dg\_sata\_ip\_host\_demo\_instruction\_7series\_en.doc <u>SATA-IP Host Demo Instruction on 7-Series and KCU105</u> <u>Rev2.0\_23-Aug-23</u>

1	En۱	vironment	2
2	Eva	aluation procedure	6
		in Menu <sup>'</sup>	
	3.1	SATA RESET	11
	3.2	IDENTIFY DEVICE	11
	3.3	WRITE DMA (EXT)	12
	3.4	READ DMA (ÈXT)	14
	3.6	TEST READ & WRITE BACK	19
4	Rev	vision History	20



# SATA-IP Host Demo Instruction on 7-Series and KCU105

Rev2.0 23-Aug-23

This document describes SATA-IP Host evaluation procedure using SATA-IP Host reference design bit-file on AC701, KC705, ZC706, VC707, VC709, and KCU105 board. The design on VC709 and KCU105 can support fixed SATA-III device only while other boards can run for both SATA-II and SATA-III devices.

## 1 Environment

For real board evaluation of Host reference design, environment setup is shown in Figure 1-1 - Figure 1-6.

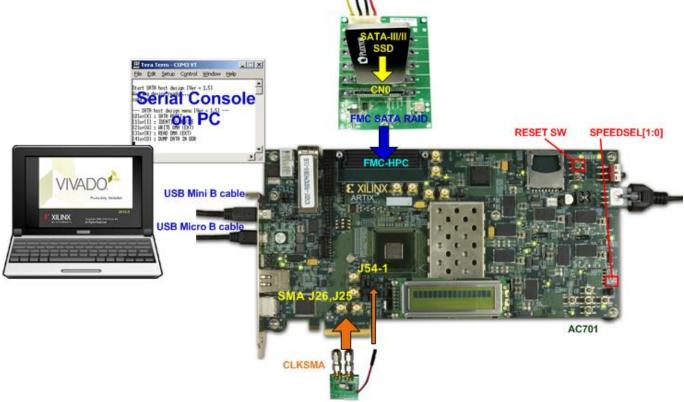
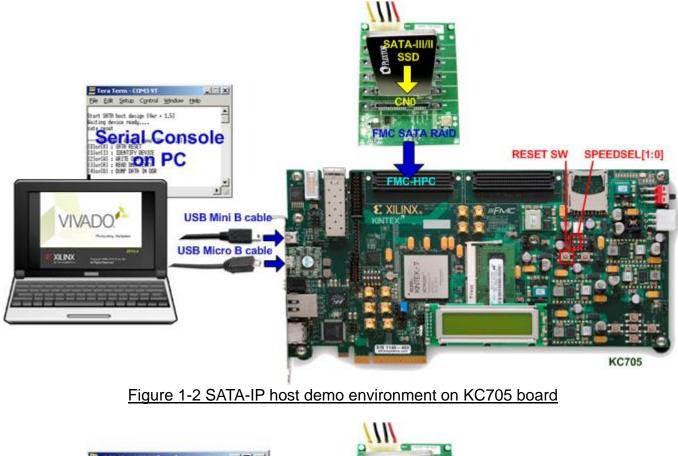


Figure 1-1 SATA-IP host demo environment on AC701 board





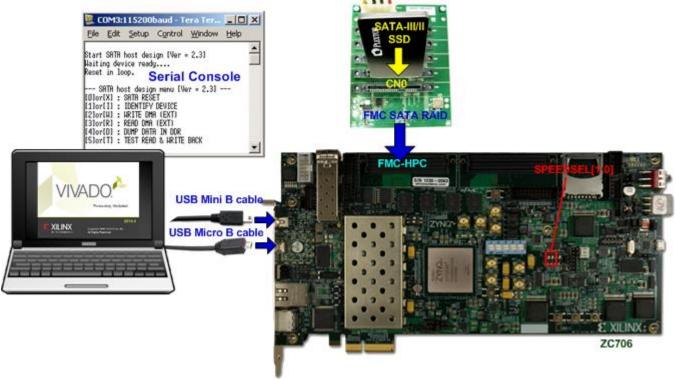


Figure 1-3 SATA-IP host demo environment on ZC706 board



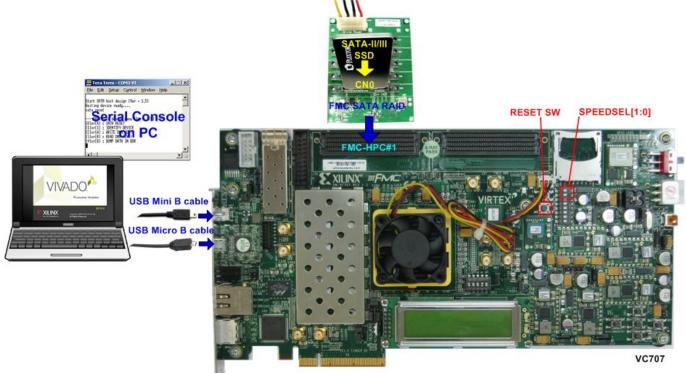
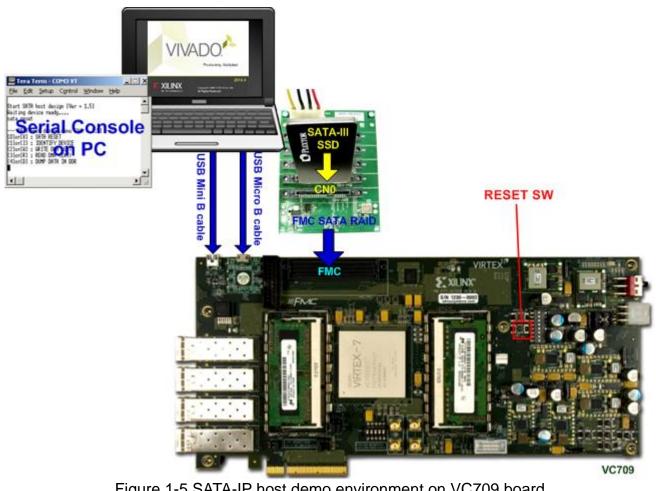


Figure 1-4 SATA-IP host demo environment on VC707 board





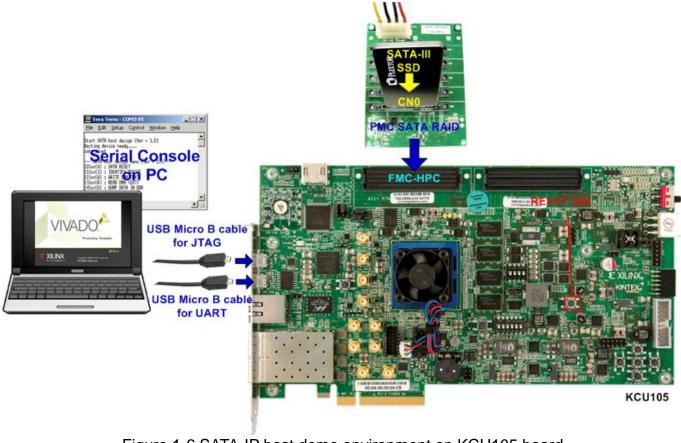
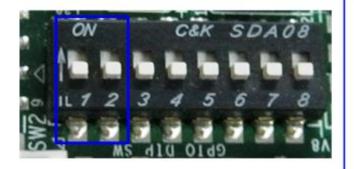


Figure 1-6 SATA-IP host demo environment on KCU105 board



## 2 Evaluation procedure

- Check all system is power off.
- Connect FMC SATA RAID board to FMC-HPC(#1) connector. <u>Note</u>: FMC SATA RAID board is provided by Design Gateway.
- Connect power to power connector on FMC SATA RAID board.
- Connect SATA3/SATA2 Device to CN0 on FMC SATA RAID board.
- For AC701, KC705, ZC706, and VC707 board which support both SATA-II and SATA-III device, set DIPSW bit [2:1] at GPIO DIPSW position (SW2@AC701, SW4@KC705, SW12@ZC706, and SW2@VC707) to select SATA speed mode. The description is described in Table 2-1.





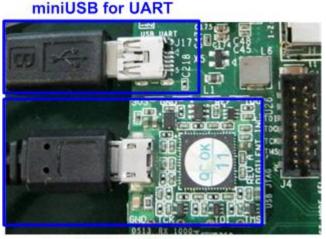
VC707

## AC701/KC705/ZC706

Figure 2-1 DIPSW setting for selecting auto-negotiation mode

DIPSW[2]	DIPSW[1]	Description	
'1'	'1'	Fixed-speed at SATA3 (6.0 Gbps)	
'1'	'0'	Fixed-speed at SATA2 (3.0 Gbps)	
'0'	'X'	Auto-speed negotiation mode	
Table 2-1 DIPSW setting to select SATA speed			

- Connect USB mini/micro B cable from USB UART port on FPGA board to PC for Serial Console. KCU105 uses micro B cable while others use mini B cable.
- Connect USB micro B cable from Digilent on FPGA board to PC for JTAG programming.



microUSB for JTAG Figure 2-2 mini and micro USB cable connection for AC701/KC705/ZC706/VC707/VC709





microUSB for JTAG Figure 2-3 two micro USB cable connections on KCU105

- Connect Power cable to FPGA board and then power up.
- Open serial monitoring software such as HyperTerminal. Terminal settings should be (Baud Rate=115,200 Data=8 bit Non-Parity Stop=1).
- Download bit-file to FPGA board
  - For ZC706 board, please follow below steps.
  - 1) Copy "ready\_for\_download" folder to PC.
  - 2) Open "ISE Design Suite Command Prompt" and change working directory to "ready\_for\_download" folder.

🛤 ISE Design Suite Command Prompt	- D ×
C:\Xilinx\14.4\ISE_DS>d:	
D:\>cd ready_for_download	
D:\ready_for_download>	
	-

Figure 2-4 ISE Design Suite Command Prompt for ZC706 board



3) Type "zc706\_bist.bat" to start downloading configuration file and the firmware. On the console, "Download 10 ... Done" is displayed after both bit file and firmware file are loaded complete. Then, user can exit this menu and see LED status and Serial console.

🛤 ISE Design Suite Command Prompt	
D:\ready_for_download.zc706_bist.bat	
D:\ready_for_download>xmd -tcl download_bit.tcl system.bit Xilinx Microprocessor Debugger (XMD) Engine	
Xilinx Hicroprocessor bebugger (And/ Engine Xilinx EDK 14.4 Build EDK_P.49d Copyright (c) 1995-2012 Xilinx, Inc. All rights reserved. Executing user script : download_bit.tcl Terminate batch job (Y/N)? y	
D:\ready_for_download>zc706_bist.bat	
D:\ready_for_download>xmd -tcl download_bit.tcl system.bit Xilinx Microprocessor Debugger (XMD> Engine Xilinx EDK 14.4 Build EDK_P.49d	
Copyright (c) 1995-2012 Xilinx, Inc. All rights reserved. Executing user script : download_bit.tcl	
Programming Bitstream system.bit Fpga Programming Progress	
Successfully downloaded bit file.	-



🛤 ISE Design Suite Command Prompt	
CortexA9 Processor Configuration	
Version	
No of PC Breakpoints6 No of Addr/Data Watchpoints1	
Connected to "arm" target. id = 64 Starting GDB server for "arm" target (id = 64) at TCP port no 1234 Target reset successfully	
<pre>Info: Enabling level shifters and clearing fabric port resets Downloading Program sata_host.elf     section, .text: 0x0000000-0x0000629f     section, .init: 0x00062a0-0x000062b7     section, .fini: 0x000062b8-0x000062cf     section, .rodata: 0x000062d0-0x00006b07     section, .data: 0x00006b08-0x00007263     section, .eh_frame: 0x00007264-0x00007267     section, .bss: 0x00007268-0x0000730b     section, .init_array: 0x0000c000-0x0000c007     section, .init_array: 0x0000c000-0x0000c007     section, .fini_array: 0x0000c000-0x0000c007     section, .heap: 0x0000c008-0x0000c007     section, .heap: 0x0000c002-0x0000c007     section, .heap: 0x0000c002-0x0000c40f     section, .heap: 0x0000c002-0x0000c407     section, .heap: 0x0000c002-0x000002-0x000002 Processor started. Type "stop" to stop processor </pre>	
RUNNING> Disconnected from Target 64	
Disconnected from Target 352	
D:\ready_for_download>	<b>_</b>

#### Figure 2-6 End of Downloading Firmware for ZC706 board



- For other boards, bit file can be downloaded from Vivado or iMPACT tool.

Vivado 2015.4 File Edit Flow <u>T</u> ools <u>Wi</u> ndow Layon			
2) 📾 💵 🖬 🖿 🗙 🚳 🗄	Default Layout	💌 🎉 🗞 🍾 🖾 Dashboard 🕶 🔅	
Hardware Manager · localhost/xilinx_tcf/t	Digilent/210308956844		
Hardware	_ D & X		🔶 Program Device 📃
< Z ⇔ 🛃 🗣 🕨 🕨 🔳			Select a bitstream programming file and download it to your hardware
Name	Status		device. You can optionally select a debug probes file that corresponds to the debug cores contained in the bitstream programming file.
☐ localhost (1) ☐ d xiinx tcf/Digient/210308956844			Bitstream file: D:/Downloads/download.bit
xdtu040_0 (2) SysMon (System Monitor)	Programmed	Hardware Device Properties     Ctrl+E	Debug probes file:
MIG_1 (MIG)	O Not Supported	Program Device	
		🤣 Refresh Device	Enable end of startup check
		Add Configuration Memory Device Boot from Configuration Memory Device	Program
		Program BBR Key Clear BBR Key	
		Program eFUSE Registers	
e	) • E	Export to Spreadsheet	

Figure 2-7 Download bit file from Vivado tool for other boards except ZC706

 After FPGA start operation, check (PL) GPIO LEDs status on FPGA board at LED0/L-LED1/C. Both LEDs must be ON, as shown in Figure 2-8 and Figure 2-9. LED2/R status depends on SATA speed status. Each LED description is described as follows.

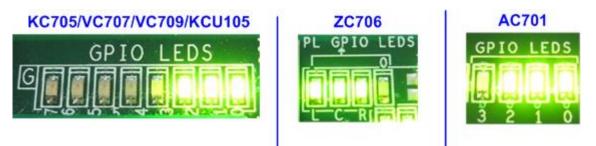


Figure 2-8 LED status after system set up complete on SATA-3 speed







Figure 2-9 LED status after system set up complete on SATA-2 speed

LED	ON	OFF
LED0/L	OK	150 MHz of SATA clock on FMC SATA RAID cannot lock. Please check
		150 MHz clock source on FMC SATA RAID board.
LED1/C	OK	SATA-IP cannot detect SATA device. Please check SATA device and the
		connection.
LED2/R	SATA-III	SATA-II (Not support on VC709 and KCU105)
LED3/0	Always OFF	
	Table	2-2 LED Status of host reference design on FPGA board



• At serial console on PC, main menu will be displayed as shown in Figure 2-10. Then, user can execute each command operation. Please check serial-cable connection if this menu is not displayed on console.

💆 COM3:115200baud - Tera 💼 💼 🞫	
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
Start SATA host design [Ver = 2.3] Haiting device ready Reset in loop.	
SATR host design menu [Ver = 2.3] [O]or[X] : SATR RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : HRITE DHA (EXT) [3]or[R] : READ DHA (EXT) [4]or[D] : DUMP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK	
	Ŧ
Figure 2-10 Main Menu of host demo	<u>2</u>



## 3 Main Menu

## 3.1 SATA RESET

Select '0' or 'X' for sending hardware reset signal to SATA-IP. Hardware reset is designed to reset both SATA-IP and SATA-PHY module. So, SATA initialize process will be started again and display "SATA RESET selected", as shown in Figure 3-1, after sending this reset.

🐸 COM3:115200baud - Tera Te 💼 💼 📻	
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
+++ SATA RESET selected +++ Reset in loop.	
SATA host design menu [Ver = 2.3] [D]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[I] : HRITE DHA (EXT) [3]or[R] : READ DHA (EXT) [4]or[D] : DUMP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK	
	Ŧ

Figure 3-1 SATA Reset Output

### 3.2 IDENTIFY DEVICE

Select '1' or 'I' for sending "IDENTIFY DEVICE" command to HDD/SSD. Disk information (Model name, 48-bit LBA Supported, disk capacity) will be displayed by using this menu, as shown in Figure 3-2.

😃 COM3:115200baud - Tera 💼 💷 🞫	
<u>File E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
+++ IDENTIFY DEVICE selected +++ Model name : Samsung SSD 840 PRO Series 48bit LBA is supported Capacity : 256GB (MAX LBA = 500118192) UDMA mode 6 supported	
SATA host design menu [Ver = 2.3] [O]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : HRITE DHA (EXT) [3]or[R] : READ DHA (EXT) [4]or[D] : DUHP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK	
Figure 3-2 Disk Information from IDENTIFY DEVICE	<u>E command</u>



## 3.3 WRITE DMA (EXT)

Select '2' or 'W' for sending "WRITE DMA (EXT)" command to HDD/SSD. Three inputs are required for this menu, i.e.

- Start LBA: this value is the start sector number of HDD/SSD to write data.
- Sector Count: this value is the total transfer size in sector unit (512 byte) for writing HDD/SSD. This size is the data size for CPU to fill to write buffer. If the input is more than 65536 (maximum size for one SATA command), only 65536 sector data is filled and the later command will use same data area with the first command.
- Write Pattern: this value is used for selecting test pattern to write to buffer and then forward to HDD/SSD. There are six test patterns in this demo, i.e. 32-bit increment [0], 32-bit decrement [1], 0000000H [2], FFFFFFFH [3], current data in read buffer [4], and LFSR counter [5].

After Software receives all inputs correctly,

- "Prepare data" will be displayed during CPU writing test pattern data to write buffer.
- "Execute Write" will be displayed during CPU sending WRITE DMA (EXT) command and transferring data from write buffer to HDD/SSD.
- Transfer speed will be displayed after write operation complete.

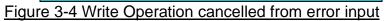
Figure 3-3 shows the example of test result when operation complete. Write operation will be cancelled from two cases, i.e. receiving error input or receiving any input from user during CPU processing this operation, as shown in Figure 3-4 and Figure 3-5 sequentially.

🐸 COM3:115200baud - Tera Term VT	X I
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
+++ HRITE DMA EXT selected +++ Enter Start LBA : 0 - 500118191 (0x10CF32AF) => 0 Enter Sector Count : 1 - 500118192 (0x10CF32B0) => 0x10000000 Hrite Pattern ? : [0]Inc32 [1]Dec32 [2]All_0 [3]All_1 [4]Rdbuf [5]LFSR => 5 Prepare Data Data ready Execute Hrite 12345678 Total = 8[GB] , Time = 18[s] , Transfer speed = 470[MB/s] SATA host design menu [Ver = 2.3]	
(D)or(X) : SATA REŠET (1)or(I) : DENTIFY DEVICE (2)or(H) : HRITE DHA (EXT) (3)or(R) : READ DHA (EXT) (4)or(D) : DUMP DATA IN DDR (5)or(T) : TEST READ & HRITE BACK	Ŧ

Figure 3-3 WRITE DMA (EXT) command input and output



<u>File Edit Setup Control Window Help</u> +++ WRITE DMA EXT selected +++ Enter Start LBA : 0 - 500118191 (0x10CF32AF) => 0 Enter Sector Count : 1 - 500118192 (0x10CF32B0) => 500118193 SATA host design menu [Ver = 2.3] [0]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : WRITE DMA (EXT) [3]or[R] : READ DMA (EXT) [3]or[R] : READ DMA (EXT) [4]or[D] : DUMP DATA IN DDR [5]or[T] : TEST READ & WRITE BACK	😃 COM3:115200baud - Tera Term VT 🛛 🗖 💷 🔤	
Enter Start LBA : 0 - 500118191 (0x1DCF32AF) => 0 Enter Sector Count : 1 - 500118192 (0x1DCF32B0) => 500118193 SATA host design nenu [Ver = 2.3] [0]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[I] : HENTIFY DEVICE [2]or[I] : HEND DHA (EXT) [3]or[R] : READ DHA (EXT) [4]or[D] : DUMP DATA IN DDR	<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
÷	Enter Start LBA : O - 500118191 (0x1DCF32AF) => O Enter Sector Count : 1 - 500118192 (0x1DCF32BO) => 500118193 SATA host design menu [Ver = 2.3] [O]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[M] : HRITE DHA (EXT) [3]or[R] : READ DHA (EXT) [4]or[D] : DUMP DATA IN DDR	4



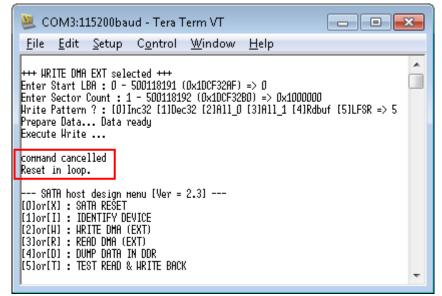


Figure 3-5 Write Operation cancelled from receiving input during operation

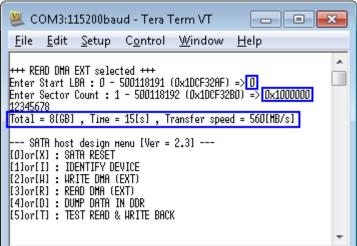


## 3.4 READ DMA (EXT)

Select '3' or 'R' for sending "READ DMA (EXT)" command to HDD/SSD. Two or three inputs are required for this menu, i.e.

- Start LBA: same description with Start LBA in WRITE DMA (EXT) menu.
- Sector Count: same description with Sector Count in WRITE DMA (EXT) menu. If this input is not more than 65536, the third input will be displayed for selecting verification pattern. If input is more than 65536, the third input will not be displayed to skip data verification process for checking performance only, as shown in Figure 3-6.
- Verify Pattern (Optional): this value is used for selecting verification pattern. This input should be matched with the pattern in WRITE DMA (EXT) menu. Six verification patterns can be selected, similar to write pattern. "Verify Data ... Success" is displayed for success case, and "Data Mismatch with failure value" is displayed for failure case, as shown in Figure 3-7.

Similar to WRITE DMA (EXT) menu, Read operation will be cancelled if receiving error input or receiving any input from user during CPU processing, as shown in Figure 3-8 and Figure 3-9 sequentially.



#### Figure 3-6 READ DMA (EXT) command without verify

COM3:115200baud - Tera Term VT	COM3:115200baud - Tera Term VT       File       Edit       Setup       Control       Window       Help
+++ READ DMA EXT selected +++ Enter Start LBA : O - 500118191 (0x10CF32AF) => <mark>O</mark> Enter Sector Count : 1 - 500118192 (0x10CF32BO) => <mark>65536</mark>	+++ READ DMA EXT selected +++ Enter Start LBA : O - 500118191 (0x1DCF32AF) => O Enter Sector Count : 1 - 500118192 (0x1DCF32BO) => 65536
Total = 32[MB] , Time = 60[ms] , Transfer speed = 555[MB/s] Verify Pattern ? : [0]Inc32 [1]Dec32 [2]All_0 [3]All_1 [4]Hrbuf [5]LFSR =>5 Verify DataStart check ISuccess	Total = 32[MB] , Time = 60[ms] , Transfer speed = 554[MB/s] Verify Pattern ? : [O]Inc32 [1]Dec32 [2]All_O [3]All_1 [4]Wrbuf [5]LFSR =>[O] Verify DataStart check
	Data Mismatch ADDR[0x83000000]=> T[0x00000000] F[0x00000001] SATA host design nenu [Ver = 2.3] [0]or[X]: SATA RESET [1]or[I]: IDENTIFY DEVICE [2]or[H]: HRITE DHA (EXT) [3]or[R]: READ DHA (EXT) [3]or[R]: READ DHA (EXT) [3]or[R]: READ DHA (EXT) [3]or[R]: READ DHA (EXT) [4]or[D]: DUMP DATA IN DDR [5]or[T]: TEST READ & HRITE BACK

#### Figure 3-7 READ DMA (EXT) with verify process



🐸 COM3:115200baud - Tera Term VT 🛛 📼 🔳	×
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	_
+++ READ DMA EXT selected +++ Enter Start LBA : O - 500118191 (0x1DCF32AF) => O Enter Sector Count : 1 - 500118192 (0x1DCF32BO) => <mark>5001181</mark> 9	3
SATA host design menu [Ver = 2.3] [D]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : HRITE DMA (EXT) [3]or[R] : READ DMA (EXT)	
[4]or[D] : DUHP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK	Ŧ

Figure 3-8 Read Operation cancelled from error input

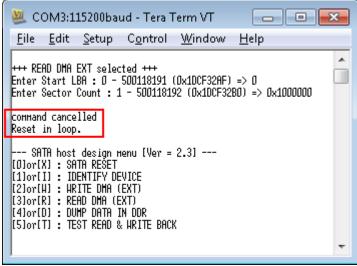


Figure 3-9 Read Operation cancelled from receiving input during operation



### 3.5 DUMP DATA IN DDR

Select '4' or 'D' to dump data from buffer to display on Serial Console. In this demo, DDR3/4 is mapped to address = 8000\_0000h - BFFF\_FFFh (for all boards except KCU105) or 8000\_0000h - FFFF\_FFFh (for KCU105). Six submenus can be selected, i.e.

- 'G': this submenu is used to select the address to read, as shown in Figure 3-10. The address can be input to be hex value by adding prefix "0x", so normally input will be received in decimal value.

🛎 COM3:115200baud - Tera Term VT 👘 💼 💼	×			
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp				
[Gloto [N]ext [P]rev [H]rbuf [R]dbuf [C]learbuf ? g				
Durp Address         ? 0x830001001         Ca           [0x830001001]         00000040         00000041         00000042         00000043           [0x830001201]         00000044         00000045         00000046         00000047           [0x830001201]         00000048         00000049         00000048         00000048           [0x830001301]         00000040         00000041         00000048         00000047           [0x830001401]         00000050         00000051         00000052         00000053           [0x830001501]         00000054         00000055         00000056         00000057           [0x830001601]         00000050         00000050         00000058         00000058         00000058           [0x830001701]         00000050         00000050         00000058         00000058         00000058           [0x830001801]         00000060         00000061         00000062         00000067         00000067           [0x830001801]         00000064         00000065         00000066         00000067         00000067           [0x830001801]         00000064         00000069         00000066         00000067         00000067           [0x830001801]         000000070         00000071         00000072				
Figure 3-10 Goto submenu example				

- 'N': this submenu is used to read next 256 byte data in buffer, as shown in Figure 3-11.
- 'P': this submenu is used to read previous 256 byte data in buffer, as shown in Figure 3-11.



🖉 COM3:115200baud - Tera Term VT 🛛 🗖 🖻 🖻				😃 сомз:11	.5200baud -	Tera Term	VT [	- 0	×
<u>F</u> ile <u>E</u> dit <u>S</u> etup	C <u>o</u> ntrol <u>W</u> ine	dow <u>H</u> elp	0	<u>F</u> ile <u>E</u> dit	<u>S</u> etup C <u>o</u>	ontrol <u>W</u> i	ndow <u>H</u> el	р	
[0x83000100] 00000040	00000041	00000042	00000043	[0x83000200]	00000080	00000081	00000082	00000083	
[0x83000110] 00000044 [0x83000120] 00000048	00000045 00000049	00000046 00000048	00000047	[0x83000210] [0x83000220]	00000084 00000088	00000085	00000086 00000088	00000087 00000088	
[0x83000130] 0000004C	00000049 00000040	0000004H 0000004E	0000004B 0000004F	[0x83000230]	00000088 0000008C	00000089 0000008D	DDDDDD8H	0000008F	
[0x83000140] 00000050	00000051	00000052	00000053	[0x83000240]	000000000	00000091	000000002	000000093	
[0x83000150] 00000054	00000055	00000056	00000057	[0x83000250]	00000094	00000095	00000096	00000097	
[0x83000160] 00000058	00000059	00000058	0000005B	[0x83000260]	00000098	00000099	0000009A	0000009B	
[0x83000170] 0000005C	00000050	0000005E	0000005F	[0x83000270]	0000009C	0000009D	0000009E	0000009F	
[0x83000180] 00000060	00000061	00000062	00000063	[0x83000280]	000000A0	000000A1	000000A2	000000A3	
[0x83000190] 00000064	00000065	00000066	00000067	[0x83000290]	000000A4	000000A5	000000A6	000000A7	
[0x830001A0] 00000068	00000069	0000006A	0000006B	[0x830002A0]	000000A8	000000A9	000000AA	000000AB	
[0x830001B0] 0000006C	0000006D	0000006E	0000006F	[0x830002B0]	000000AC	000000AD	000000AE	000000AF	
[0x830001C0] 00000070	00000071	00000072	00000073	[0x830002C0]	000000B0	000000B1	000000B2	000000B3	
[0x83000100] 00000074	00000075	00000076	00000077	[0x83000200]	000000B4	000000B5	000000B6	000000B7	
[0x830001E0] 00000078	00000079	0000007A	0000007B	[0x830002E0]	000000B8	000000B9	000000BA 000000BE	000000BB	
[[0x830001F0]] 0000007C [[G]oto [N]ext [P]rev [H	0000007D ]rbuf [R]dbuf [(	0000007E	0000007F	[[0x830002F0]] [[G]oto_[N]ext	000000BC	000000BD	[C]learbuf ?	000000BF	
	00000081	00000082	0000083	[0x83000100]	00000040	00000041	00000042	00000043	
[0x83000210] 00000084	00000085	00000086	00000087	[0x83000110]	00000044	00000045	00000046	00000047	
[0x83000220] 00000088	00000089	0000008A	0000008B	[0x83000120]	00000048	00000049	00000048	0000004B	
[0x83000230] 0000008C	0000008D	0000008E	0000008F	[0x83000130]	0000004C	0000004D	0000004E	0000004F	
[0x83000240] 00000090	00000091	00000092	00000093	[0x83000140]	00000050	00000051	00000052	00000053	
[0x83000250] 00000094	00000095	00000096	00000097	[0x83000150]	00000054	00000055	00000056	00000057	
[0x83000260] 00000098	00000099	0000009A	0000009B	[0x83000160]	00000058	00000059	0000005A	0000005B	
[0x83000270] 0000009C	00000090	0000009E	0000009F	[0x83000170]	0000005C	00000050	0000005E	0000005F	
[0x83000280] 000000A0	000000A1	000000A2	000000A3	[0x83000180]	00000060	00000061	00000062	00000063	
[0x83000290] 000000A4		000000A6	000000A7	[0x83000190]	00000064	00000065	00000066	00000067	
[0x830002A0] 000000A8	000000A9	000000AA	000000AB	[0x830001A0]	00000068	00000069	0000006A	0000006B	
[0x830002B0] 000000AC	000000AD	000000AE	000000AF	[0x830001B0]	00000060	0000006D	0000006E	0000006F	
[0x830002C0] 000000B0 [0x830002D0] 000000B4	000000B1 000000B5	000000B2 000000B6	000000B3 000000B7	[0x830001C0] [0x83000100]	00000070 00000074	00000071 00000075	00000072 00000076	00000073 00000077	
[0x830002E0] 00000084	000000B9	DDDDDDB6	000000BB	[0x830001E0]	00000074 00000078	00000075	00000076 0000007A	00000077 00000078	
[0x830002F0] 000000BC	000000BD	000000BE	000000BF	[0x830001F0]	00000078 0000007C	00000079 0000007D	0000007E	00000076 0000007F	
[G]oto [N]ext [P]rev [H			0000000	[G]oto [N]ext					-
prototo thisent traffer th	near thiged to	vilearoar r		prototo thitent			torregion i		

Figure 3-11 Read Next/Previous 256 byte data in buffer

- 'W': this submenu is used to read 256 byte data at top of write buffer, as shown in Figure 3-12.
- 'R': this submenu is used to read 256 byte data at top of read buffer, as shown in Figure 3-12.

😃 COM3:115200baud - Tera Term VT 👘 💼 💼	😃 COM3:115200baud - Tera Term VT 💦 💼 💌						
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp						
End         getap         General         Intervert           [6]oto         [N]ext         [P]rev         [H]rbuf         [R]dbuf         [C]learbuf         ? H           [0x81000001]         00000001         00000001         00000002         00000003           [0x81000010]         00000008         00000005         00000006         00000007           [0x81000020]         00000008         00000009         00000008         00000008           [0x81000040]         00000010         00000011         00000012         00000013           [0x81000050]         00000014         00000015         00000014         00000014         00000018           [0x81000060]         00000018         00000019         00000014         00000018         00000018           [0x81000080]         00000020         00000021         00000018         00000018         00000018           [0x81000080]         00000024         00000025         00000026         00000027         [0x81000080]         00000028         00000028         00000028         00000028         00000028         00000028         00000028         00000028         00000028         00000028         00000027         [0x81000080]         00000033         00000033         000000033         00000032 </td <td>Entre       Line       Line       Line       Line         [6]oto       [N]ext       [P]rev       [H]rbuf       [R]dbuf       [C]learbuf       ? r         [0x83000001)       00000001       00000002       00000003       00000003       00000003         [0x83000010]       00000008       00000005       00000006       00000007         [0x83000020]       00000002       00000009       00000008       00000008         [0x83000030]       00000010       00000011       00000012       00000013         [0x83000040]       0000010       0000011       0000012       00000013         [0x83000050]       00000014       00000015       00000016       00000017         [0x83000060]       0000012       0000011       0000011       0000011         [0x83000070]       0000012       0000011       0000011       0000011         [0x83000080]       0000012       0000011       00000011       00000012         [0x83000080]       00000024       0000025       00000026       00000027         [0x83000080]       00000022       00000026       00000027       [0x83000026]       00000028         [0x83000080]       00000022       00000026       00000027       [0x83000</td>	Entre       Line       Line       Line       Line         [6]oto       [N]ext       [P]rev       [H]rbuf       [R]dbuf       [C]learbuf       ? r         [0x83000001)       00000001       00000002       00000003       00000003       00000003         [0x83000010]       00000008       00000005       00000006       00000007         [0x83000020]       00000002       00000009       00000008       00000008         [0x83000030]       00000010       00000011       00000012       00000013         [0x83000040]       0000010       0000011       0000012       00000013         [0x83000050]       00000014       00000015       00000016       00000017         [0x83000060]       0000012       0000011       0000011       0000011         [0x83000070]       0000012       0000011       0000011       0000011         [0x83000080]       0000012       0000011       00000011       00000012         [0x83000080]       00000024       0000025       00000026       00000027         [0x83000080]       00000022       00000026       00000027       [0x83000026]       00000028         [0x83000080]       00000022       00000026       00000027       [0x83000						

Figure 3-12 Read 256 byte data at top of write/read buffer



- 'C': this submenu is used to clear data in write/read buffer to be zero value. Select 'Y' to confirm for clear write/read buffer, but user can select 'N' to not clear the current buffer.

Eile         Edit         Setup         Control         Window         Help           [Gloto         [Nlext<[P]rev         [H]rbuf         [R]dbuf         [C]lear Write         Buffer?         [VN] => []:         Clear Write         Buffer?         [C]           Clear Write         Buffer?         [VN] => []:         Clear Write         Buffer         [C]           Clear Read         Buffer?         [VN] => []:         Clear Write         Buffer         [C]           Clear Read         Buffer?         [VN] => []:         Clear Write         Buffer         [C]           Clear Write         SATA RESET         [I]	🐸 COM3:115200baud - Tera Term VT 👘 💼	×
Clear Hrite Buffer ? [Y/N] => y : Clear Hrite Buffer Clear Read Buffer ? [Y/N] => y : Clear Read Buffer SRTR host design menu [Ver = 2.3] [0]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : HRITE DMA (EXT) [3]or[R] : READ OHA (EXT) [3]or[R] : READ OHA (EXT) [4]or[D] : DUMP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK [0x83000000] 0000000 00000000 00000000 000000	<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
[3]or(R]: READ DMA (EXT)         [4]or(D]: DUHP DATA IN DDR         [5]or(T]: TEST READ & HRITE BACK         [0:883000000]       00000000       00000000       00000000         [0:883000010]       00000000       00000000       00000000       00000000         [0:883000020]       00000000       00000000       00000000       00000000         [0:883000020]       00000000       00000000       00000000       00000000         [0:883000020]       00000000       00000000       00000000       00000000         [0:883000050]       00000000       00000000       00000000       00000000         [0:883000060]       00000000       00000000       00000000       00000000         [0:883000060]       00000000       00000000       00000000       00000000         [0:883000080]       00000000       00000000       00000000       00000000         [0:883000080]       00000000       00000000       00000000       00000000         [0:883000080]       00000000       00000000       00000000       00000000         [0:883000080]       00000000       00000000       00000000       00000000         [0:883000080]       00000000       00000000       000000000       00000000 <td>Clear Hrite Buffer ? [V/N] =&gt;y: Clear Hrite Buffer Clear Read Buffer ? [V/N] =&gt;y: Clear Read Buffer  SATA host design непи [Ver = 2.3] [D]or[X] : SATA RESET</td> <td>^</td>	Clear Hrite Buffer ? [V/N] =>y: Clear Hrite Buffer Clear Read Buffer ? [V/N] =>y: Clear Read Buffer SATA host design непи [Ver = 2.3] [D]or[X] : SATA RESET	^
[0x83000010]         0000000         0000000         0000000         0000000           [0x83000020]         0000000         0000000         0000000         0000000         0000000           [0x83000020]         0000000         0000000         0000000         0000000         0000000           [0x83000040]         0000000         0000000         0000000         0000000         0000000           [0x83000050]         0000000         0000000         0000000         0000000         0000000           [0x83000050]         0000000         0000000         0000000         0000000         0000000           [0x83000050]         0000000         0000000         0000000         0000000         0000000           [0x83000070]         0000000         0000000         0000000         0000000         0000000           [0x83000080]         0000000         0000000         0000000         0000000         0000000           [0x83000080]         0000000         0000000         0000000         0000000         0000000           [0x83000000]         0000000         0000000         0000000         0000000         0000000           [0x83000000]         0000000         0000000         00000000         00000000         00	[3]or[R] : READ DMA (EXT) [4]or[D] : DUMP DATA IN DDR	
	[0x83000010]         00000000         0000000         0000000	

Figure 3-13 Clear buffer to be zero

User can exit this menu by input other key, such as 'x'.

💆 COM3:115200baud - Tera Ter 💼 💼 🗾	<
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
[G]oto [N]ext [P]rev [H]rbuf [R]dbuf [C]learbuf ?x	*
SATA host design menu [Ver = 2.3] [D]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : HRITE DMA (EXT) [3]or[R] : READ DMA (EXT) [4]or[D] : DUMP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK	
	Ŧ

Figure 3-14 Exit dump menu



### 3.6 TEST READ & WRITE BACK

Select '5' or 'T' for test sending "WRITE DMA (EXT)" and "READ DMA (EXT)" without fill or verify test pattern. Write and Read speed will be displayed as output for this menu, as shown in Figure 3-15. Two inputs are required to use this menu, similar to "READ DMA (EXT).

The test operation will be cancelled if CPU receives any input from user during operation.

🐸 COM3:115200baud - Tera Term VT 📃 💼	×
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
<pre>+++ TEST (Read&amp;Write back) selected +++ Enter Start LBR : 0 - 500118191 (0x10CF32RF) =&gt; 0 Enter Sector Count : 1 - 500118192 (0x10CF32B0) =&gt; 0x1000000 Total [256] loop 12345678901280681 </pre>	90
SATA host design menu [Ver = 2.3] [O]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : HRITE DMA (EXT) [3]or[R] : READ DMA (EXT) [4]or[D] : DUMP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK	•

Figure 3-15 Test Menu Operation

🐸 COM3:115200baud - Tera Term VT 📃 🔲 🖃	3
<u>F</u> ile <u>E</u> dit <u>S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
+++ TEST (Read&Hrite back) selected +++ Enter Start LBA : O - 500118191 (Ox1DCF32AF) => O Enter Sector Count : 1 - 500118192 (Ox1DCF32BO) => Ox1000000 Total [256] loop 1234	
command cancelled Reset in loop.	
Read : Total = 131[MB] , Time = 249[ms] , Transfer speed = 538[MB/s] Write : Total = 131[MB] , Time = 285[ms] , Transfer speed = 470[MB/s]	
SATA host design menu [Ver = 2.3] [O]or[X] : SATA RESET [1]or[I] : IDENTIFY DEVICE [2]or[H] : HRITE DHA (EXT) [3]or[R] : READ DHA (EXT) [4]or[D] : DUMP DATA IN DDR [5]or[T] : TEST READ & HRITE BACK	
	Ŧ

Figure 3-16 Test Operation cancelled



dg\_sata\_ip\_host\_demo\_instruction\_7series\_en.doc
4 Revision History

Revision	Date	Description	
1.0	21-Apr-14	Initial version release	
2.0	21-Jan-16	Support 7-series and KCU105 board	