

SATA-IP Device reference design manual

Rev1.2 02-Jun-09

1. Introduction

Serial ATA (SATA) is an evolutionary replacement for the Parallel ATA (PATA) physical storage interface. SATA interface increases speed transfer to be 1.5 Gbps for SATA-I and 3.0 Gbps for SATA-II. To communication by SATA protocol, there are four layers in its architecture, i.e. Application, Transport, Link, and Phy.

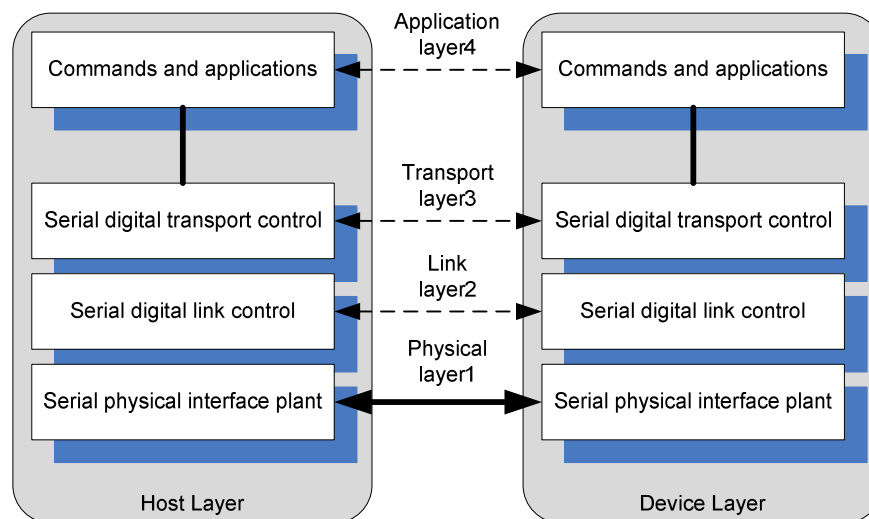


Figure1 SATA Communication Layer

The Application layer is responsible for overall ATA command execution, including controlling Command Block Register accessed. The Transport layer is responsible for placing control information and data to be transferred between the host and device in a packet/frame, known as a Frame Information Structure (FIS). The Link layer is responsible for taking data from the constructed frames, encoding or decoding each byte using 8b/10b, and inserting control characters such that the 10-bit stream of data may be decoded correctly. The Physical layer is responsible for transmitting and receiving the encoded information as a serial data stream on the wire.

This reference design provides evaluation system which implements all SATA communication layers for Device side to transfer high speed data with SATA-II Host PC. The SATA-IP core is designed to operate with GTP transceiver of the Virtex-5 platform and this SATA-IP reference design is implemented on ML506/505 Evaluation board. More details are described as follows.

2. Environment

This reference design is based on the following environment as Figure2.

- ML506 Platform (For ML505, changing only target device)
- ISE 10.1.03 / EDK 10.1.03
- SATA Host PC to connect via cross-over SATA cable at J40 on ML506 (“Cross-over SATA cable” is attached to ML506/505 board kit.)
- Serial (RS232C) communication, connect RS232C cable to P3 on ML506 (Set baud rate=115,200 / data=8bit / Non-Parity / Stop=1bit)
Serial communication is not mandatory, but be helpful for debug environment.

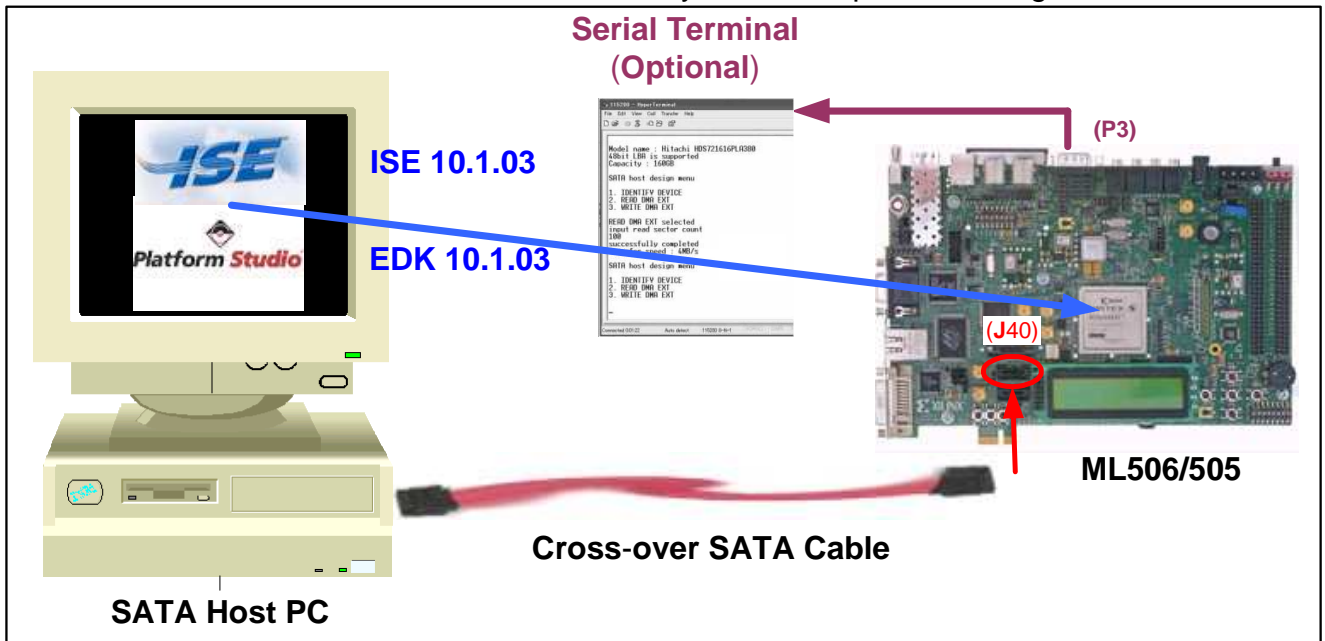


Figure2 Reference design environment

Refer to “SATA_IP Device Demo Instruction” for operation procedure of this reference design. For evaluation version, IP-Core has 1-hour time limitation to use. After 1-hour use, IP-core will stop any data transfer.

3. Hardware description

- SATA IP Device design implementation on Virtex5 FPGA

As shown in Figure3, SATA IP consists of only Link Layer and some part of Transport Layer, so that users need to prepare other Layer such as PHY Layer and Transport Layer by themselves. This reference design describes Transport Layer and PHY Layer example based on ML506 Evaluation board from Xilinx.

In this reference design, ML506 board will emulate SATA RAMDISK by using on board DDR2-SDRAM memory. WindowsXP on Host PC can recognize this RAMDISK as a SATA peripheral disk drive and can execute read or write operation.

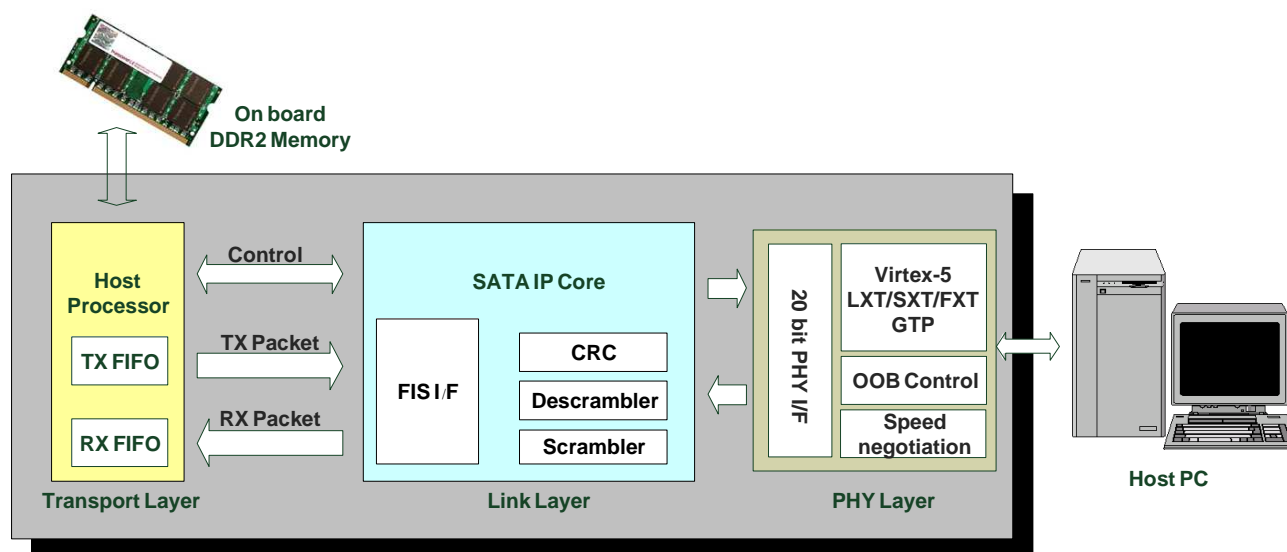


Figure3 Connection between Link Layer and Transport/PHY Layer

- PHY Layer

Virtex-5 LXT/SXT/FXT has a built-in high-speed serial circuit in its GTP block, and PHY Layer of SATA will be implemented with this GTP block. PHY Layer also includes OOB (Out-of-Band) Control, Automatic Speed Negotiation logic to support SATAII/I, and 20bit PHY interface for Link Layer communication.

This reference design is modified from Xilinx application note (XAPP870), available from Xilinx website, so that PHY Layer is also optimized with GTP resource usage of Virtex5. SATA characteristic report of rpt087 is also available from Xilinx web site. PHY modification is designed because the sequence of OOB control in Host and Device is different.

Before building user board, user must read carefully and must follow design guide line described in UG196 (Virtex-5 FPGA RocketIO GTP Transceiver User Guide).

PHY Layer circuit source code in this reference design is stored in “sata2phy_ml505.v” that also includes “dev_oob_control.v” and “speed_neg_control.v” module. “sata2phy_ml505.v” is designed to support three modes, i.e. Fixed-SATAI, Fixed-SATAII, and Auto-negotiation. One of three modes is selecting by define one of three parameters in line 20-22 of “sata2phy_ml505.v” file, as shown in Figure4. Default mode in reference design is Auto-negotiation.

```

14:
15: // Select
16: // - FIXI    for Fixed-SATAI speed PHY design
17: // - FIXII   for Fixed-SATAII speed PHY design
18: // - AUTONEG for Auto speed negotiation PHY design
19:
20: `define FIXI
21: `define FIXII
22: `define AUTONEG // Auto speed negotiation
23:

```

Figure4 Header in “sata2phy_ml505” for selecting SATA PHY function

SATA connector (J40) with auto-negotiation function can be connected both SATA-I and SATA-II Host PC, but only SATA-I/SATA-II Host PC can be connected for Fixed-SATAI/Fixed-SATAII function. Though two SATA-PHYs are available in each GTP_Tile of Virtex-5 LXT/SXT device, only one PHY is recommended to use for one GTP_Tile in this reference design. The advantage of Fixed-SATAII/SATAI function is DCM utilization which is required only one DCM instead of two DCMs like in auto-negotiation function.

Note: Changing parameter in “sata2phy_ml505.v” requires to re-implement reference design hardware, so this features is permitted only production version.

- Transport Layer

Transport Layer in this reference design uses NPI interface to operate with MicroBlaze. Similar to typical SATA controller, this reference design will build FIS data on the main memory and communicate with Link Layer by DMA mechanism.

Transport Layer circuit source code in this reference design is stored in “npi_sata_target.vhd” that also includes SATA IP Core and PHY Layer instance. Compared with Host reference design, hardware in Transport Layer has no change. Thus, the different part from Host design is existed only in the software. Same as Host reference design, Transport Layer in this Device reference design also use MicroBlaze as a local processor and also use MPMC (Multi-port memory controller) as a main memory controller.

● Build connection circuit between Memory controller and IP Core

Interface signal of SATA IP core is shown in Table 1. Transport Layer interface signals are separated into two groups, i.e. transmit and receive. Transport signal waveform of data transmit transaction and data receive transaction is shown at Figure5 and 6 respectively. Figure7 shows detailed block diagram of logic connection. Table 2 is a register mapping from MicroBlaze side.

Software on MicroBlaze along with DMA mechanism will transmit FIS data on the main memory to the Link Layer of SATA IP, or will receive data from the Link Layer to the main memory. During receiving data from the Link Layer, control logic circuit will transfer Data FIS to the different address space for different FIS type by checking Data FIS header information. Also, during transmitting data, control logic circuit will automatically add Data FIS header at the top of the data packet. With this mechanism, user logic do not need to mind about header information control for data management.

Signal	Signal Direction	Description
Common Interface Signal		
trn_reset	In	Reset SATA IP core. Active high.
trn_link_up	Out	Transaction link up is asserted when the core establish the communication with SATA PHY.
trn_clk	In	Clock which is synchronized with trn_XXX signal for interface with the Host. There is no global clock buffer inside SATA IP core for this signal, so external global clock buffer should be inserted. This clock frequency is required to be higher than core_clk frequency.
core_clk	In	IP Core operating frequency output (37.50MHz for SATA-I, 75.00MHz for SATA-II). This clock is generated from SATA PHY.
dev_host_n	In	Device or Host design assignment. '0': ATA Host IP Core, '1': ATA Device IP Core (Use '1' for the device reference design)
Transmit Transaction Interface		
trn_tsof_n	In	Transmit Start-Of-Frame (SOF): Indicate start each SATA FIS packet. Active low.
trn_teof_n	In	Transmit End-Of-Frame (EOF): Indicate end each SATA FIS packet. Active low.
trn_td[31:0]	In	Transmit Data: SATA FIS packet data to be transmitted.
trn_tsrc_rdy_n	In	Transmit Source Ready: Indicates that trn_td[31:0] from the Host is valid. Active low.
trn_tdst_rdy_n	Out	Transmit Destination Ready: Indicate that the core is ready to accept data on trn_td[31:0]. Active low. - trn_tsrc_rdy_n must be de-asserted within 4 period of trn_clk after trn_tdst_rdy_n is de-asserted. So the core can accept 4 DWORD of trn_td[31:0] after trn_tdst_rdy_n is de-asserted.
trn_tsrc_dsc_n	In	Transmit Source Abort: Indicates that the Host cancel the current SATA FIS packet. May be asserted any time between SOF and EOF. Active low. - Once asserted, the Core will send out SYNC primitive to abort the current transfer
trn_tdst_dsc_n	Out	Transmit Destination Abort: Indicates that the core is aborting the current SATA FIS packet. Asserted when the physical link is going into reset. Active low.

Table1 SATA IP interface signal definition

Signal	Signal Direction	Description
Receive Transaction Interface		
trn_rsof_n	Out	Receive Start-Of-Frame (SOF): Indicate start each SATA FIS packet. Active low.
trn_reof_n	Out	Receive End-Of-Frame (EOF): Indicate end each SATA FIS packet. Active low.
trn_rd[31:0]	Out	Receive Data: SATA FIS packet data to be transmitted.
trn_rsrc_rdy_n	Out	Receive Source Ready: Indicates that the core is presenting valid data on trn_rd[31:0]. Active low.
trn_rdst_rdy_n	In	Receive Destination Ready: Indicate that the Host is ready to accept data on trn_rd[31:0]. Active low. - trn_rsrc_rdy_n will be de-asserted within 4 period of trn_clk after trn_rdst_rdy_n is de-asserted. So Host should be supported to accept 4 DWORD of trn_rd[31:0] after trn_rdst_rdy_n is de-asserted.
trn_rsrc_dsc_n	Out	Receive Source Abort: Indicates that the core cancel the current SATA FIS packet. May be asserted any time between SOF and EOF. Active low.
trn_rdst_dsc_n	In	Receive Destination Abort: Indicates that the Host cancel the current SATA FIS packet. Active low. - Once asserted, the core will send out SYNC primitive to abort the current transfer
SATA PHY Interface for Virtex5 GTP		
PHYRESET	In	SATA PHY reset. Active high. This signal is used to reset data buffer interface with SATA PHY when PHY is reset.
PHYCLK	In	Reference Clock for 16-bit SATA PHY (Virtex5 GTP) - 75MHz for SATA-I - 150MHz for SATA-II This clock is generated from DCM inside SATA PHY. It's used for both both TX and RX data by elastic buffer in GTP of SATA PHY.
TXDATA[15:0]	Out	16-bit transmit data from the core to the GTP
TXDATAK[1:0]	Out	2-bit Data/Control for the symbols of transmitted data. ("00": data byte, "01": control byte, "1X": undefined).
RXDATA[15:0]	In	16-bit receive data from the GTP to the core.
RXDATAK[1:0]	In	2-bit Data/Control for the symbols of received data. ("00": data byte, "01": control byte, "1X": undefined)
RXDATAVALID	In	Indicates symbol lock and valid data on RXDATA and RXDATAK.
LINKUP	In	Indicates that SATA link communication is established. Active high.
PLLLOCK	In	Indicates that DCM of GTP is locked. Active high.

Table1 SATA IP interface signal definition (Cont'd)

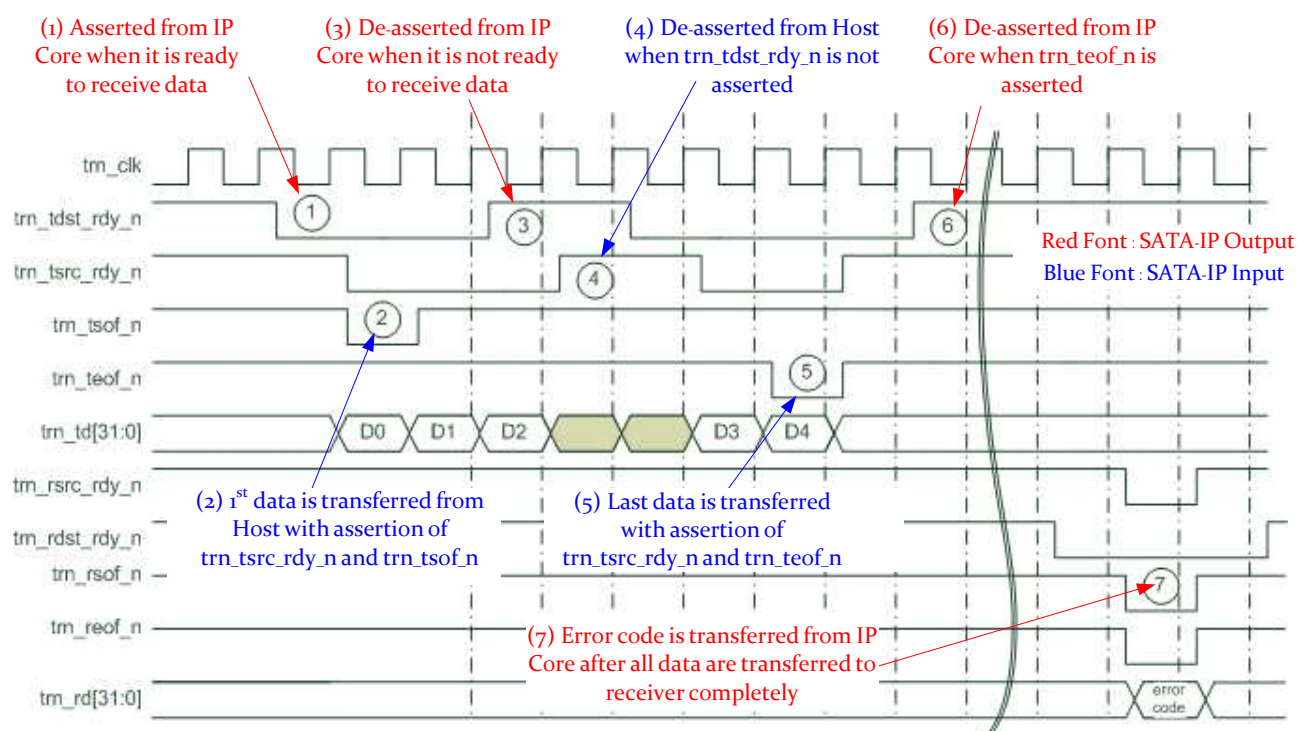


Figure5 Waveform of data transmit transaction

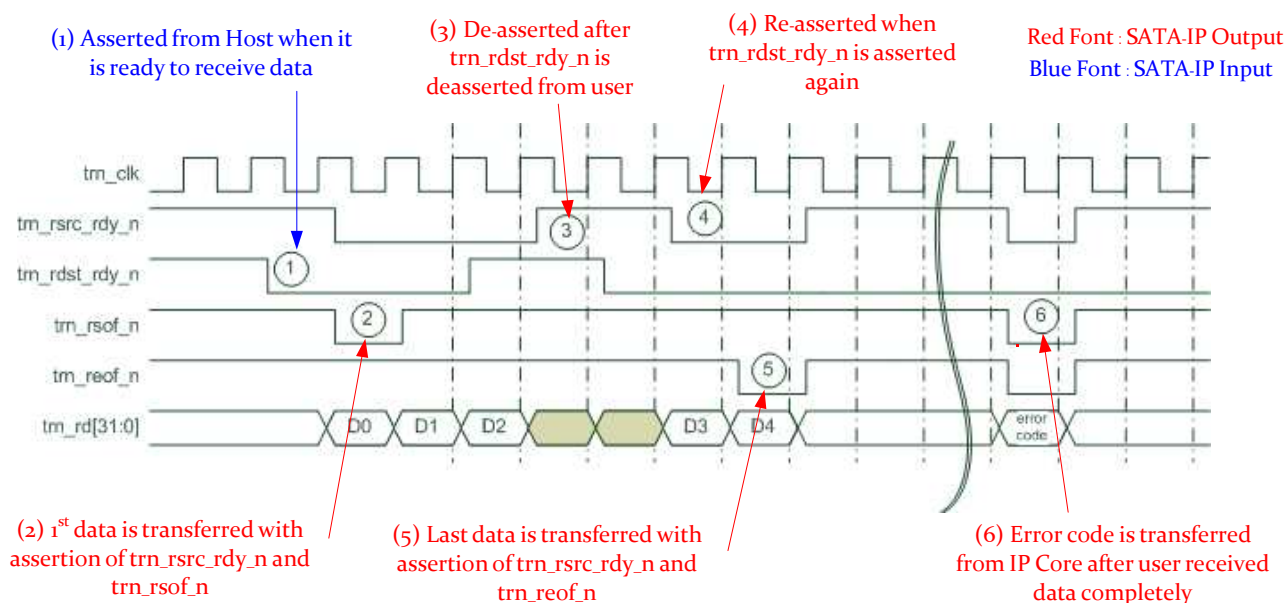


Figure6 Waveform of data receive transaction

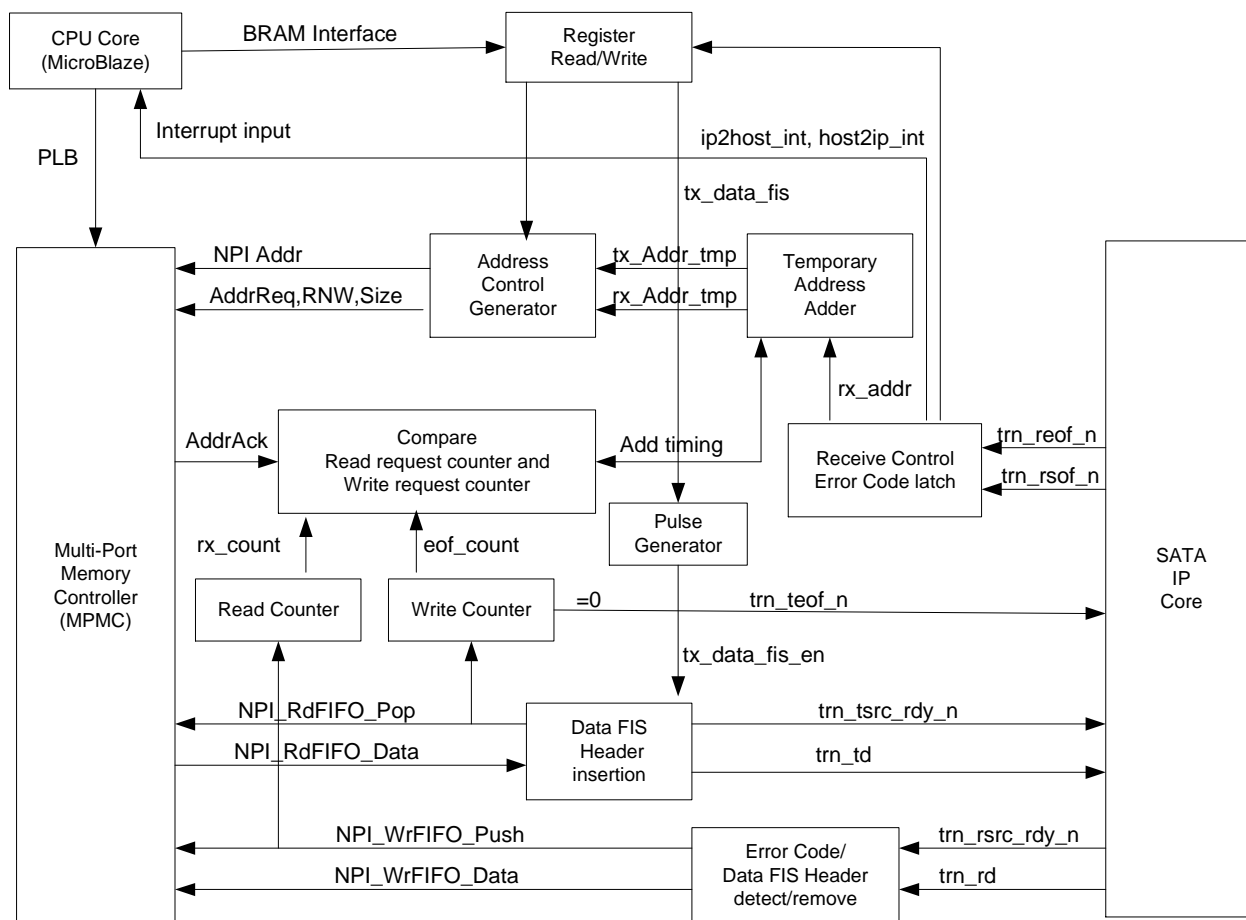


Figure7 Block diagram of connection logic

Address	Register Name	Description (Bit order is little endian)
Rd/Wr	(Label in the "sata_host.c")	
BA+0x00	Status Reg. (STATUS)	[8]MPMC Ready [7:4]OOB status data [3]GTP PLL lock status [1]Gen2 Link OK [0]SATA IP Link OK
BA+0x04	Error Code Reg. (ERROR_CODE)	SATA IP Error code. Set at transmit/receive completion. User can detect CRC or FIS error.
BA+0x08	Interrupt Clear Reg. (INT_CLEAR)	Clear interrupt by read operation at this register
BA+0x0C	Receive Word Count Reg. (RX_COUNT)	Received word count, the sum up of all FIS data count until clear.
BA+0x00	Transmit Data Address Reg. (TX_ADDR)	Set top address of transmit data storage area.
BA+0x04	Receive Data Address1 Reg. (RX_ADDR)	Set top address of receive data except Data FIS. Top address[7:0] needs to be equal to 0.
BA+0x08	Control Reg. (CONTROL)	[31]SATA Reset [30]Transmit Request [29]Send Data FIS [15:0]Transmit word count. Write operation will reset RX_COUNT register
BA+0x0C	Receive Data Address2 Reg. (RX2_ADDR)	Set top address of receive Data FIS.

(BA : Base Address)

Table2 Register mapping from CPU side

4. Software description

- SATA Device operation

Basically, SATA peripheral Device must support all mandatory commands sent from the Host. But this reference design supports only the minimum command to simplify software so that user can easily understand fundamental SATA Device software operation.

Like Host reference design, communication between the Host and the Device via SATA is done by FIS (Frame Information Structure) data structure. MicroBlaze in the Device design will build FIS data structure on its main memory space, and will send it to the Host by DMA controller that operates bus master. And FIS data sent from the Host is also transferred on the main memory by DMA controller.

MicroBlaze in the SATA Device will operates as below sequence.

- (1) After boot-up, send RegD2H FIS to the Host.
- (2) Wait command receive.
- (3) Execute received command operation.
- (4) Send FIS Data
- (5) Additional FIS data transmit/receive if necessary.

- Software of reference design description

Software source code of this reference design is stored in "sata_device.c". As a minimum implementation, this source code has a following limitation.

- There is no optional function support such as S.M.A.R.T (Self-monitoring, Analysis and Reporting Technology)
- 48bit LBA is not supported
- Cache is not supported (Not need because this peripheral is RAMDISK)
- Ultra DMA mode support is Mode 5 or slower.

After Link is established, software sends RegD2H (Device to Host FIS), and then process command sent from the Host. This design prepares original IDENTIFY DEVICE data to limit operation from the Host. Because default DDR memory capacity of ML506 is 256MBytes, this software declares 128MBytes capacity.

To make easy implementation, maximum value of SET MULTIPLE MODE is set to 1 so that READ/WRITE SECTOR command sequence and READ/WRITE MULTIPLE command sequence is identical. In this case, performance is not good. But in practically, there is no problem for typical use because the Host will use UDMA mode rather than legacy PIO mode.

There are two READ/WRITE address modes, i.e. CHS (Cylinder/Head/Sector) mode and LBA (Logical Block Address) mode. Generally, CHS mode is used in the small size HDD access. For this reference design, CHS is converted to be LBA mode access.

The commands implemented in this reference design are READ SECTOR, READ MULTIPLE, WRITE SECTOR, WRITE MULTIPLE, READ DMA, and WRITE DMA. For the other mandatory command, this reference design will do nothing and simply return RegD2H without error.

For other command which is not mandatory command, this reference design will return RegD2H with error.

This reference design implements the minimum command required for simple board operation check, but user should support all the mandatory command defined in the ATA Standard specification in their application.

5. ML506 real board operation

- OS Boot-up

After WindowsXP boot-up completion, user can recognize that WindowsXP can detect ML506 as a SATA Peripheral Device and transfer mode is set to Ultra DMA Mode 5, as shown in Figure8. Some mandatory commands for OS Boot-up are shown in Table3. The command sequence depends on PC, Device, and OS.

Command code	Command operation
ECh	IDENTIFY DEVICE
E3h	IDLE
B0h	SMART
EFh	SET FEATURES (Set transfer mode)
C8h	READ DMA
C6h	SET MULTIPLE MODE
E1h	IDLE IMMEDIATE

Table3 Command sample for WindowsXP Boot-up

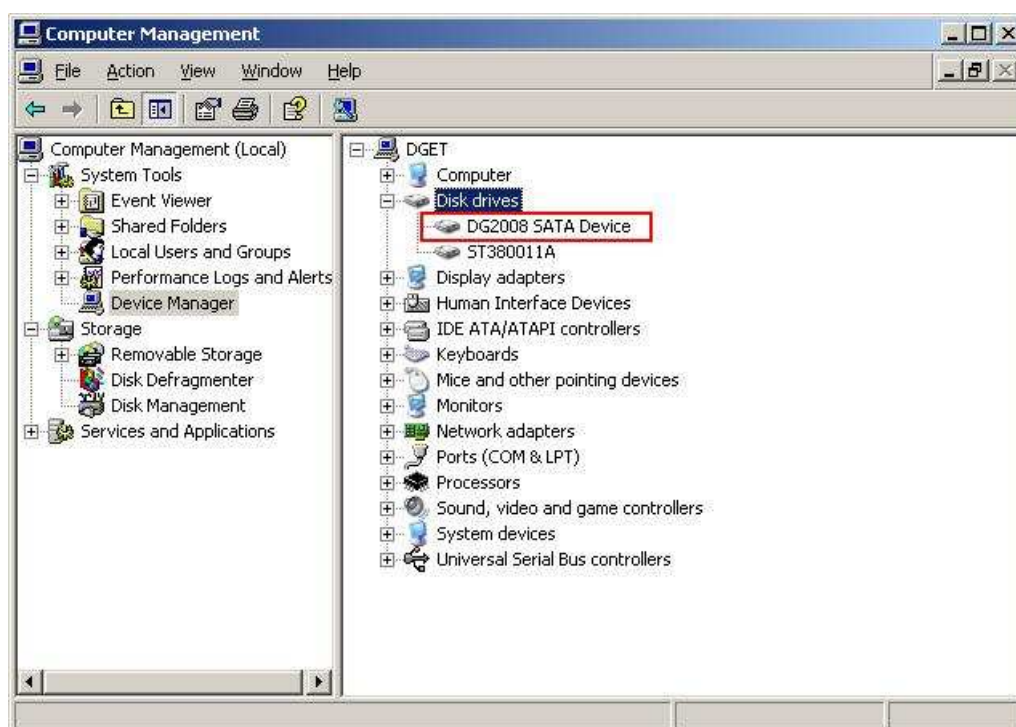


Figure8 WindowsXP detect ML506 as SATA Device

Transfer mode settings will be sent after IDENTIFY DEVICE command. Though this reference design does not support optional S.M.A.R.T command, OS boot-up is successful because this design will send error response of command not support by RegD2H. This design simply returns RegD2H without error when receive IDLE, or IDLE IMMEDIATE command.

Because this design reports maximum value of SET MULTIPLE MODE as 1, Host sets 1 at SET MULTIPLE MODE command.

After WindowsXP boot-up completion, user can recognize that WindowsXP can detect ML506 as a SATA Peripheral Device and transfer mode is set to Ultra DMA Mode 5.

- OS Format operation

Table4 shows a command sample for Format operation. Test results are shown in Figure9.

Command code	Command operation
C8h	READ DMA
40h	READ VERIFY
CAh	WRITE DMA

Table4 Command sample for Format

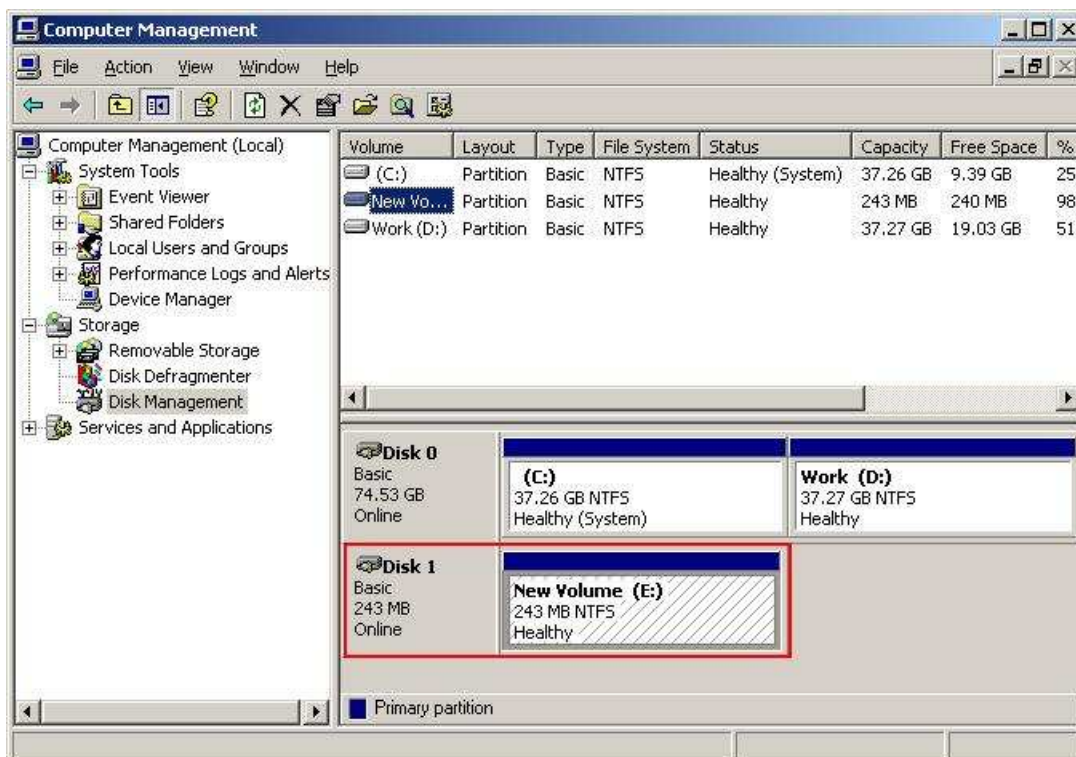


Figure9 New drive is displayed after Format

Before format real media, WindowsXP sends READ VERIFY command. To operate this command, SATA Device can select to read real data or not because there is no data transfer to the Host for READ VERIFY command. So, this reference design simply returns RegD2H without error for this command.

Table5 shows command sample for WindowsXP shut down. WindowsXP executes remained write data, and then execute FLUSH CACHE and STANDBY IMMEDIATE command.

Command code	Command operation
CAh	WRITE DMA
E7h	FLUSH CACHE
E0h	STANDBY IMMEDIATE

Table5 Command sample for WindowsXP shut down

- Performance result

Figure10 shows test result comparison between ML506 and General SATA HDD by benchmark. Compared with HDD, this reference design on ML506 can achieve more than 3 times better performance.

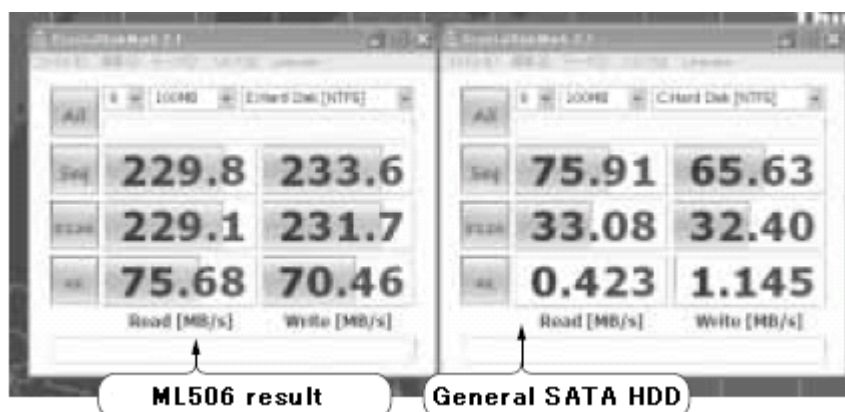


Figure10 Performance result of read/write data transfer

6. Revision History

Revision	Date	Description
0.1	06-Oct-08	Initial draft
1.0	20-Nov-08	Add introduction and test result
1.1	12-Dec-08	Add dev_host_n signal description
1.2	02-Jun-09	Add trn_clk signal description

Copyright: 2008 Design Gateway Co,Ltd.