SATA AHCI IP by Baremetal reference design manual

Rev1.0   30-Jun-23

SATA AHCI IP by Baremetal reference design manual

Rev1.0 30-Jun-23

1 Overview

This document describes AHCI-IP demo by using baremetal OS firmware running on CPU. It is recommended to read more details of AHCI-IP demo from "SATA AHCI-IP reference design manual" firstly to understand IP operation. The document can download from following website.
http://www.dgway.com/products/IP/SATA-IP/dg_sataahciip_refdesign_en.pdf

Comparing to AHCI-IP reference design which run with Linux OS, this demo runs without OS (baremetal OS). So, overhead time for processing each command will be reduced and write/read performance output will be better. Main menu of firmware is almost same design as SATA host reference design, but ATA command is different. "SATA host reference design manual" can download from following website.
http://www.dgway.com/products/IP/SATA-IP/dg_sata_ip_refdesign_host_7series_en.pdf

While SATA host reference design uses WRITE/READ DMA EXT command, AHCI-IP can use NCQ command (WRITE/READ FPDMA QUEUED) which can send up to 32 commands at the same time. By using command queue, the overhead time to fetch next command will be reduced. SATA device can select next command from the queue in the cache without waiting the host.

2 Hardware

The hardware within AHCI-IP reference design run Linux OS and baremetal OS demo is similar, as shown in Figure 2-1.
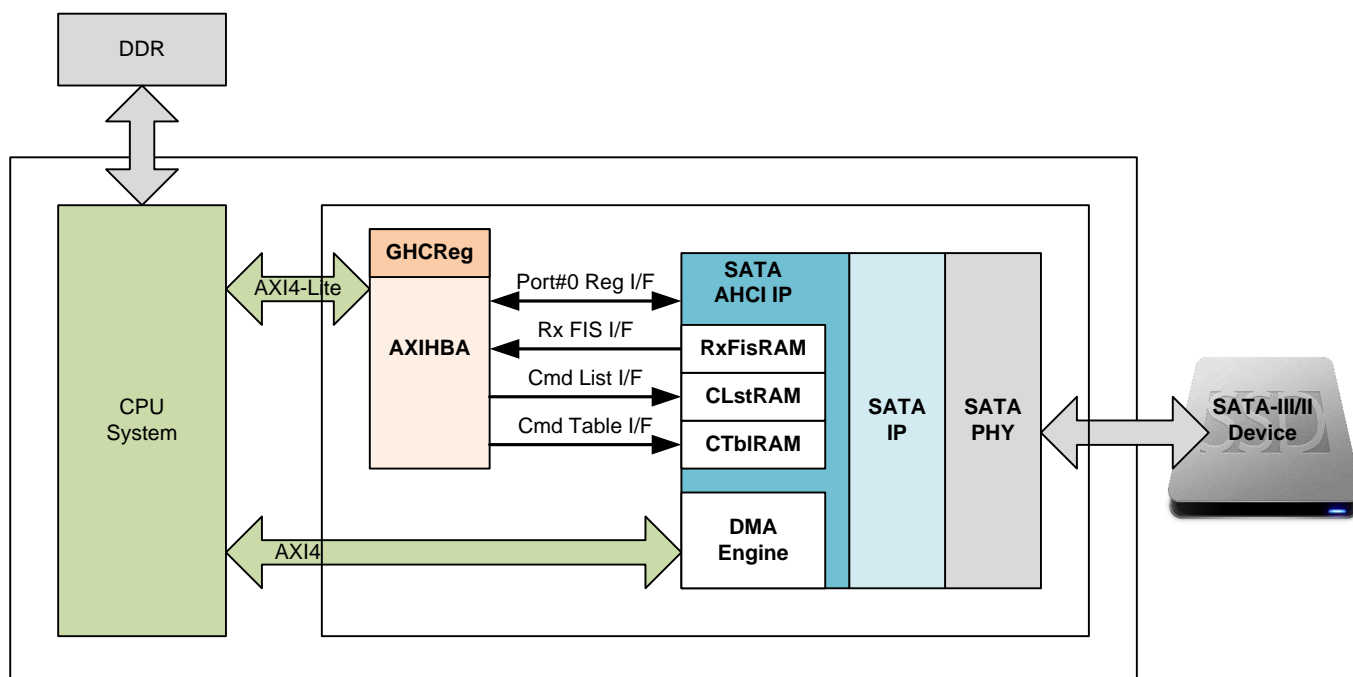

Figure 2-1 SATA AHCI IP Block diagram

CPU System exports two AXI4 buses to connect with AHCI system, i.e. AXI4-Lite for register/memory access and AXI4 for data transfer with DDR memory. Port#0 register and three RAM interface of AHCI-IP are mapped to AXI4-Lite bus by using AXIHBA module to decode address and convert AXI4-Lite bus interface to be RAM I/F. GHC register is also designed within AXIHBA module. Command Table RAM inside the IP has 64-kByte size, so address range of AXIHBA is 128 kByte. Memory map of this design is shown as below.
Note: AXIHBA module is provided in HDL code, so user can modify address range of each part in HDL.

Address	Description
BA+0x00000 –0x000FF	GHC Register
BA+0x00100 –0x0017F	Port#0 Register
BA+0x00180 –0x07FFF	Reserved
BA+0x08000 –0x080FF	Received FIS RAM (RxFisRAM)
BA+0x08100 – 0x08FFF	Reserved
BA+0x09000 –0x093FF	Command List RAM (CLstRAM)
BA+0x09400 – 0x0FFFF	Reserved
BA+0x10000 –0x1FFFF	Command Table RAM (CTblRAM)

Table 2-1 Memory map of AXIHBA module

The details of GHC and Port#0 register are described in topic 3 of "Serial SATA AHCI 1.3.1 Specification" document which can download from following website.
http://www.intel.com/content/dam/www/public/us/en/documents/technical-specifications/serial-ata-ahci-spec-rev1_3.pdf

2.1 GHC Register

GHC register is designed within AXIHBA module. Some read registers are defined to be constant value such as CAP, PI, VS, EM_LOC. Some registers are designed to be write/read register without any internal hardware operation such as CCC_CTRL, CCC_PORTS. Only two registers are effect to AHCI-IP operation, as shown in Table 2-2.

Address	Bit		Name	Description
0x04–0x07			GHC	Global HBA Control
	[0]	RW	HR	HBA Reset: Set to '1' to reset AHCI-IP. This bit will be cleared by hardware.
	[1]	RW	IE	Interrupt Enable: Set to '1' to enable global interrupt from hardware. If set '0', interrupt from all ports are disabled.
0x08–0x0B			IS	Interrupt Status Register of 32 device ports. Only bit[0] is used to support SATA device at Port#0.
	[0]	RWC	IPS0	Interrupt Pending Status: This bit is set by hardware to show that there is interrupt pending from Port#0. After software completes interrupt routine, write '1' to this bit to clear interrupt flag.

Table 2-2 GHC Register Description

Note: Before set '1' to clear IPS0 flag, user needs to clear Port0 Interrupt status (P0IS) firstly. IPS0 is OR condition of all bits in P0IS register.

2.2 Port0 Register

Typically, AHCI standard can connect with many devices. In reference design, only one SATA device is connected to Port0. To simplify software operation, only six Port0 registers are used and interrupt is asserted from Set Device Bits FIS and error condition only. Set Device Bits FIS will generate interrupt after end of each command. More details of Port0 register using in baremetal OS firmware is follows.

Address	Bit		Name	Description
0x10–0x13			P0IS	Port0 Interrupt Status
	[3]	RWC	SDBS	Set Device Bits Interrupt: A Set Device Bits FIS has been received with the 'I' bit set.
	[4]	RWC	UFS	Unknown FIS Interrupt: Unknown FIS has been received.
	[6]	RO	PCS	Port Connect Change Status: Set when connect status is changed.
	[22]	RO	PRCS	PhyRdy Change Status: Set when internal PhyRdy signal changed state.
	[24]	RWC	OFS	Overflow Status: Set when hardware receives more bytes from a device than specified in the PRD table of the command.
	[30]	RWC	TFES	Task File Error Status: Set when error bit in status register is set.
0x14–0x17	[31:0]	RW	P0IE	Port0 Interrupt Enable. Set '1' to enable interrupt to system software, and '0' to disable interrupt. The description of bit in this register is same as P0IS register. In reference design, this register is set to 4140_0058h to enable six interrupts.
0x18-0x1B			P0CMD	Port0 Command and Status
	[0]	RW	ST	Start: Set '1' to start hardware to process the command list. Before set this bit = '1', P0CMD.FRE need to set to '1'.
	[4]	RW	FRE	FIS Receive Enable: Set '1' to enable hardware to post received FISes to RxFIS RAM.
0x30-0x33	[31:0]		P0SERR	Port0 Serial ATA Error: Check this register for error condition when error interrupt is asserted.
0x34-0x37	[31:0]	RW	P0SACT	Port0 Serial ATA Active: Each bit corresponds to command slot. Set P0SACT[slot] to '1' before writing P0CI[slot] to '1'. This bit will be cleared by hardware after completes command operation in that slot.
0x38-0x3B	[31:0]	RW	P0CI	Port0 Command Issue: Each bit corresponds to command slot. Set bit to '1' to issue new command to hardware. This bit will be cleared by hardware after command packet is transferred to SATA device.

Table 2-3 Port0 Register Description

2.3 DDR memory

DDR in baremetal OS is used to store test pattern data to write and read command only because Received FIS, Command List, and Command Table use RAM within AHCI-IP. Two 32 MB areas are mapped to be test pattern for write and read separately, as shown in Figure 2-2. On ZC706 board, the first 256 kByte address area of DDR (0x0000_0000) is shared with On-Chip memory (OCM), so start address of TX_DATA_ADDR is mapped to the 0x2000_0000 instead.

```
DDR_BASE_ADDR ┌─────────────────┐
              │/////////////////│
              │/////////////////│
 +0x2000_0000 ├─────────────────┤
              │  TX_DATA_ADDR   │
              │     (32 MB)     │
 +0x2200_0000 ├─────────────────┤
              │  RX_DATA_ADDR   │
              │     (32 MB)     │
              ├─────────────────┤
              │/////////////////│
              │/////////////////│
              └─────────────────┘
```
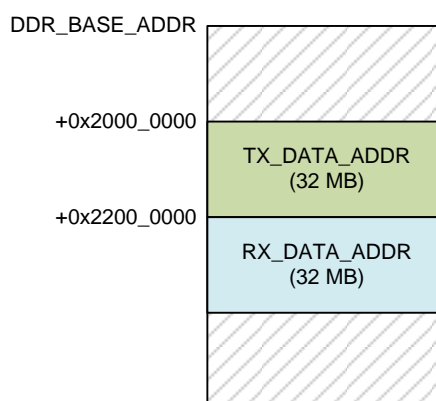
Figure 2-2 DDR memory map

Data base address (DBS) to set in PRD table of each command slot will map to the same 32 MB area. Write command will use TX_DATA_ADDR while Read command will use RX_DATA_ADDR area. In real system, user can modify DBS of each command to be different area for command slot#0-31.

3 Software

The software sequence in baremetal OS which will be described as below is different from the example sequence described in SATA AHCI reference design manual. Register access and interrupt sequence are optimized for NCQ command in this design.

Three ATA commands are implemented in firmware, i.e. IDENTIFY DEVICE (ECh), WRITE FPDMA QUEUED (61h), and READ FPDMA QUEUED (60h). Up to 32 commands can be sent to the device at the same time and one command can support up to 32 MB data transfer. 8 PRDs will be used because 1 PRD can map to 4 MB data.

The details of the software sequence are follows.

3.1 INITIALIZATION

1) Clear 256-byte Received FIS memory for monitoring D2H Register FIS from SATA device.
2) Reset hardware system by setting GHC.HR='1'.
3) Enable to receive FIS by setting P0CMD.FRE='1'.
4) Enable to process command list by setting P0CMD.ST='1'.
5) Polling D2H Register FIS area until FIS type value = 34h
   If D2H Register FIS is not found within 4 sec, it will go to step 2) to reset hardware again.
6) Enable interrupt by setting GHC.IE='1'.
7) Enable Port0 interrupt by setting P0IE=41400058h (enable Set Device Bits interrupt and five error interrupts).

3.2 IDENTIFY DEVICE

This command is used to check disk information such as disk capacity, disk model number. Since this command is not command queue, all command slots must be available before running this command.
Note: Based on AHCI specification, non-command queued and command-queued cannot run with the same queue.

Command slot#0 is used and Identify device data is returned to RX_DATA_ADDR memory area. The sequence for IDENTIFY DEVICE command is described as follows.

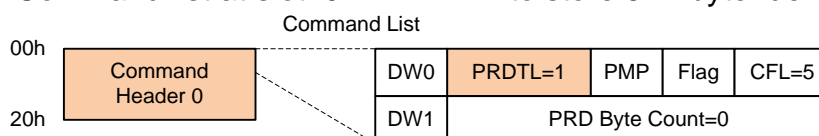1) Set value to Command list at slot#0. PRDTL=1 to store 512-byte Identify device data.

Figure 3-1 Set Command List for Identify Device Command

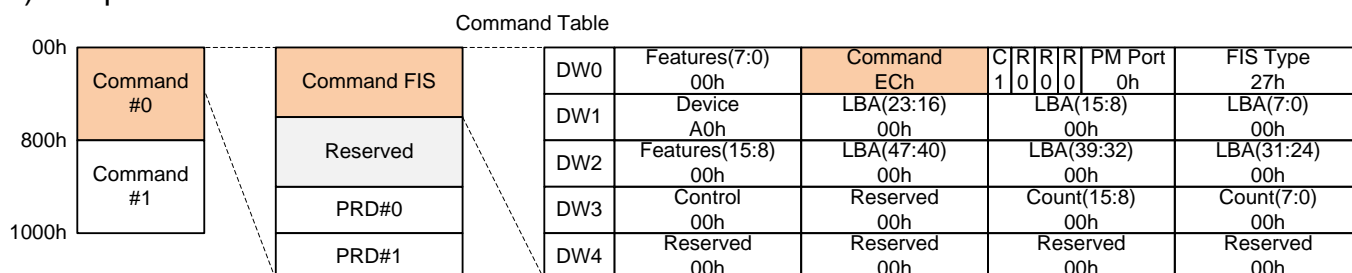2) Prepare 5-Dword Command FIS in Slot#0 of Command Table RAM.

Figure 3-2 Set Command FIS for Identify Device Command

3) Set Data Base Address (DBA) = RX_DATA_DDR to store 512-byte Identify device data, and set Byte Count=1FFh to PRD#0 in Slot#0 of Command Table RAM.
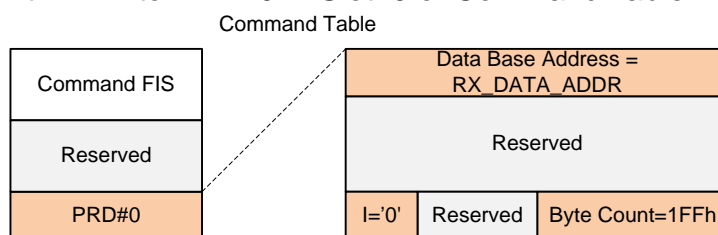

Figure 3-3 Set PRD for Identify Device Command
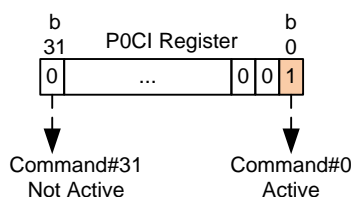
4) Set P0CI[0] ='1' to send out Command FIS in slot#0.

Figure 3-4 Set P0CI Register for Identify Device Command

5) Polling P0CI[0]='0' to wait command process complete.
6) Read and display Identify device data from RX_DATA_ADDR to the console.

3.3 WRITE FPDMA QUEUED

The sequence for WRITE FPDMA QUEUED command is follows.

1) Prepare test data to TX_DATA_ADDR area. In case of 32 MB, it can split into 8 parts for 4-MB PRDs. All 8 areas are contiguous.
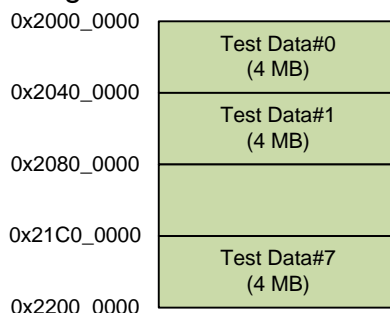

Figure 3-5 Prepare Test data pattern to TX_DATA_ADDR

2) Initialize current slot number = 0.
3) If interrupt from Set Device Bits FIS is found, read P0SACT register and latch value to update the available slot. If all slots are available (end of all commands) and remaining transfer length is 0, command process will complete.
4) If remaining transfer length is not 0 and current slot is available, send new command to current slot by following steps.
   - Calculate the value of PRDTL from remaining transfer length. One PRD can map to 4 MB, so PRDTL value will be equal to 1 – 8 depending on remaining transfer length.
     If remaining size is more than or equal to 32 MB, PRDTL = 8.
     If remaining size is less than 32 MB, PRDTL = roundup (remaining size/4 MB).

Command List

	PRDTL 0001h – 0008h	PM Port 0h	Flag 0h	P 0	W 1	A 0	CFL 00101
DW0	PRDTL 0001h – 0008h	PM Port 0h	Flag 0h	0	1	0	CFL 00101
DW1	PRD Byte Count 0000_0000h						

Figure 3-6 Set Command List for Write FPDMA Queued Command

   - Prepare 5-Dword Command FIS to current slot of Command Table RAM. Then, update LBA address for next transfer.

Command Table

Command FIS	DW0	Features(7:0) SectorCount(7:0)	Command 61h	C 1	R 0	R 0	R 0	PM Port 0h	FIS Type 27h
	DW1	Device 40h	LBA(23:16)		LBA(15:8)				LBA(7:0)
Reserved	DW2	Features(15:8) SectorCount(15:8)	LBA(47:40)		LBA(39:32)				LBA(31:24)
PRD#0	DW3	Control 00h	Reserved 00h		Count(15:8) Reserved			Count(7:3) Tag(4:0)	RSV
PRD#1	DW4	Reserved 00h	Reserved 00h		Reserved 00h			Reserved 00h	
PRD#7									

Figure 3-7 Set Command FIS for Write FPDMA Queued Command

- Set DBA value in PRD#0-7 = TX_DATA_ADDR + (PRD no. x 4 MB).
If remaining size is more than or equal to 4 MB, set Byte count = (4 MB – 1).
If remaining size is less than 4 MB, set Byte count = (remaining transfer byte – 1).
Update remaining size after filling Byte count value to each PRD.
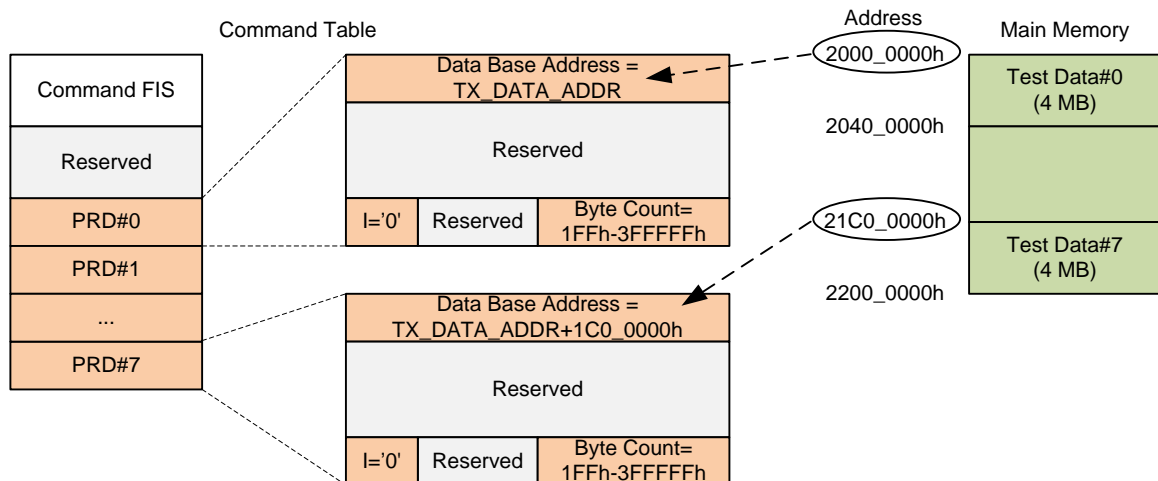

Figure 3-8 Set PRD for Write FPDMA Queued Command

- Set P0SACT[CurSlot]='1' to change slot status to be unavailable.
- Set P0CI[CurSlot]='1' to send out Command FIS to SATA device.
- Increase current slot number. Reset to 0 if current slot number is 31.

5) Loopback to Step 3) to check new Interrupt.

3.4   READ FPDMA QUEUED

The sequence of READ FPDMA QUEUED is same as WRITE FPDMA QUEUED, but this reference design uses different memory address to store received data.

1) Follow step 2) – 4) of Write FPDMA Queued until read process complete, but some setting value is different as follows.
   - W flag set in Command List = '0'.

Command List

	PRDTL 0001h – 0008h	PM Port 0h	Flag 0h	P 0	W 0	A 0	CFL 00101
DW0							
DW1	PRD Byte Count 0000_0000h						

Figure 3-9 Set Command List for Read FPDMA Queued Command

   - Command value in Command FIS = "60h".

Command Table

	Features(7:0) SectorCount(7:0)	Command 60h	C 1	R 0	R 0	R 0	PM Port 0h	FIS Type 27h
DW0								
DW1	Device 40h	LBA(23:16)		LBA(15:8)			LBA(7:0)	
DW2	Features(15:8) SectorCount(15:8)	LBA(47:40)		LBA(39:32)			LBA(31:24)	
DW3	Control 00h	Reserved 00h		Count(15:8) Reserved			Count(7:3) Tag(4:0)	RSV
DW4	Reserved 00h	Reserved 00h		Reserved 00h			Reserved 00h	

Figure 3-10 Set Command FIS for Read FPDMA Queued Command

   - DBA value in PRD0 – 7 is set to RX_DATA_ADDR + (PRD no. x 4 MB).
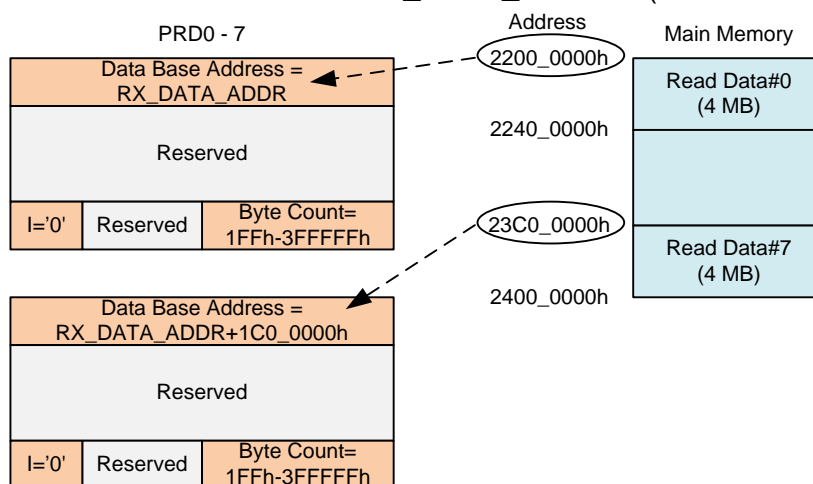
2) Verify read data from RX_DATA_ADDR area if total transfer length is not more than 32 MB.

4   Revision History

Revision	Date	Description
1.0	9-Mar-16	Initial Release

Copyright: 2016 Design Gateway Co,Ltd.