Minerva
SATA Controller IP Core

Product Specification

Version 1.0 (Draft)

Notices
Minerva SATA Controller IP Core Product Specification

First edition: Dec 2012

Date modified: June 6, 2013

For information, contact Minerva LLC.

10/3 build 5, Akademichesky ave., Tomsk, Russia, 635055

More information can be found at: http://www.minerva-tec.com

For Technical Support, please contact the Minerva Technical Support Team: tsup@minerva-tech.com

CONTENTS

Terms, Definitions, Abbreviations

GT (Xilinx FPGA RocketIO Gigabit Transceiver) — FPGA's Xilinx primitives (GTP, GTX, GTH, GTZ) that are intended for implementation of high-speed data interfaces.

PCS (Physical Coding Sublayer) — PHY coding sub layer. Contains block of "point" selection and the single-byte data alignment, 8b/10b encoder and decoder, the elastic buffer, etc.

OOB (Out-of-Band) — bunch of SATA interface signals that are implemented as group of the similar type of primitives and interactions between them, are intended to establish a connection with the device.

FIS (Frame Information Structure) — SATA interface data packet.

Introduction

Overview

SATA Controller IP Core is an implementation of the transport and link layer for SATA interface. It provides communications systems implemented in FPGAs Xilinx (Virtex-5, Virtex-6, Spartan-6 and 7 Series) with devices via SATA II.

Features

- The physical layer (PHY) is implemented using Xilinx FPGA RocketIO Gigabit Transceiver primitives;
- Supports both SATA I and SATA II.

Application

- Embedded storage system;
- RAID controller;
- High speed and large capacity data acquisition system.

1 Functional Description

1.1 Block Diagram

A simplified block diagram of SATA Controller IP Core is shown in Figure 1. SATA Controller IP Core includes the following blocks:

- The transport layer (**Transport Layer**);
- The inter-layer interface (**Link Layer interface**);
- The coding data and the calculation of the cyclic checksum (**Scrambler and CRC**);
- The link layer finite state machine (**Link Layer FSM**);

- The connection set (**Link-up controller**).

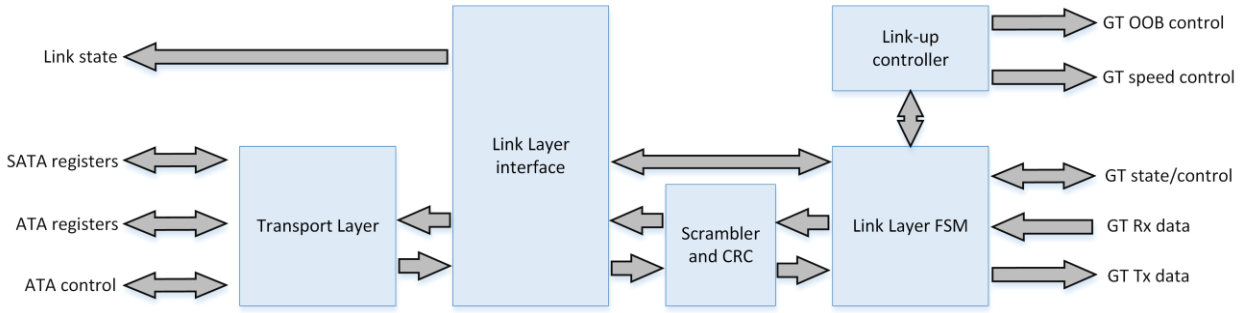

Figure 1: A simplified block diagram of SATA Controller IP Core

SATA Controller has two clock domains: the system clock domain and the link layer clock domain. The system clock domain provides clock for transport layer and works with a clock signal of the processor bus. Link layer clock domain running on the clock signal generated by the PLL GT. Link layer clock domain frequency depends on the current transmission rate.

Received and transmitted data buffers can accommodate maximum FIS size and have the capacity 2049 of 32-bit words each of them (2048 data words and title for the Data FIS).

1.2 Functioning

To set the GT on the required data rate is used the dynamic reconfiguration controller port (DRP). DRP controller is an external in relation to SATA Controller. Such implementation allows to configure and to use a second GT of pair tile for other purposes.

When the system reset is done, Link-up controller connects with device at 3 GB/s data rate. If the device is found but the connection is not established, then an attempt is repeated at a data rate of 1.5 Gb/s.

If the connection at 1.5 Gb/s is failed - then a flag (***unsup_spd***) will be set and indicate that the device is working with an unsupported data rate. Attempts to connect at different speeds will be repeated infinitely.

Once the connection is successfully established then ***link_up*** flag will be set. ***unsup_spd*** flag will be cleared if it was previously installed. ***link_up*** flag will be reset if the connection is broken. Then Link Layer FSM performs following tasks:

- handshakes with the opposite device link layer of the FIS transmitting/receiving;
- adds/removes FIS wrapper – SOF and EOF primitives;
- controls the process of FIS transmission/receiving;
- adds to the traffic flow of a pair of primitives ALIGN every 254 sent 32 bit words.

Scrambler and CRC calculation block encodes/decodes the data, calculates CRC for them and adds CRC to the traffic flow. These blocks also compares the received and calculated CRC and removes it from the stream.

Link Layer Interface block contains the received and transmitted data buffers, the synchronization clock domains, the state and control link layer signals.

Transport Layer block contains ATA registers with control signals and two SATA registers. The behavior of Transport Layer block conforms to the SATA specification except two issues: data register is 32 bits instead of 16 bit and the DMA request signal is divided into two (the DMA request for data transfer and DMA request to receive data).

2 Implementation

2.1 Synthesis Parameters

SATA Controller IP Core is shown on the Table 1.

Table 1: Synthesis parameters of the SATA Controller IP Core

Parameter	Valid values	Description
SYNT_A_RST	true/false	[1)] Allow the synthesis of asynchronous reset signal.

Notices:

1. Collaborative synthesis of synchronous and asynchronous reset leads to the additional overhead of the FPGA resources. Preference should be given to synchronous restart of a less resource-intensive.

2.2 Interface Description

SATA Controller IP Core interface ports are shown on the Figure 2. SATA Controller signals are divided into groups in accordance with the clock domains. Signals the system clock domain are prefixed ***sys_*** and the link layer clock domain — ***lnk_***. All signals are synchronous within the corresponding domains except the system asynchronous reset ***sys_a_rst***, ***phy_rdy*** and ***rx_eidle*** signals. The synthesis of both synchronous and asynchronous reset leads to the additional overhead of the FPGA resources. Preference should be given to synchronous restart of a less resource-intensive.

Table 2 contains general port information, followed by a more detailed description of each port.

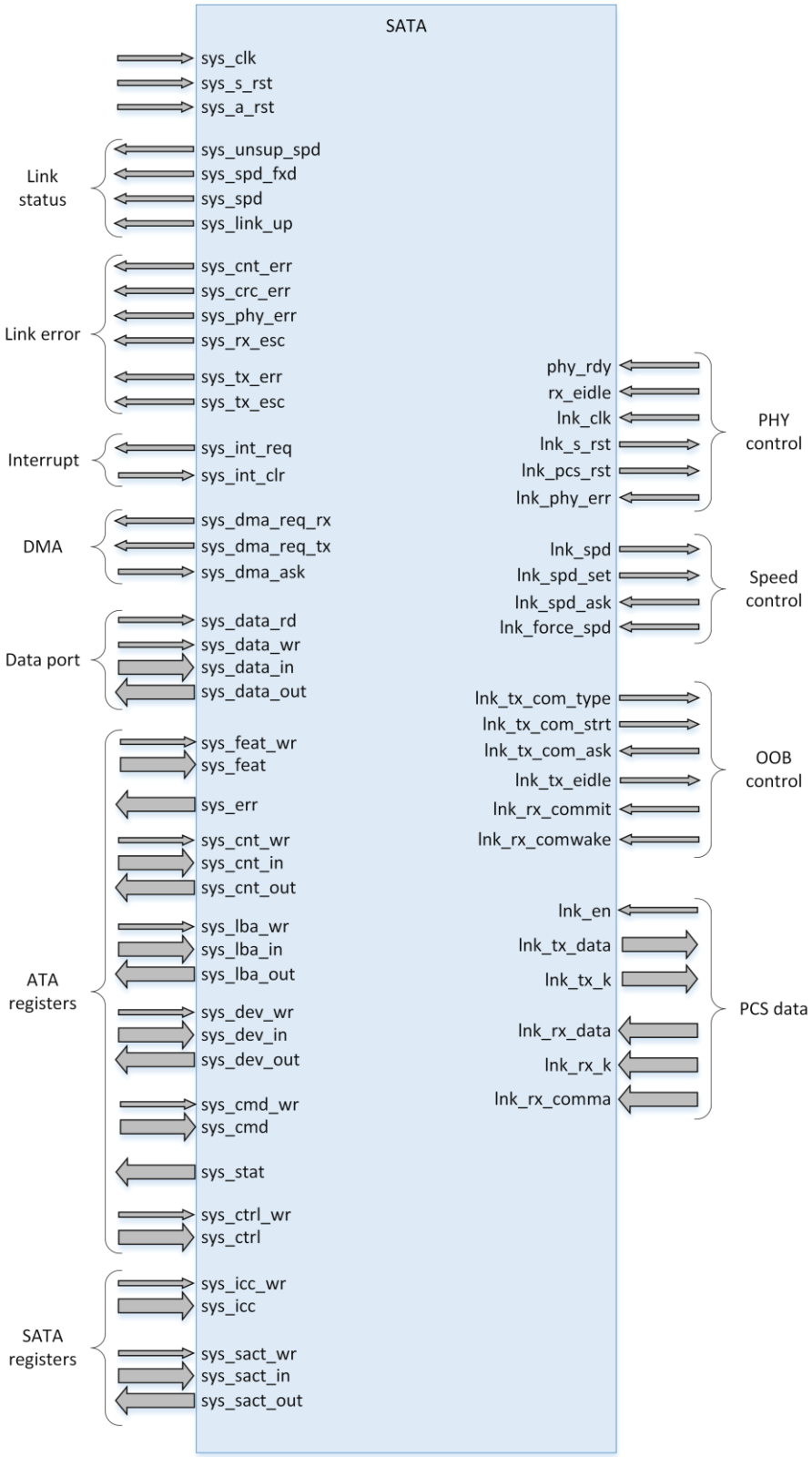

Figure 2: SATA Controller IP Core interface

Table 2: Port description

Signal	Direction	Описание
System clock domain signals: synchronization and reset		
sys_a_rst	Input	System clock domain asynchronous reset signal. Active level - high
sys_clk	Input	System clock domain signal. Clocking is performed on the rising edge
sys_s_rst	Input	System clock domain synchronous reset signal. Active level - high
System clock domain signals: link state		
sys_unsup_spd	Output	Connection has been established with an unsupported data rate
sys_spd_fxd	Output	Data transfer rate is set
sys_spd	Output	Data transfer speed: ’1’ — 3 Gb/s; ’0’ — 1,5 Gb/s
sys_link_up	Output	Connection is established
System clock domain signals: link error		
sys_tx_err	Output	Data transfer is completed with an error (R_ERR primitive was sent by neighboring link layer)
sys_tx_esc	Output	The device requested a forced transaction termination (SYNC primitive was sent by neighboring link layer during FIS transferring)
sys_cnt_err	Output	Length of the received FIS does not match its type
sys_crc_err	Output	The checksum of the received FIS and calculated checksum are different
sys_phy_err	Output	Physical layer error
sys_rx_esc	Output	The device requested a forced transaction termination (SYNC primitive was sent by neighboring link layer during FIS transferring)
System clock domain signals: interruption and DMA		
sys_int_req	Output	Interruption request
sys_int_clr	Input	Reset of the interruption request
sys_dma_req_rx	Output	Write DMA request
sys_dma_req_tx	Output	Read DMA request
sys_dma_ack	Input	DMA acknowledgment
System clock domain signals: data port		
sys_data_rd	Input	Read
sys_data_wr	Input	Write
sys_data_in[31:0]	Input	Transmitted data
sys_data_out[31:0]	Output	Received data
System clock domain signals: ATA registers		
sys_feat_wr[1:0]	Input	Write *Features* register
sys_feat[15:0]	Input	Register *Features* input
sys_err[7:0]	Output	Register *Error* output
sys_cnt_wr[1:0]	Input	Write *Count* register
sys_cnt_in[15:0]	Input	Register *Count* input
sys_cnt_out[15:0]	Output	Register *Count* output

sys_lba_wr[5:0]	Input	Write *LBA* register
sys_lba_in[47:0]	Input	Register *LBA* input
sys_lba_out[47:0]	Output	Register *LBA* output
sys_dev_wr	Input	Write *Device* register
sys_dev_in[7:0]	Input	Register *Device* input
sys_dev_out[7:0]	Output	Register *Device* output
sys_cmd_wr	Input	Write *Command* register
sys_cmd[7:0]	Input	Register *Command* input
sys_stat[7:0]	Output	Register *Status* output
sys_ctrl_wr	Input	Write *Device Control* register
sys_ctrl[7:0]	Input	Register *Device Control* input
System clock domain signals: SATA registers		
sys_icc_wr	Input	Write *ICC* register
sys_icc[7:0]	Input	Register *ICC* input
sys_sact_wr	Input	Write *SActive* register
sys_sact_in[31:0]	Input	Register SActive input
sys_sact_out[31:0]	Output	Register SActive output
Link layer clock domain signals: physical layer control		
lnk_clk	Input	Clock signal. Timing is performed on the rising edge
lnk_s_rst	Output	Synchronous reset signal of the link layer clock domain
lnk_pcs_rst	Output	PCS reset
lnk_phy_err	Input	Physical layer error
Link layer clock domain signals: data transfer speed control		
lnk_force_spd[1:0]	Input	[1)] Mode setting of the data transfer speed: "00" — auto detection; "01" — 1,5 Gb/s; "10" — 3 Gb/s; "11" — reserved
lnk_spd	Output	Data transfer rate: ’1’ — 3 Gb/s; ’0’ — 1,5 Gb/s
lnk_spd_set	Output	Set the data transfer rate
lnk_spd_ack	Input	Data transfer rate setting is completed
Link layer clock domain signals: OOB		
lnk_tx_com_type	Output	Type the transmitted sequence COM последовательности: ’0’ — COMRESET; ’1’ — COMWAKE.
lnk_tx_com_strt	Output	Start the transfer COM sequence
lnk_tx_com_ack	Input	COM sequence transfer is completed
lnk_tx_eidle	Output	Set the state of the output of the differential pair in the "electrical idle" (active during COM sequences transmission)
lnk_rx_cominit	Input	COM sequence COMINIT was received
lnk_rx_comwake	Input	COM sequence COMWAKE was received
Link layer clock domain signals: data		
lnk_en	Output	[2)] Allowing the action link layer FSM.
lnk_tx_data[31:0]	Output	Transferred data

lnk_tx_k	Output	The least byte of the transmitted data (K type symbol)
lnk_rx_data[31:0]	Input	Received data
lnk_rx_k[1:0]	Input	The least byte of the received data (K type symbol)
lnk_rx_comma[1:0]	Input	The least byte of the received data (“comma”)

Notes:

1. Restarting the data transfer rate adjustment process is required after changing (to reset the SATA controller or to send reset command to the device);
2. It must be set to '1' when the physical layer has aligned data on 32-bit words.

2.3 Application Diagram

Figure 3 shows the application diagram of SATA Controller IP Core based on GTP Virtex-5 family.

Data processing in PCS GTP is carrying out byte-by-byte at a system clock frequency of 300 MHz at 3 Gb/s data transfer rate, and 150 MHz at 1,5 Gb/s data transfer rate. GTP is configured with a 16-bit data (at a clock frequency of 150 MHz and 75 MHz, respectively).

As an internal clock signal of GTP (TXUCLK0, RXUCLK0 inputs) the signal synthesized in PLL GTP from reference clock **ref_clk** is used. As a link layer domain clock signal the synthesized signal in PLL GTP which frequency is divided on 2 is used. Division of frequency is carried out used of BUFR primitive.

The external block is necessary for adapting 16 bit GTP and 32 bit SATA Controller (is presented as Gear 16:32 on the diagram).

The signal of that the GTP has delivered word aligned data appears earlier than the valid data. Therefore, the delay line on 6+CLK_COR_MIN_LAT steps is necessary. CLK_COR_MIN_LAT — the GTP parameter indicating the minimum latency of the elastic buffer.

For setting transfer rate data the dynamic reconfiguration port controller of GTP is used. On the diagram it is presented as GTP DRP Controller.

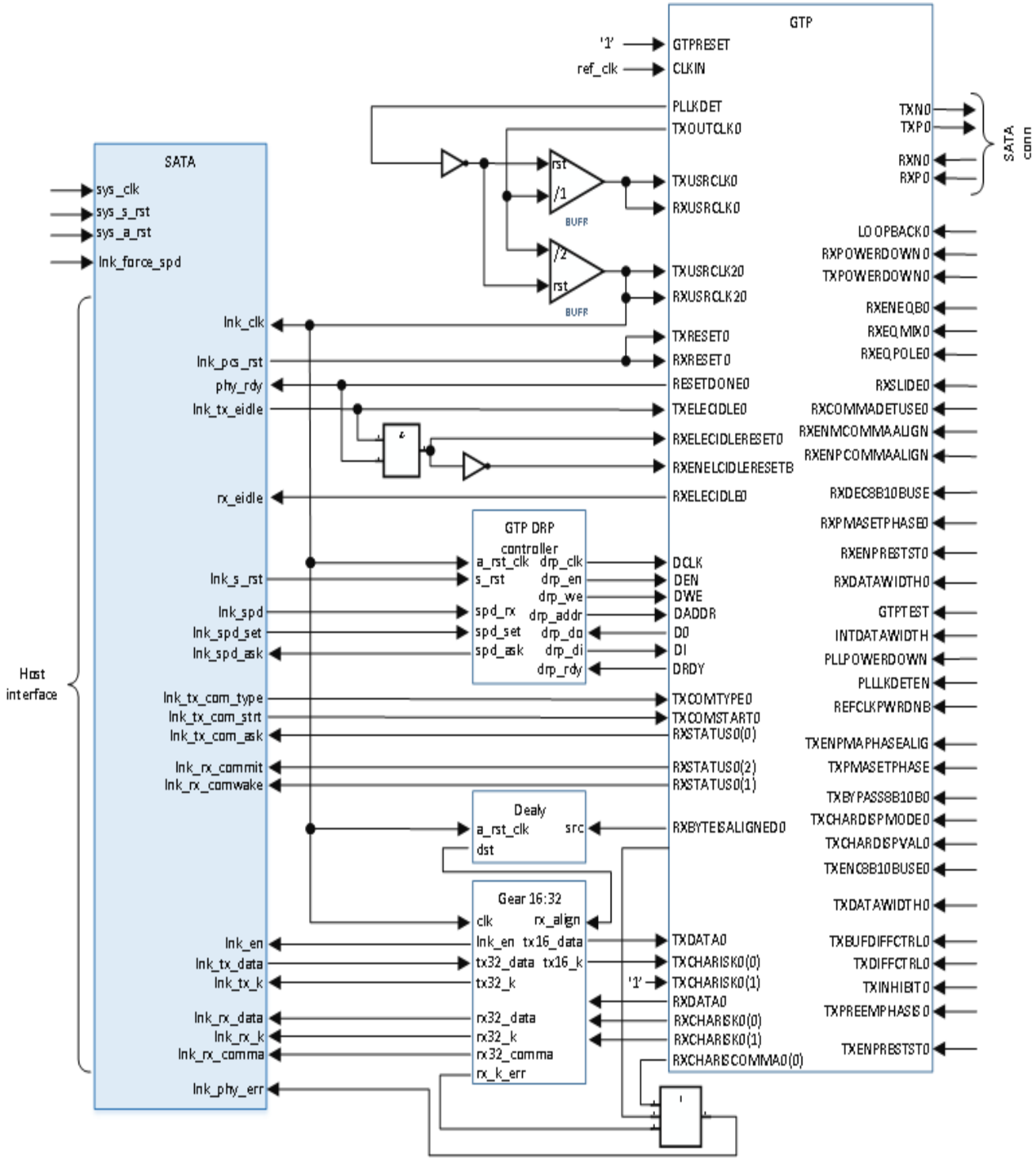

Figure 3 Application diagram of the SATA Controller IP Core

2.4 GT Parameters

Some GT parameters depend on the frequency of a reference clock signal and hardware realization of the interface. Below the example of a configuration (generic) of GTP for ML506 board (150 MHz reference clock) is given.

```
generic map
{
------------------------ Simulation-Only Attributes -------------------
SIM_RECEIVER_DETECT_PASS0 => TRUE,
SIM_RECEIVER_DETECT_PASS1 => TRUE,
SIM_MODE => "FAST", -- Set to Fast Functional Simulation Model
SIM_GTPRESET_SPEEDUP => 0, -- Set to 1 to speed up sim reset
SIM_PLL_PERDIV2 => x"14d", -- Set to the VCO Unit Interval time
------------------------------ Shared Attributes ------------------------
--------------------------- Tile and PLL Attributes ----------------------
CLK25_DIVIDER => 6,
CLKINDC_B => TRUE,
OOB_CLK_DIVIDER => 6,
OVERSAMPLE_MODE => FALSE,
PLL_DIVSEL_FB => 2,
PLL_DIVSEL_REF => 1,
PLL_TXDIVSEL_COMM_OUT => 1,
TX_SYNC_FILTERB => 1,
PCS_COM_CFG => x"1680a0e",
--------------------- Transmit Interface Attributes ---------------------
------------------- TX Buffering and Phase Alignment --------------------
TX_BUFFER_USE_0 => TRUE,
TX_XCLK_SEL_0 => "TXOUT",
TXRX_INVERT_0 => "00000",
---------------------- TX Serial Line Rate settings ----------------------
PLL_TXDIVSEL_OUT_0 => 1,
---------------------- TX Driver and OOB signaling -----------------------
TX_DIFF_BOOST_0 => TRUE,
------------------ TX Pipe Control for PCI Express/SATA ----------------
COM_BURST_VAL_0 => "0101",
--------------------- Receive Interface Attributes ---------------------
------------ RX Driver,OOB signaling,Coupling and Eq,CDR -------------
TERMINATION_CTRL => "10100",
TERMINATION_OVRD => FALSE,
-- RX0 EQ
AC_CAP_DIS_0 => FALSE,
RCV_TERM_GND_0 => FALSE,
RCV_TERM_MID_0 => TRUE,
RCV_TERM_VTTRX_0 => FALSE,
TERMINATION_IMP_0 => 50,
OOBDETECT_THRESHOLD_0 => "100",
-- RX0 CDR
PMA_CDR_SCAN_0 => X"6c07640", -- Control of the CDR sampling point
PMA_RX_CFG_0 => X"09f0089", -- Controls loop filter of the CDR
--------------------- RX Serial Line Rate Attributes -------------------
PLL_RXDIVSEL_OUT_0 => 1,
PLL_SATA_0 => FALSE,
------------------------ PRBS Detection Attributes ---------------------
PRBS_ERR_THRESHOLD_0 => x"00000001",
---------------- Comma Detection and Alignment Attributes -------------
-- Align comma to (1: either or 2: the even) byte within a 2-byte datapath
ALIGN_COMMA_WORD_0 => 2,
-- Determines which bits of MCOMMA/PCOMMA must be matched to incoming data
-- and which bits can be any value.
COMMA_10B_ENABLE_0 => "1111111111",
-- Specifies whether a comma match consists of either a comma plus or a comma minus alone,
-- or whether both are required in the sequence.
COMMA_DOUBLE_0 => FALSE,
-- Defines comma minus to raise RXCOMMADET and aligns the parallel data
MCOMMA_10B_VALUE_0 => "1010000011",
```

```
MCOMMA_DETECT_0 => TRUE, — Controls raising of RXCOMMADET on comma minus
-- Defines comma plus to raise RXCOMMADET and aligns the parallel data.
PCOMMA_10B_VALUE_0 => "0101111100",
PCOMMA_DETECT_0 => TRUE, — Controls raising of RXCOMMADET on comma plus
-- Selects between sliding in the PMA or in the PCS. Legal values are PCS (default) and PMA
RX_SLIDE_MODE_0 => "PCS",
-- Decoder 8/10
-- Enables detection of negative 8B/10B commas (RXCHARISCOMMA)
DEC_MCOMMA_DETECT_0 => TRUE,
-- Enables detection of positive 8B/10B commas (RXCHARISCOMMA)
DEC_PCOMMA_DETECT_0 => TRUE,
-- Limits the set of commas to which RXCHARISCOMMA responds (only K28.1, K28.5, K28.7)
DEC_VALID_COMMA_ONLY_0 => TRUE,
------------------ RX Loss-of-sync State Machine Attributes -----------
RX_LOSS_OF_SYNC_FSM_0 => FALSE,
RX_LOS_INVALID_INCR_0 => 8,
RX_LOS_THRESHOLD_0 => 128,
-------------- RX Elastic Buffer and Phase alignment Attributes -------
RX_BUFFER_USE_0 => TRUE,
RX_XCLK_SEL_0 => "RXREC",
------------------------ Clock Correction Attributes ------------------
CLK_CORRECT_USE_0 => TRUE,
CLK_COR_ADJ_LEN_0 => 4, — number of bytes repeated or skipped in a clock correction
-- length of the sequence that the transceiver matches
-- to detect opportunities for clock correction
CLK_COR_DET_LEN_0 => 4,
CLK_COR_INSERT_IDLE_FLAG_0 => FALSE,
CLK_COR_KEEP_IDLE_0 => FALSE,
CLK_COR_MAX_LAT_0 => 28, — Maximum RX elastic buffer latency (3-48) 18
CLK_COR_MIN_LAT_0 => 20, — Minimum RX elastic buffer latency (3-48) 16
CLK_COR_PRECEDENCE_0 => TRUE,
CLK_COR_REPEAT_WAIT_0 => 0,
CLK_COR_SEQ_1_1_0 => "01" & X"BC", --"0110111100", — BC
CLK_COR_SEQ_1_2_0 => "00" & X"4A", --"0001001010", — 4A
CLK_COR_SEQ_1_3_0 => "00" & X"4A", --"0001001010", — 4A
CLK_COR_SEQ_1_4_0 => "00" & X"7B", --"0001111011", — 7B
CLK_COR_SEQ_1_ENABLE_0 => "1111",
CLK_COR_SEQ_2_1_0 => "0000000000",
CLK_COR_SEQ_2_2_0 => "0000000000",
CLK_COR_SEQ_2_3_0 => "0000000000",
CLK_COR_SEQ_2_4_0 => "0000000000",
CLK_COR_SEQ_2_ENABLE_0 => "0000",
CLK_COR_SEQ_2_USE_0 => FALSE,
-- Sequences are matched the output to the 8B/10B decoder (else the input)
RX_DECODE_SEQ_MATCH_0 => TRUE,
------------------------ Channel Bonding Attributes -------------------
CHAN_BOND_1_MAX_SKEW_0 => 1,
CHAN_BOND_2_MAX_SKEW_0 => 1,
CHAN_BOND_LEVEL_0 => 0, — 0 to 7. See UG for details
CHAN_BOND_MODE_0 => "OFF", — "MASTER", "SLAVE", or "OFF"
CHAN_BOND_SEQ_1_1_0 => "0000000000",
CHAN_BOND_SEQ_1_2_0 => "0000000000",
CHAN_BOND_SEQ_1_3_0 => "0000000000",
CHAN_BOND_SEQ_1_4_0 => "0000000000",
CHAN_BOND_SEQ_1_ENABLE_0 => "0000",
CHAN_BOND_SEQ_2_1_0 => "0000000000",
CHAN_BOND_SEQ_2_2_0 => "0000000000",
CHAN_BOND_SEQ_2_3_0 => "0000000000",
CHAN_BOND_SEQ_2_4_0 => "0000000000",
CHAN_BOND_SEQ_2_ENABLE_0 => "0000",
CHAN_BOND_SEQ_2_USE_0 => FALSE,
CHAN_BOND_SEQ_LEN_0 => 1,
PCI_EXPRESS_MODE_0 => FALSE,
------------------ RX Attributes for PCI Express/SATA ---------------
RX_STATUS_FMT_0 => "SATA",
SATA_BURST_VAL_0 => "100",
SATA_IDLE_VAL_0 => "100",
SATA_MAX_BURST_0 => 7,
```

```
SATA_MAX_INIT_0 => 22,
SATA_MAX_WAKE_0 => 7,
SATA_MIN_BURST_0 => 4,
SATA_MIN_INIT_0 => 12,
SATA_MIN_WAKE_0 => 4,
TRANS_TIME_FROM_P2_0 => x"003c",
TRANS_TIME_NON_P2_0 => x"0019",
TRANS_TIME_TO_P2_0 => x"0064"
};
```

2.5 Implementation

The software should transfer information about acceptable delays of signals upon transition from one clock domain to another. In a case with Xilinx following constraints in ucf the file must be considered:

```
NET "sys_clk" TNM_NET = "sys_clk";
TIMESPEC "TS_sys_clk" = PERIOD "sys_clk" <CPU bus frequency> MHz;
NET "*<>*/gtp_tx_out_clk" TNM_NET = "gtp_tx_out_clk";
TIMESPEC "TS_gtp_tx_out_clk" = PERIOD "gtp_tx_out_clk" 300 MHz;
NET "*sys_clk" TNM=FFS "sys_clk_dmn";
NET "*<>*/lnk__clk" TNM=FFS "lnk_clk_dmn";
TIMESPEC "TS_s2l" = FROM "sys_clk_dmn" TO "lnk_clk_dmn" TS_gtp_tx_out_clk*4 DATAPATHONLY;
TIMESPEC "TS_l2s" = FROM "lnk_clk_dmn" TO "sys_clk_dmn" TS_sys_clk*2 DATAPATHONLY;
```

This ucf file provides to synthesizer information about system clock domain and synthesized clock in PLL GTP frequency. It also assigns names for clock domains and specifies maximum delay in propagation signals between the respective clock domains. Propagation signal delay in the transition from one clock domain to another must not exceed two periods such signal of receiver domain.

If clock signals are renamed during synthesis, it will be necessary to edit VHDL file with SATA Controller and specify:

attribute KEEP : string;
attribute KEEP of sys_clk: signal is "TRUE";
attribute KEEP of lnk_clk: signal is "TRUE";

2.6 Resource and Performance

Synthesis XST report of resource expenses and maximum operating clock frequencies is shown on Table 3. Asynchronous reset signals are not used (SYNT_A_RST = false). In addition to resources in table, 4 BRAM's are used (two for each unit data buffer).

Table 3 Resources and performance

Family	Chip	Frequency, MHz	Slice Flip Flops	Slice LUTs	LUT Flip Flop pairs
Virtex-5™	XC5VSX50T-1FF1136	200	652	1195	1359
		165	652	1000	1094
Spartan-6™	XC6SLX75T-4CSG484	165	599	1102	503

Virtex-6™	XC6VCX75T-2FF484	265	630	1126	496
		240	423	990	567
Kintex-7™	7K70TFBG676-2	300	425	992	567
Atrix-7™	7A100TCSG324-2	240	427	994	567

3 Additional information

3.1 Recommended Design Experience

The experience of writing HDL, knowledge of Xilinx software tools and SoC building are required for IP Core integration. Minerva Technology provides design services to produce a fully integrated solution if required.

3.2 Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Тел: +1 408-559-7778
Факс: +1 408-559-7114
URL: www.xilinx.com

3.3 Technical Support

For technical support contact the Minerva Technical Support Team:
tsup@minerva-tech.com

3.4 Ordering Information

Please contact Minerva Sales department for pricing and additional information about the product.
sales@minerva-tech.com