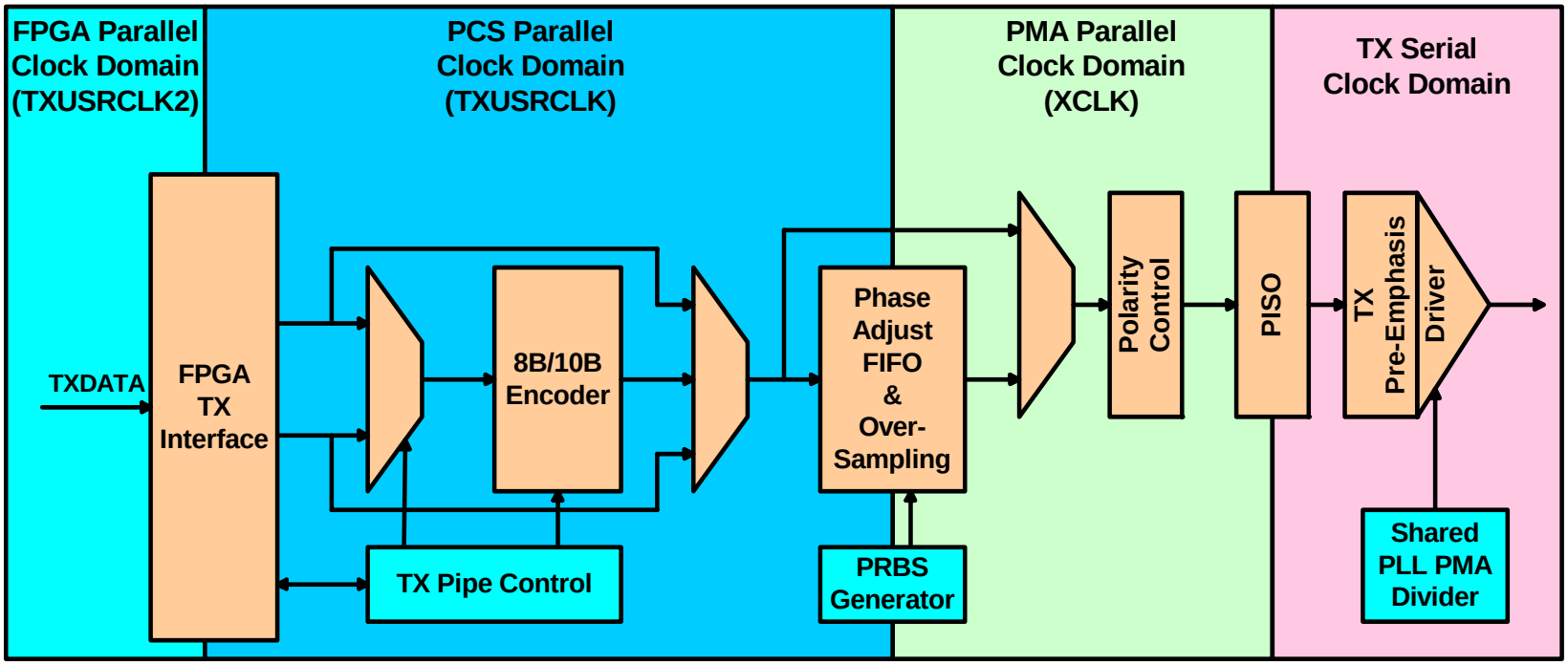
Virtex-5 GTP Transmitter

- **Each GTP transceiver in the GTP_DUAL tile includes an independent transmitter, made up of a PCS and a PMA**
  - Parallel data flows from the FPGA into the FPGA TX interface, through the PCS and PMA, and then out the TX driver as high-speed serial data
  - The GTP transceiver includes a TX polarity control function to invert outgoing data from the PCS before serialization and transmission via the TXPOLARITY port



GTP TX Clock Domains

- **The GTP TX data path has two internal parallel clock domains, the PMA parallel XCLK and the PCS parallel TXUSRCLK clock domains**
  - To transmit data, the XCLK rate must match the TXUSRCLK rate, and all phase differences between the two domains must be resolved
  - The GTP transmitter includes a TX buffer and a TX phase-alignment circuit to resolve phase differences between the XCLK and TXUSRCLK clock domains



FPGA TX Interface

- **The FPGA TX interface allows parallel data to be written to the GTP transceiver for transmission as serial data. To use the interface**
  - The width of the data interface (TXDATA) must be configured
  - TXUSRCLK2 and TXUSRCLK must be connected to clocks running at the correct rate
  - The FPGA TX Interface data path resides in the TXUSRCLK2 clock domain
  - The TX internal data path resides in the TXUSRCLK clock domain
  - TX internal data path is always 1-byte wide while the TX FPGA interface can be 1 or 2-bytes wide

INTDATAWIDTH	TXDATAWIDTH	TXENC8B10BUSE	FPGA TX Interface Width	Internal Data Width
0	0	N/A	8 bits	8 bits
0	1	N/A	16 bits	8 bits
1	0	0	10 bits	10 bits
1	0	1	8 bits	10 bits
1	1	0	20 bits	10 bits
1	1	1	16 bits	10 bits

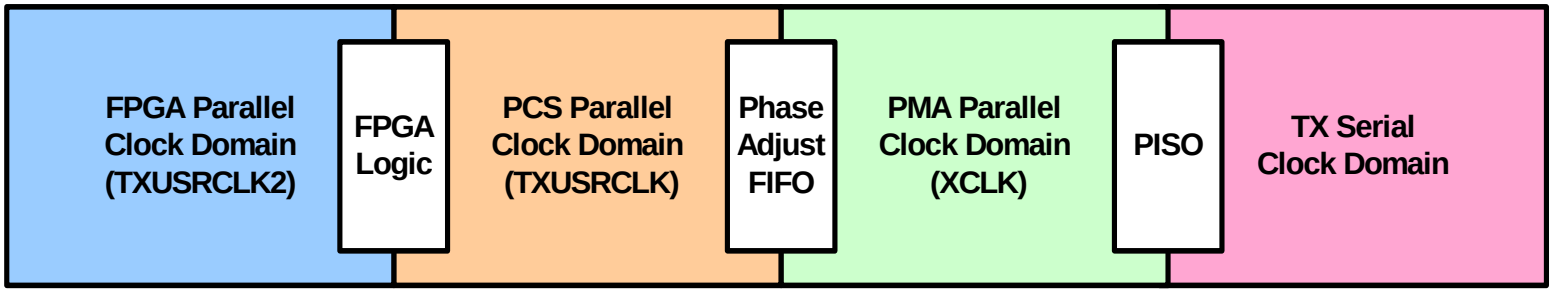
TXUSRCLK and TXUSRCLK2 Clock Inputs

- **The FPGA TX interface includes two parallel clocks: TXUSRCLK and TXUSRCLK2**
  - TXUSRCLK is the internal clock for the PCS logic in the GTP transmitter

    TXUSRCLK = Line Rate / Internal Data Width
  - TXUSRCLK2 is the main synchronization clock for all signals into the TX side of the GTP transceiver

    TXUSRCLK2 = TXUSRCLK, if TXDATAWIDTH = 0 (1-byte interface)

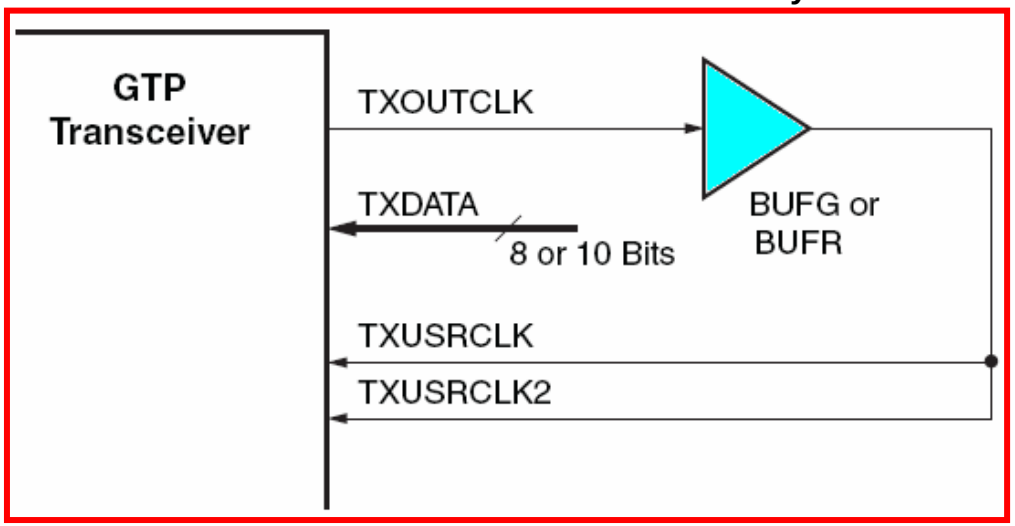
    TXUSRCLK2 = TXUSRCLK / 2, if TXDATAWIDTH = 1 (2-byte interface)
  - XCLK has the same frequency as the TXUSRCLK, but it is not in phase with the TXUSRCLK
  - TXUSRCLK and TXUSRCLK2 must be positive edge aligned, with as little skew as possible between them

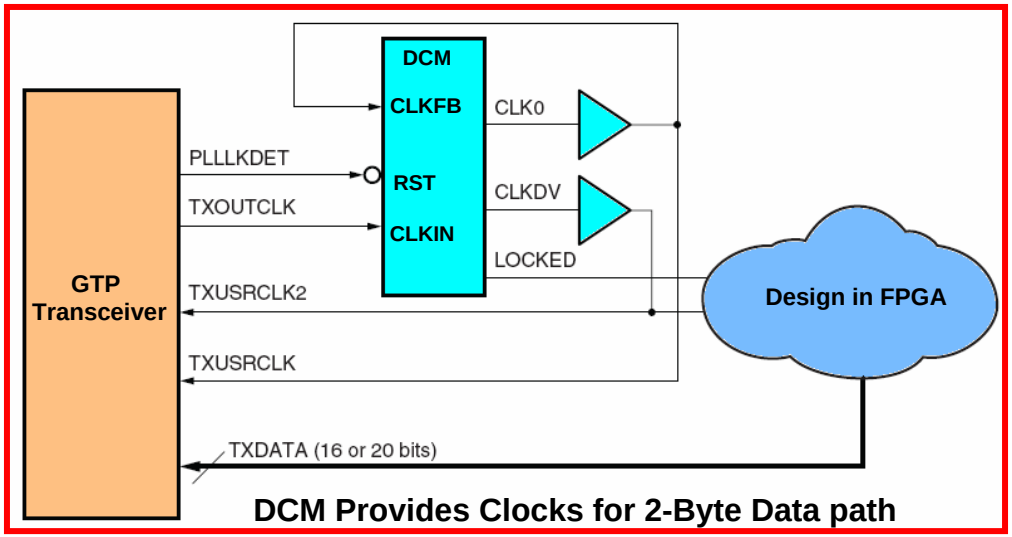


TXUSRCLK and TXUSRCLK2 Clock Inputs

- **The TXUSRCLK, TXUSRCLK2, and CLKIN must have the same oscillator as their source, even though they might run at different frequencies**
  - Thus TXUSRCLK and TXUSRCLK2 must be multiplied or divided versions of CLKIN
  - The GTP transceiver provides access to CLKIN in two ways: the REFCLKOUT pin and the TXOUTCLK pin

TXOUTCLK Drives TXUSRCLK/TXUSRCLK2 for 1-Byte Data Path

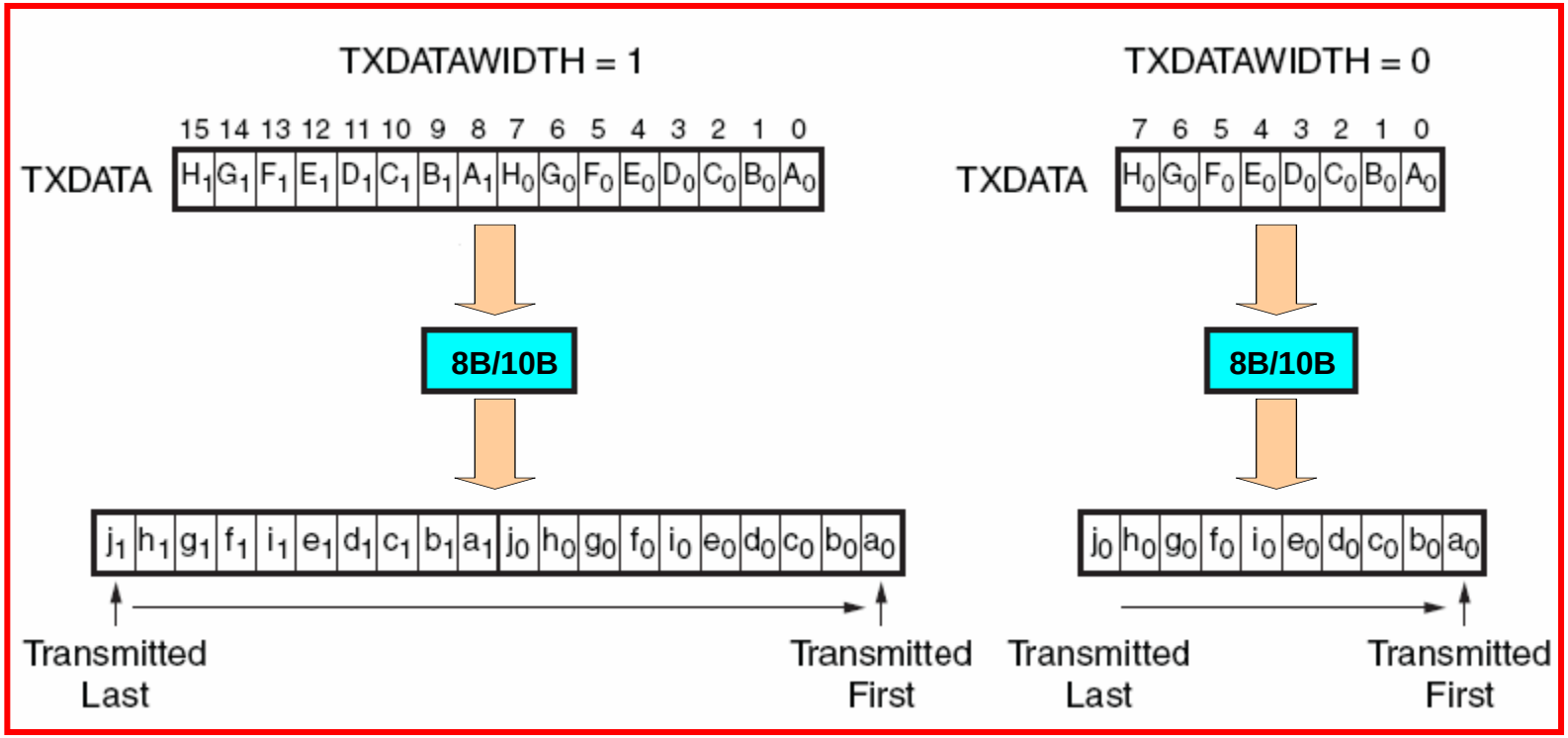




DCM Provides Clocks for 2-Byte Data path

8B/10B Encoder

- **The GTP transceiver includes an 8B/10B encoder to encode TX data without consuming FPGA resources**
  - If encoding is not needed, the block can be disabled to minimize latency
  - To enable the 8B/10B encoder, TXENC8B10BUSE must be driven High
  - The order of the bits after the 8B/10B encoder is the opposite of the order shown in the standard 8B/10B table



TX PRBS Generator

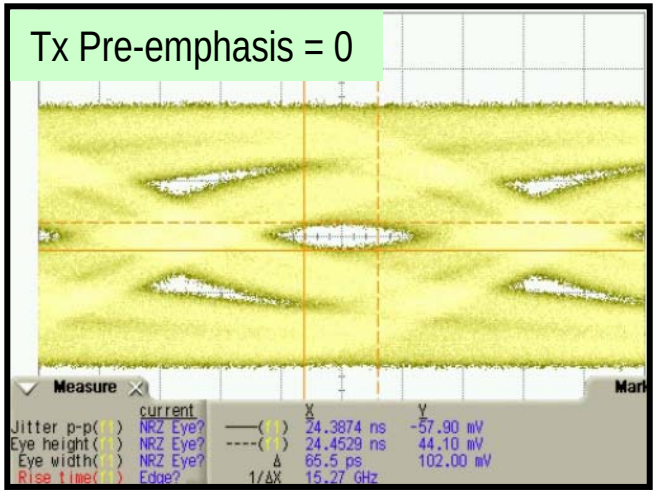
- **Pseudo Random Bit Sequences (PRBS) are commonly used to test the signal integrity of high-speed links**
  - These sequences appear random but have specific properties that can be used to measure the quality of a link
  - The GTP PRBS block can generate several industry-standard PRBS patterns shown in the following table
  - Each GTP transceiver includes a built-in PRBS generator

Name	Polynomial	Length of Sequence (bits)	Consecutive Zeros	Typical Use
PRBS-7	$1 + X^{6} + X^{7}$	$2^{7} - 1$	7	Used to test channels with 8B/10B
PRBS-23	$1 + X^{18} + X^{23}$	$2^{23} - 1$	23	ITU-T Recommendation O.150, Section 5.6. One of the recommended test patterns in the SONET specification
PRBS-31	$1 + X^{28} + X^{31}$	$2^{31} - 1$	31	ITU-T Recommendation O.150, Section 5.8. A recommended PRBS test pattern for 10 Gigabit Ethernet. See IEEE 802.3ae-2002

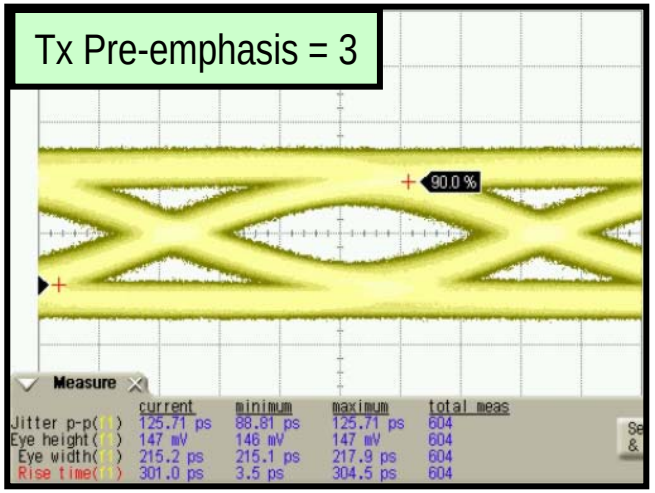
Advanced Tx Pre-emphasis
3.125 Gbps over 40” of FR4

Enabling success from the center of technology™

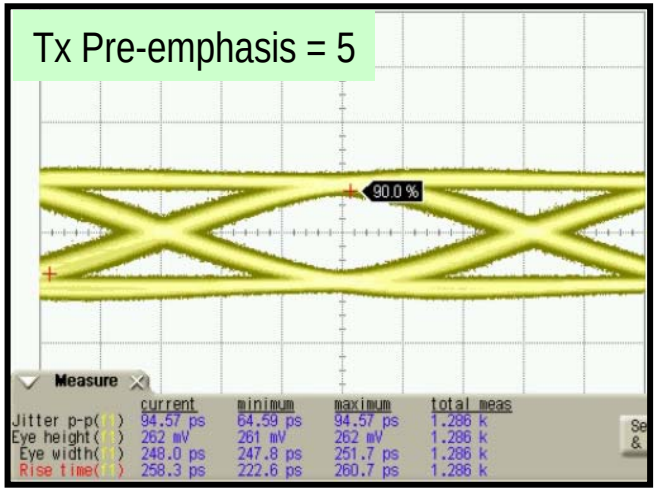
Tx Pre-emphasis = 0



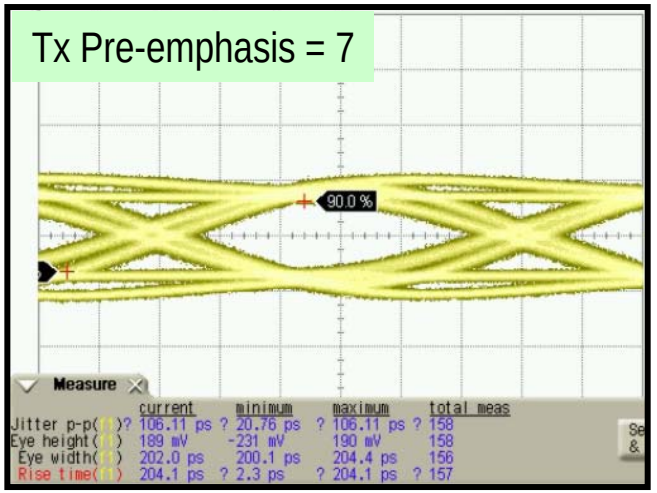
Tx Pre-emphasis = 3



Tx Pre-emphasis = 5



Tx Pre-emphasis = 7



Review Questions

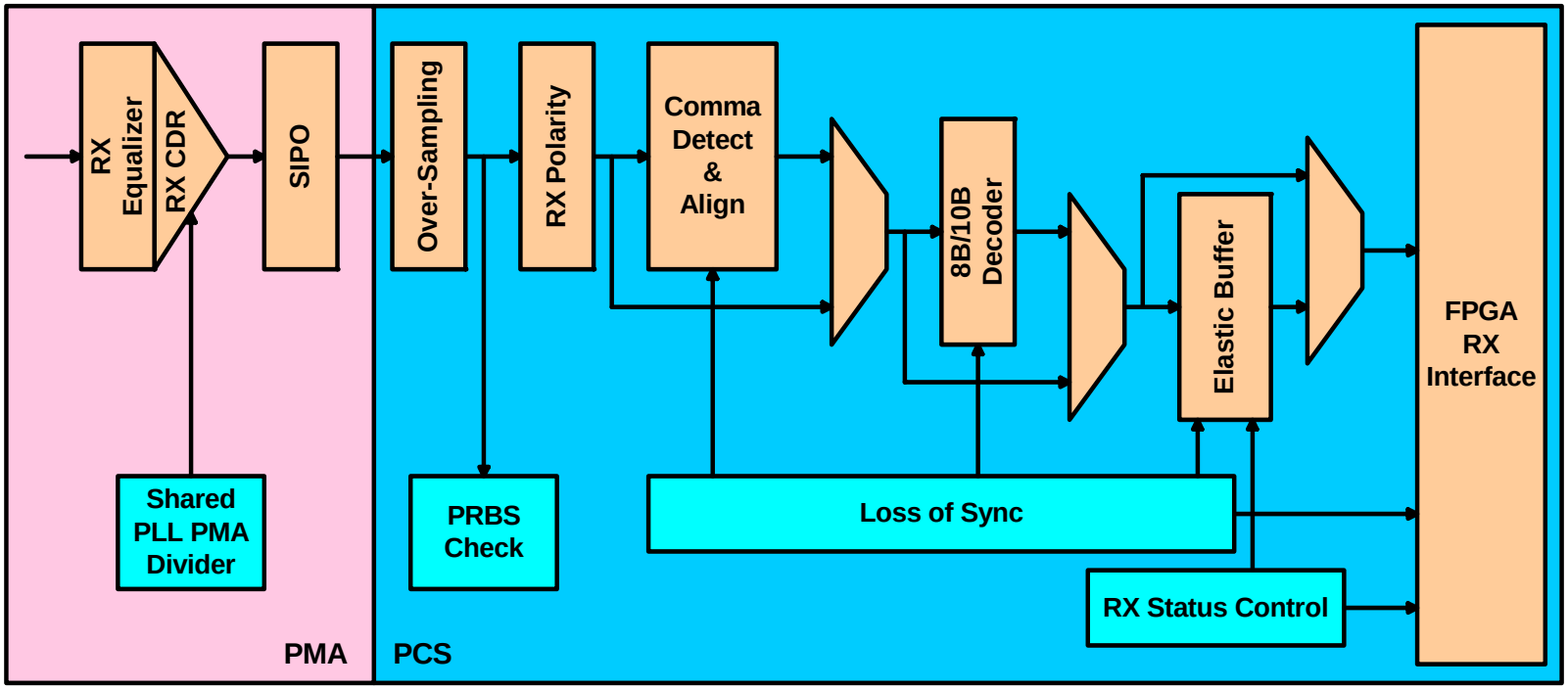
- **How many clock domains are present in the GTP transmitter?**
  - FPGA Parallel Clock domain (TXUSRCLK2)
  - PCS Parallel Clock domain (TXUSRCLK)
  - PMA Parallel Clock domain (XCLK)
  - Serial Clock domain

- **What is the frequency of the TXUSRCLK?**
  - TXUSRCLK = Line Rate / Internal Data Width

Agenda

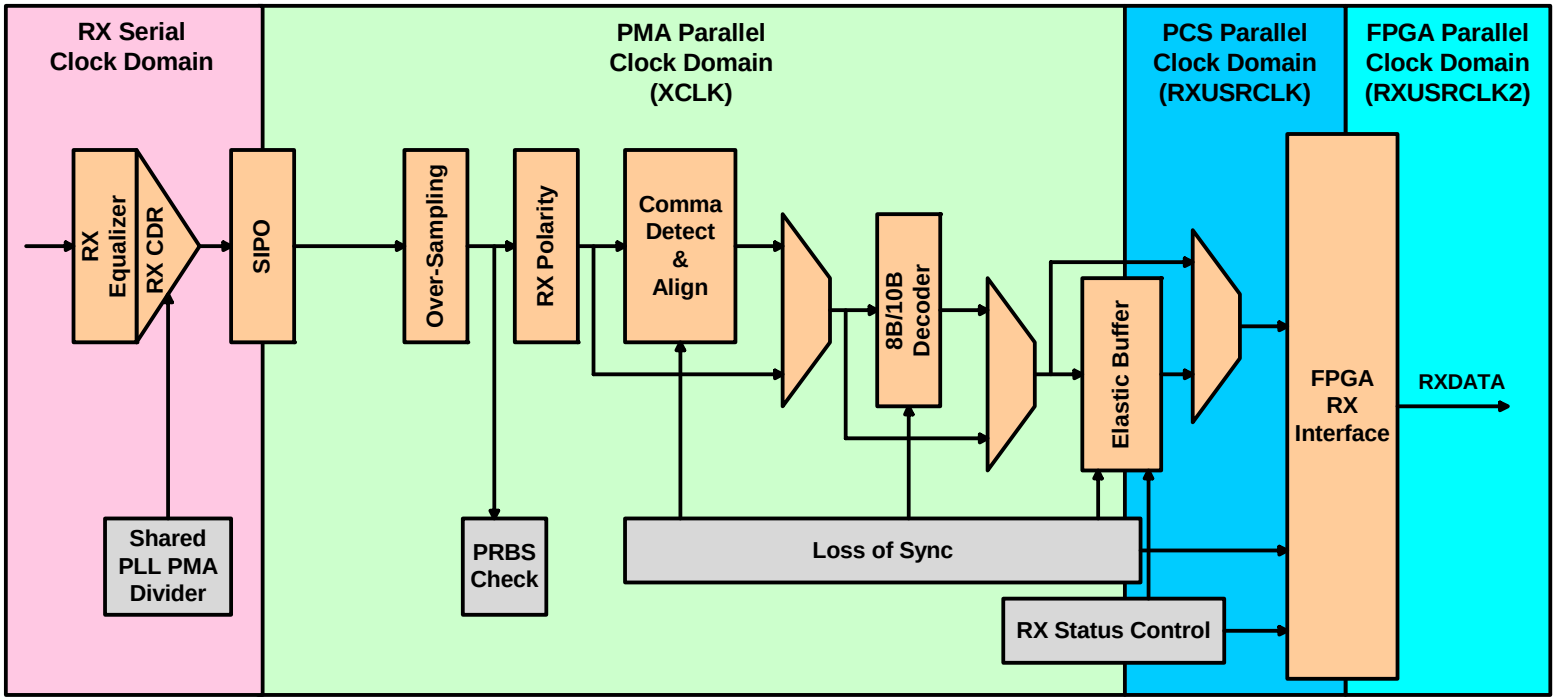
- Introduction to Virtex-5 GTP Transceiver
- Virtex-5 GTP Transceiver Shared Resources
- GTP Transmitter
- **GTP Receiver**
- GTP Loopback and CRC
- GTP Design and Verification Tools
- GTP Demo

- **Each GTP transceiver includes an independent receiver, made up of a PCS and a PMA**
  - High-speed serial data flows from traces on the board into the PMA of the receiver, into the PCS, and finally into the FPGA logic
  - The GTP receiver can invert incoming data using the RX polarity control function. The RXPOLARITY port is driven High to invert the polarity of incoming data



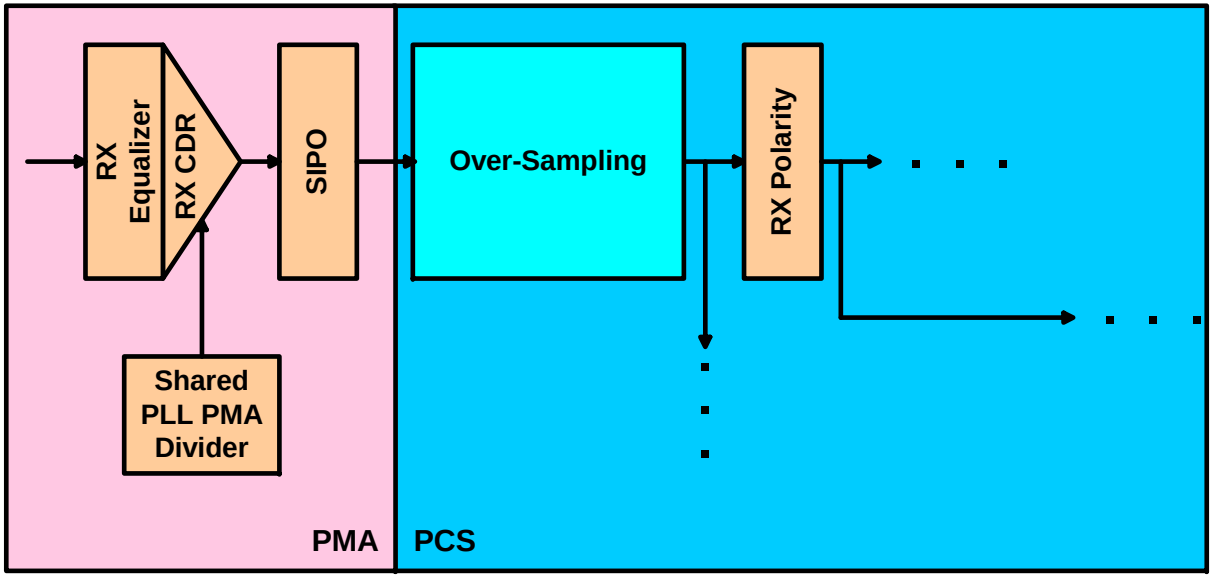
GTP RX Clock Domains

- **The GTP RX data path has two internal parallel clock domains, the PMA parallel XCLK and the PCS parallel RXUSRCLK clock domains**
  - To receive data, the PMA parallel rate must be sufficiently close to the RXUSRCLK rate, and all phase differences between the two domains must be resolved
  - The GTP transceiver includes an RX elastic buffer to resolve differences between the PMA XCLK and RXUSRCLK domain



Over-Sampling

- **Each GTP transceiver includes built-in 5x over-sampling to enable serial rates from 100Mbps to 500Mbps**
  - At these low rates, the regular CDR must operate at 5x the desired line rate to stay within its operating limits
  - The digital over-sampling circuit takes parallel data from the SIPO at 5x the desired line rate and uses the position of bit value transitions to recover a clock
  - The transition points are also used to pick an optimal sampling point to recover 2 bits of data from each set of 10 bits presented



Over-Sampling Operation

- **Over-sampling applies to both transceivers in a GTP_DUAL tile**
  - If over-sampling is activated for one transceiver, it is activated for both
- **Configuring the GTP transceiver to use over-sampling requires the following steps**
  - Configuring the 5x line rate
  - Configuring the PCS internal data path and clocks
  - Activating and operating the over-sampling block
- **The RocketIO GTP Wizard automatically configures the GTP_DUAL tile and makes the over-sampling ports available when generating a GTP wrapper with over-sampling enabled**

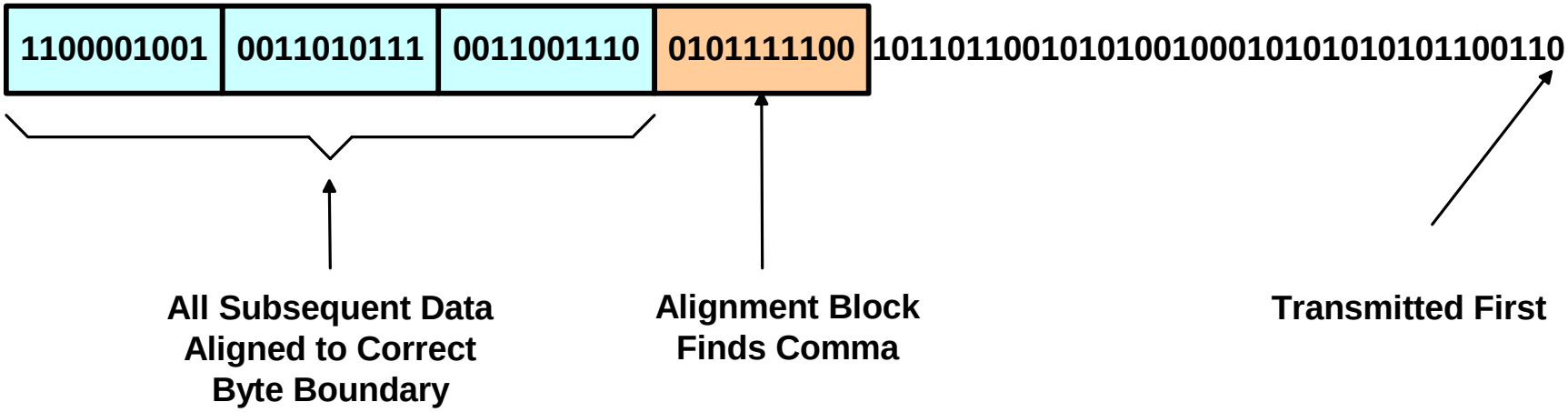
RX PRBS Checker

- **The GTP receiver includes a built-in Pseudo Random Bit Sequence (PRBS) checker**
  - This checker can be set to check for one of three industry-standard PRBS patterns
  - The checker is self-synchronizing and works on the incoming data before comma alignment or decoding
  - This function can be used to test the signal integrity of the channel

Ports	Description
PRBSCNTRESET0 PRBSCNTRESET1	Resets the PRBS error counter. PRBSCNTRESET is applied synchronously and is effective only on its rising edge.
TXENPRBSTST0[1:0] TXENPRBSTST1[1:0]	Receiver test pattern checker control: 00: Disable PRBS checker 01: Enable $2^{7} - 1$ PRBS checker 10: Enable $2^{23} - 1$ PRBS checker 11: Enable $2^{31} - 1$ PRBS checker
RXPRBSERR0 RXPRBSERR1	RXPRBSERR goes High when the number of errors in PRBS testing exceeds the value set by the PRBS_ERR_THRESHOLD attribute.

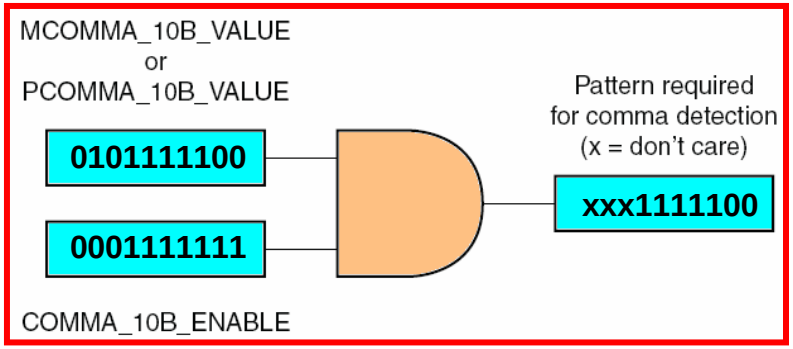
Comma Alignment and Detection

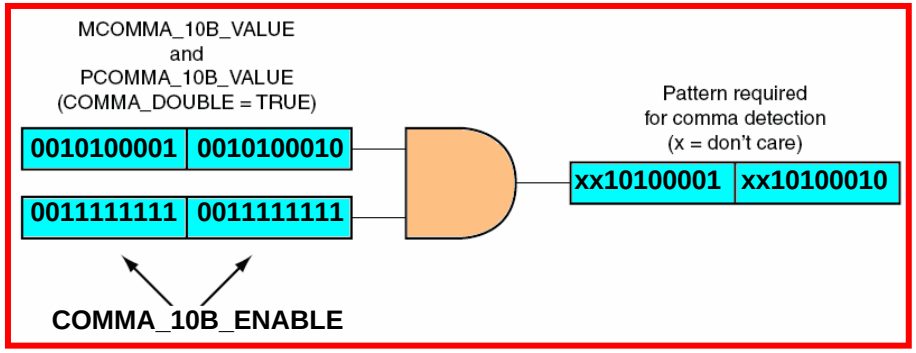
- **Serial data must be aligned to symbol boundaries before it can be used as parallel data**
  - To make alignment possible, transmitters send a recognizable sequence, usually called a *comma*
  - The receiver searches for the comma in the incoming data. When it finds a comma, it moves the comma to a byte boundary so the received parallel words match the transmitted parallel words



Enabling the Comma Alignment

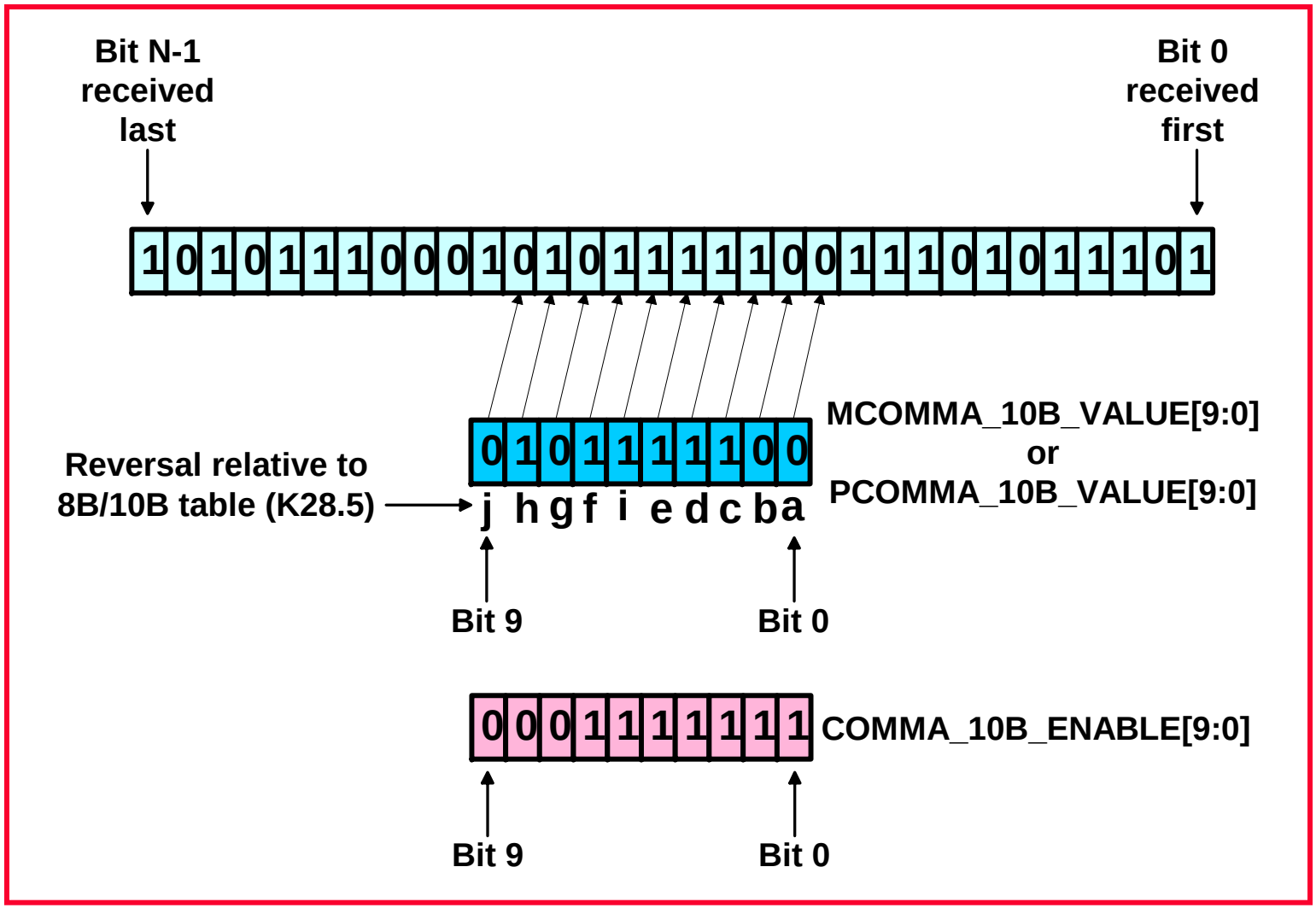
- **To enable the comma alignment block, the RXCOMMADETUSE port is driven High**
  - RXCOMMADETUSE is driven Low to bypass the block completely for minimum latency
  - The MCOMMA_10B_VALUE, PCOMMA_10B_VALUE, and COMMA_10B_ENABLE attributes are used to set the comma pattern that the block searches for in the incoming data stream





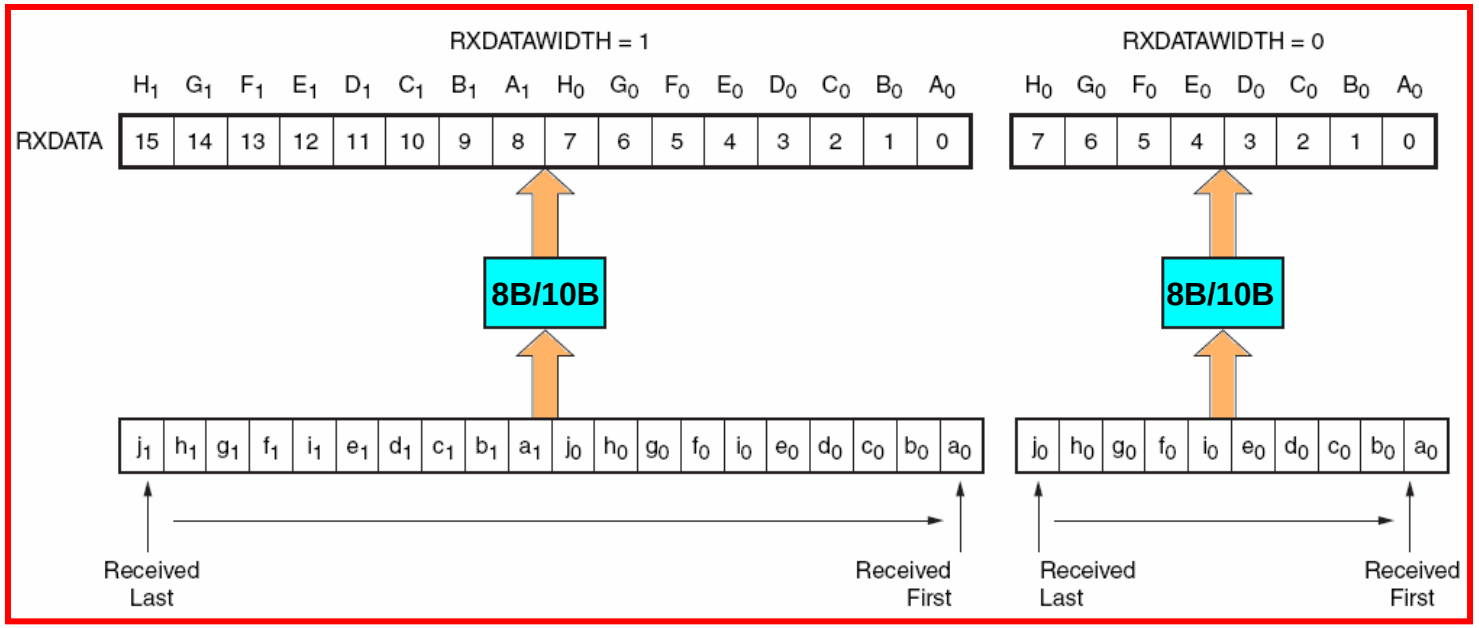
  - If COMMA_DOUBLE is TRUE, the MCOMMA and PCOMMA patterns are combined so that the block searches for two commas in a row

8B/10B Comma Detection Example



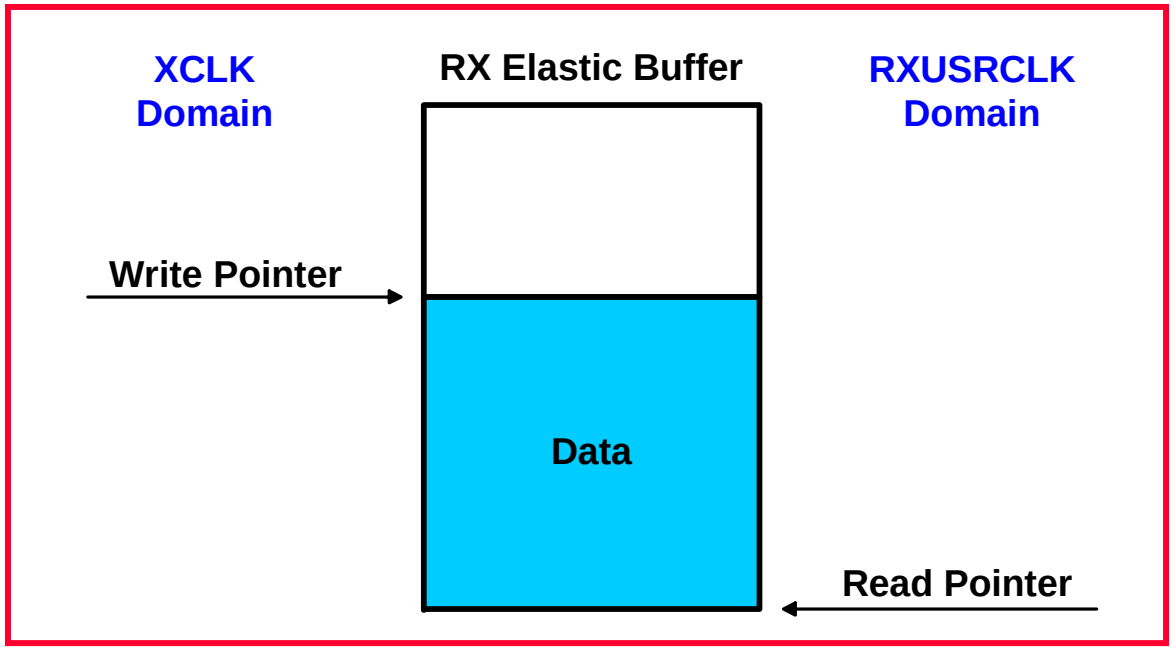
8B/10B Decoder

- **The GTP transceiver includes an 8B/10B decoder to decode RX data without consuming FPGA resources**
  - The decoder includes status signals to indicate errors and incoming control sequences
  - If decoding is not needed, the block can be disabled to minimize latency
  - To enable the 8B/10B decoder, RXDEC8B10BUSE is driven High. INTDATAPATH must also be High (10-bit internal data path) to enable the 8B/10B decoder because it requires 10-bit wide data



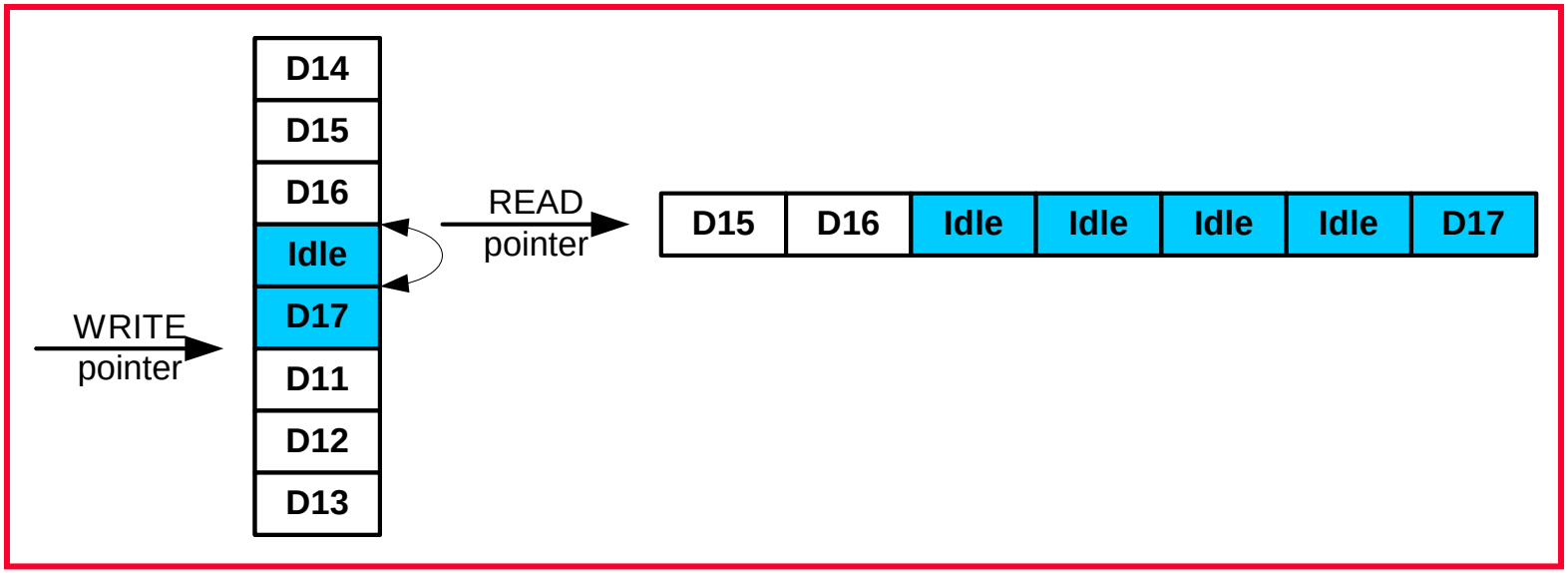
Clock Correction

- **Clock correction is needed when the rate that data is fed into the write side of the receive buffer is either slower or faster than the rate that data is retrieved from the read side of the receive buffer**
  - The rate of write data entering the buffer is determined by the frequency of XCLK clock
  - The rate of read data retrieved from the read side of the buffer is determined by the frequency of RXUSRCLK clock



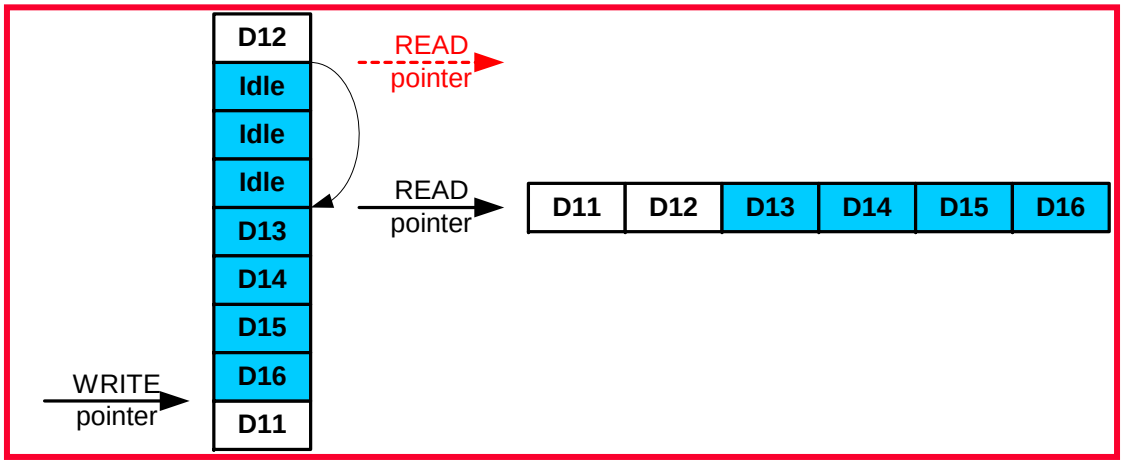
READ faster than WRITE

- **In general, the receive buffer is kept at half full**
  - If READ pointer is too fast, when an idle sequence is encountered, the same sequence will be read out repeatedly until the buffer is half full again
  - If there are no idles, READ pointer will over run WRITE pointer

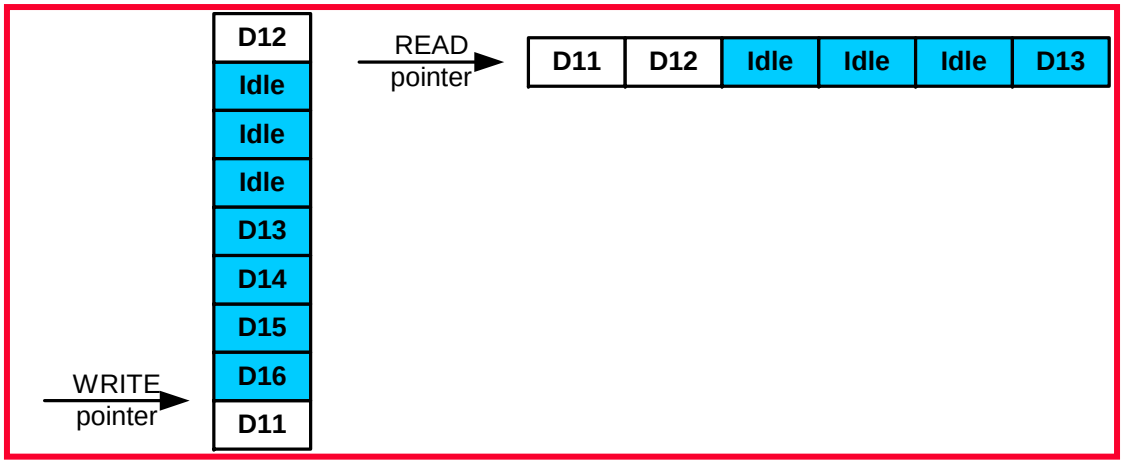


WRITE faster than READ

- **READ pointer will skip up to 4 idles (depending on how far READ is behind WRITE) at once to try to bring the buffer to a half full position**



- **If clock correction is not necessary, idle sequences will not be skipped**



FPGA RX Interface

- **The FPGA RX interface allows parallel received data to be read from the GTP transceiver. For this interface**
  - The width of the data interface (RXDATA) must be configured
  - RXUSRCLK2 and RXUSRCLK must be connected to clocks running at the correct rate
  - The FPGA RX Interface data path resides in the RXUSRCLK2 clock domain
  - The RX internal data path resides in the RXUSRCLK clock domain
  - RX internal data path is always 1-byte wide while the RX FPGA interface can be 1 or 2-byte wide

INTDATAWIDTH	RXDATAWIDTH	RXDEC8B10BUSE	FPGA RX Interface Width	Internal Data Width
0	0	N/A	8 bits	8 bits
0	1	N/A	16 bits	8 bits
1	0	0	10 bits	10 bits
1	0	1	8 bits	10 bits
1	1	0	20 bits	10 bits
1	1	1	16 bits	10 bits

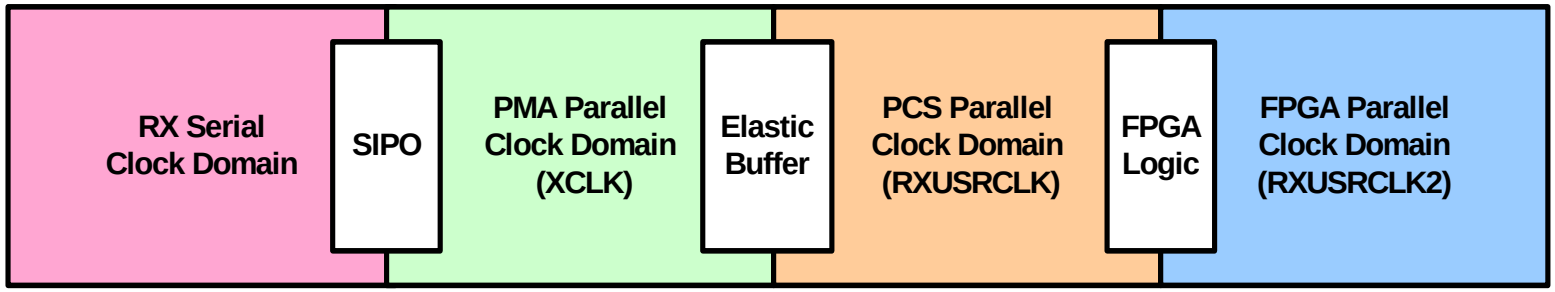
RXUSRCLK and RXUSRCLK2 Clock Inputs

- **The FPGA RX interface includes two parallel clocks: RXUSRCLK and RXUSRCLK2**
  - RXUSRCLK is the internal clock for the PCS logic in the GTP receiver

    RXUSRCLK = Line Rate / Internal Data Width
  - RXUSRCLK2 is the main synchronization clock for all signals into the RX side of the GTP transceiver

    RXUSRCLK2 = RXUSRCLK, if RXDATAWIDTH = 0 (1-byte interface)

    RXUSRCLK2 = RXUSRCLK / 2, if RXDATAWIDTH = 1 (2-byte interface)
  - XCLK has the same frequency as the RXUSRCLK, but it is not in phase with the RXUSRCLK
  - RXUSRCLK and RXUSRCLK2 must be positive edge aligned, with as little skew as possible between them



Review Questions

- **When is the Over-Sampling block used in the GTP receiver?**
  - When the line rate is between 100Mbps and 500Mbps

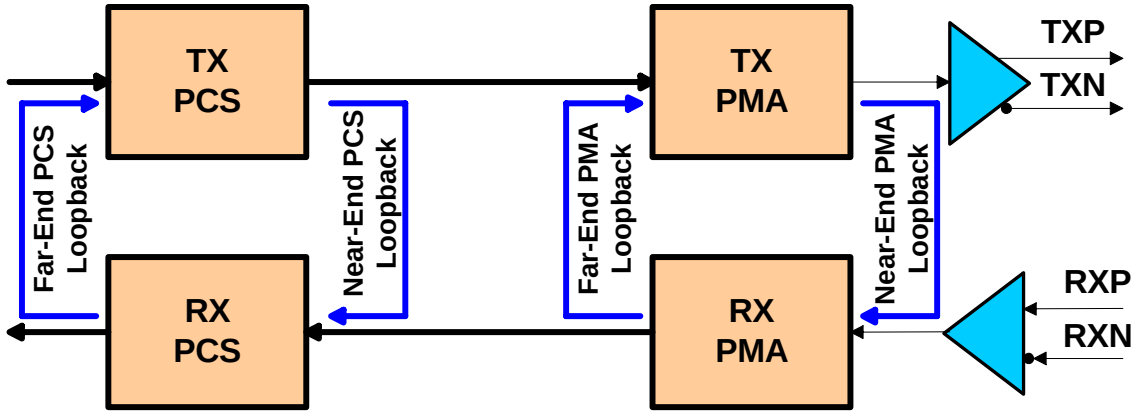
- **When is clock correction needed?**
  - Clock correction is needed when the rate that data is fed into the write side of the receive buffer is either slower or faster than the rate that data is retrieved from the read side of the receive buffer

Agenda

- Introduction to Virtex-5 GTP Transceiver
- Virtex-5 GTP Transceiver Shared Resources
- GTP Transmitter
- GTP Receiver
- **GTP Loopback and CRC**
- GTP Design and Verification Tools
- GTP Demo

Loopback

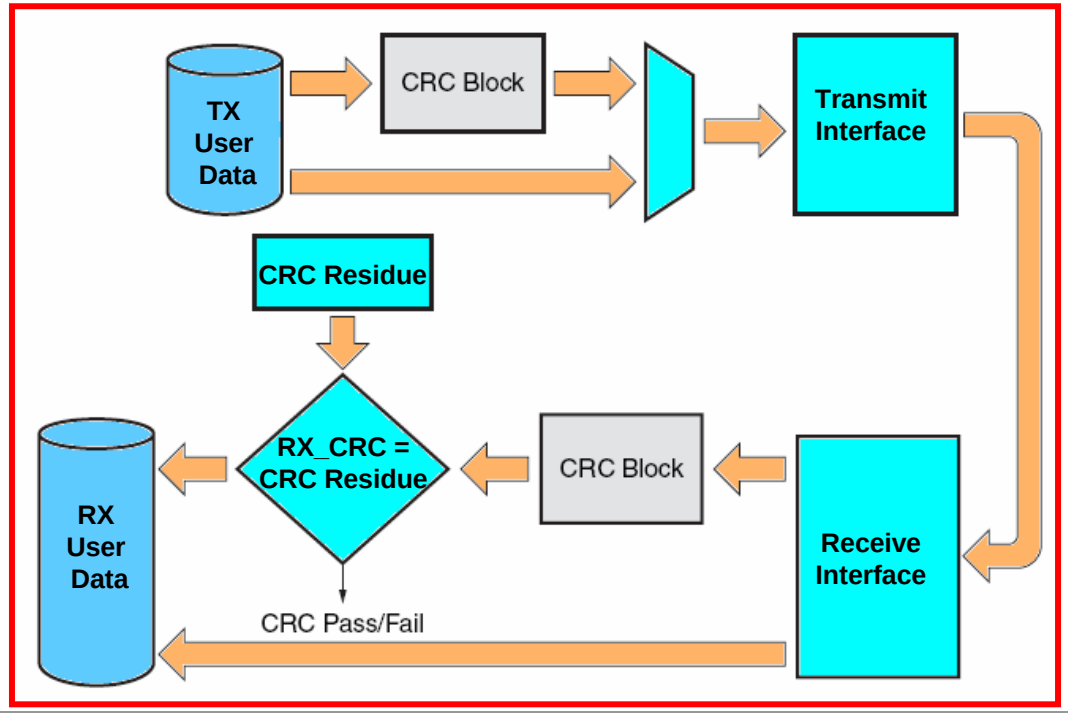
- **Loopback modes are specialized configurations of the transceiver data path where the traffic stream is folded back to the source**
  - Typically, a specific traffic pattern is transmitted and then compared to check for errors



LOOPBACK[2:0]	Function
000	Normal Operation (no loopback)
001	Near-End PCS Loopback
010	Near-End PMA Loopback
011	Reserved
100	Far-End PMA Loopback
101	Reserved
110	Far-End PCS Loopback (PCI Express compliant)
111	Reserved

GTP CRC

- **Each GTP transceiver tile is paired with two CRC hard blocks**
  - The CRC hard blocks can operate independently as two 32-bit input CRC modules (CRC32) or can be combined into a single 64-bit input CRC module (CRC64)
  - The CRC modules use the standard 32-bit Ethernet polynomial for CRC calculation
  - The CRC hard blocks are independent of the transceiver blocks



Agenda

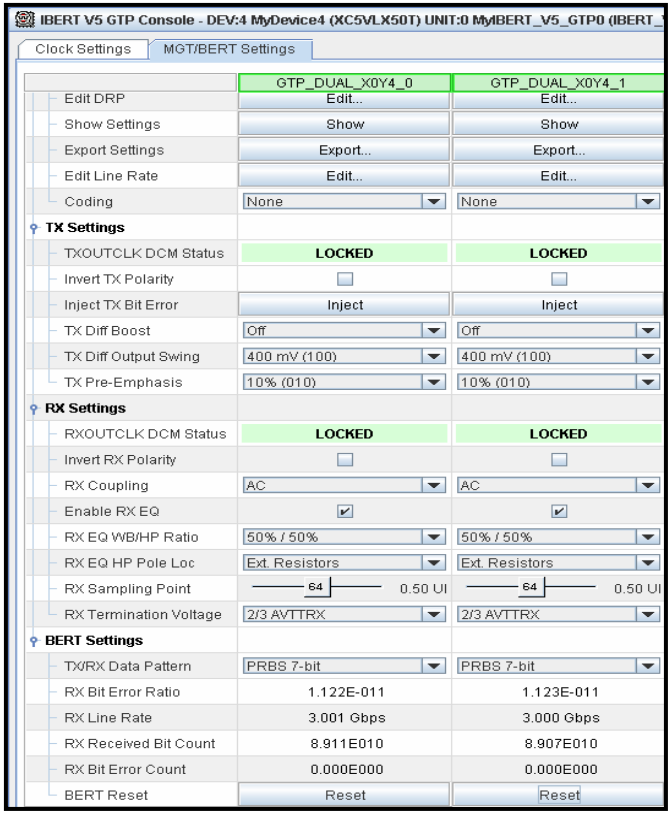
- Introduction to Virtex-5 GTP Transceiver
- Virtex-5 GTP Transceiver Shared Resources
- GTP Transmitter
- GTP Receiver
- GTP Loopback and CRC
- **GTP Design and Verification Tools**
- GTP Demo

RocketIO Wizard

- **Virtex-5 GTP configuration can be complex because of the large number of possible settings**
  - There are numerous ports and attributes available, and many of them are interrelated
  - Xilinx provides a RocketIO wizard to help manage the configuration process of the Virtex-5 GTP
  - The RocketIO Wizard for Virtex-5 automates the task of creating HDL wrapper files to configure the GTP
  - The Wizard's customization GUI allows the user to configure one or more GTPs using pre-defined templates to support popular industry standards, or a custom protocol
  - The wizard is highly recommended for any design that utilizes the Virtex-5 RocketIO GTP

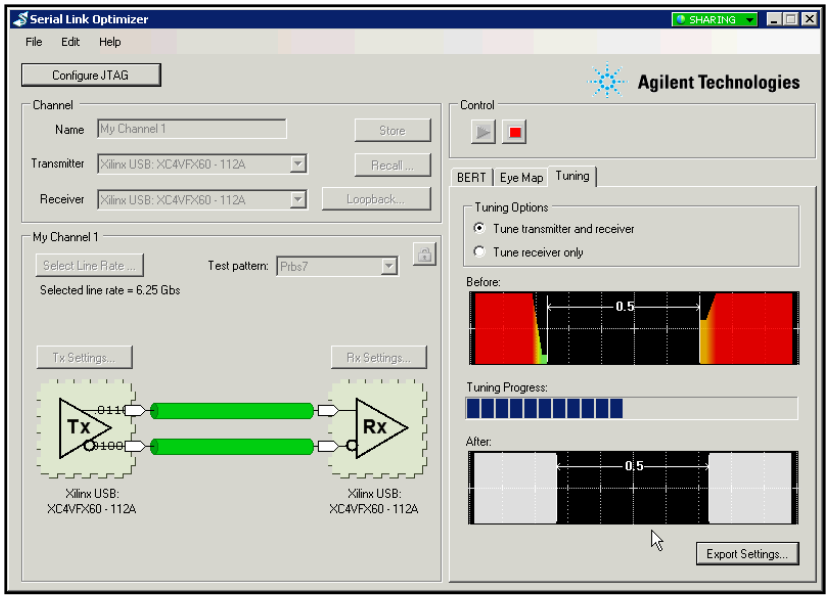
Xilinx ChipScope Pro Serial I/O Toolkit

- **The ChipScope Pro Serial IO Toolkit Integrated Bit Error Ratio Tester (IBERT) provides access to the Virtex-4 RocketIO MGT as well as the Virtex-5 RocketIO GTP and performs bit error ratio analysis on channels composed of one or more MGTs/GTPs**
  - Supports Virtex-4 FX and Virtex-5 LXT/SXT family
  - Allows real-time manual control of all MGT/GTP settings
  - Multiple channels supported
  - Allows run-time adjustment of reference clock settings and line rate
  - Allows displaying or storing the MGT/GTP attributes to the screen or a UCF via the Dynamic Reconfiguration Port (DRP) of the MGT/GTP



Agilent Serial Link Optimizer

- **Extends ChipScope Pro Serial IO Toolkit for Virtex-4 FX and Virtex-5 LXT/SXT**
  - Unique control and access to Xilinx MGT/GTP technology
  - Enabled by IBERT core
  - Channel-oriented graphical user interface
  - Eye mapping and analysis
  - Automatic tuning of TX and RX MGT/GTP settings
  - Graphical margin analysis



Upcoming Serial IO Tool Enhancements

- **ChipScope Pro Serial IO Toolkit**
  - Parameter sweeping for Virtex-5 LXT/SXT – 9.1i SP3
    - Non-graphical loop through parameters
  - Support for Xilinx Virtex-5 FXT (GTX) – 9.2 (May 2007)

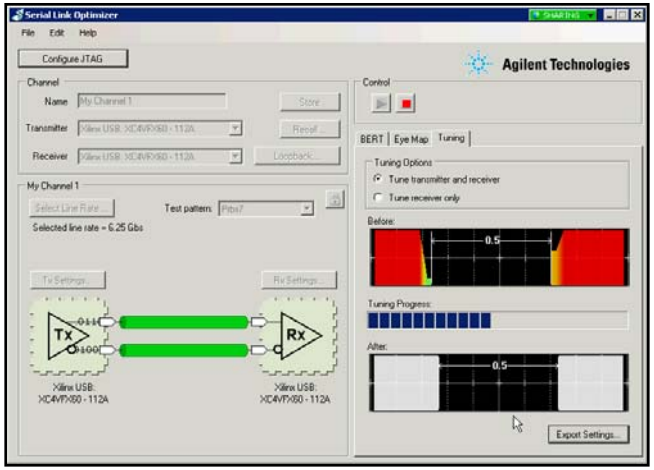
- **Agilent Serial Link Optimizer**
  - Support for Virtex-5 LXT/SXT – 9.1i SP3
  - Support for Virtex-5 FXT – 9.2
    - Advanced capability using GTX transceivers

Xilinx and Agilent On-Chip Debug and Verification Solutions

Enabling success from the center of technology™

- **ChipScope Pro On-chip Debug Solution**
  - 60-Day Evaluation version
- **ChipScope Pro Serial IO Toolkit**
  - Support for Virtex-4 FX and Virtex-5 LXT/SXT
  - Free limited time offer

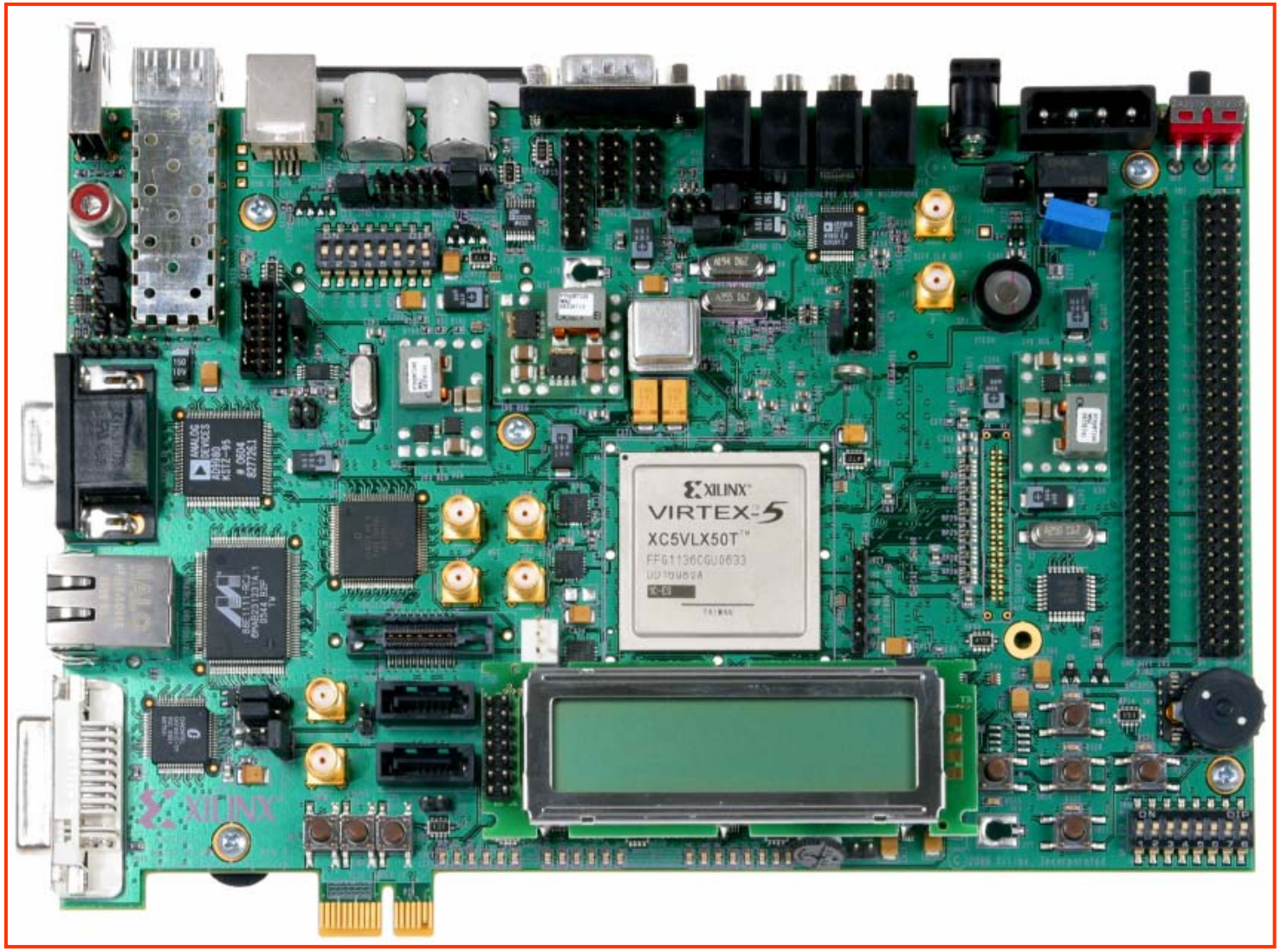
- **Agilent Serial Link Optimizer**
  - Extends ChipScope Pro Serial IO Toolkit
  - Support for Virtex-4 FX and Virtex-5 LXT/SXT
  - Available through Avnet
  - US List Price $495 (1-year node-locked license)
  - US List Price $750 (1-year floating license)



Agenda

- Introduction to Virtex-5 GTP Transceiver
- Virtex-5 GTP Transceiver Shared Resources
- GTP Transmitter
- GTP Receiver
- GTP Loopback and CRC
- GTP Design and Verification Tools
- **GTP Demo**

Xilinx ML505 Development Board



GTP Demo

- **Using the ChipScope Pro Serial IO Toolkit to perform Bit Error Ratio (BER) analysis of one or more GTPs**
  - Using the ChipScope Pro Core Generator to create an IBERT Core design
  - Implementing the design using the ISE in batch mode
  - Analyzing the design using the ChipScope Pro Analyzer and the IBERT Console