



HCTL-IP Demo Instruction

Rev1.7 6-Jul-23

1	Environment Setup.....	2
2	Demo setup.....	7
3	Test Menu.....	13
3.1	Identify Device	13
3.2	Write Device	14
3.3	Read Device	17
3.4	Security erase.....	19
4	Revision History	20

HCTL-IP Demo Instruction

Rev1.7 6-Jul-23

This document describes the instruction to run SATA HCTL-IP demo on FPGA development board by using AB09-FMCRAID board. The demo is designed to write/verify data with SATA device. For 7-series board, the design supports both SATA-II and SATA-III device. For Ultrascale board, the design supports only SATA-III device. User controls test operation through Serial console.

1 Environment Setup

To run the demo on FPGA development board, please prepare following environment.

- 1) FPGA development board: AC701/KC705/VC707/VC709/ZC706/KCU105/ZCU102/VCU118/Zynq Mini-ITX
- 2) PC with Xilinx programmer software (Vivado) and Serial console software such as HyperTerminal
- 3) SATA cable for Zynq Mini-ITX or AB09-FMCRAID board for other FPGA boards
- 4) One SATA-II/III device connecting at CN0 of AB09-FMCRAID board
Note: KCU105/ZCU102/VCU118 board support only SATA-III device
- 5) Xilinx Power adapter for Xilinx board or ATX power supply for Zynq Mini-ITX board
- 6) micro USB cable for programming FPGA between FPGA board and PC
- 7) mini/micro USB cable for Serial console connecting between FPGA board and PC
- 8) For AC701 board only, connect CLK SMA board to the board.

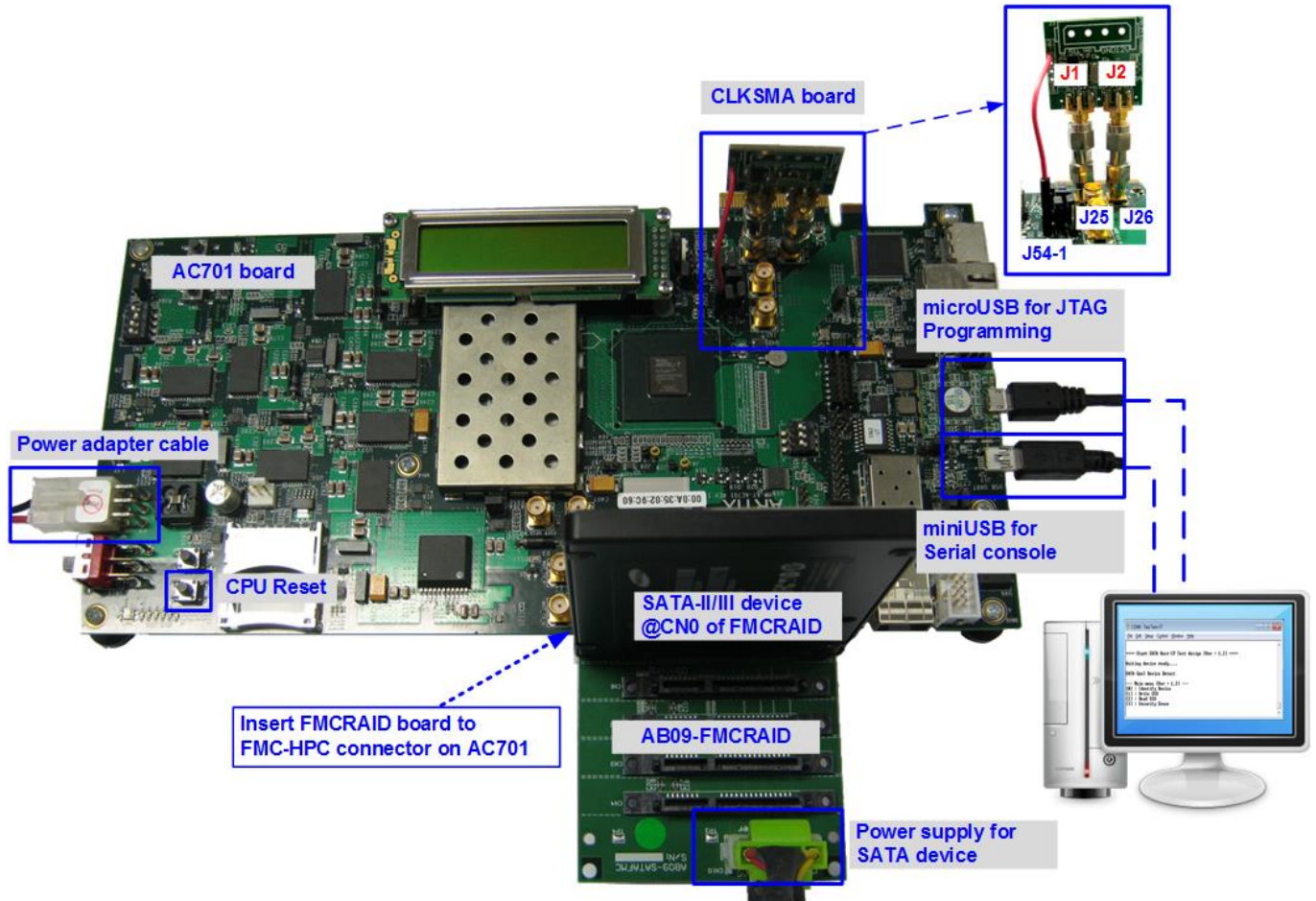


Figure 1-1 HCTL-IP demo setup on AC701

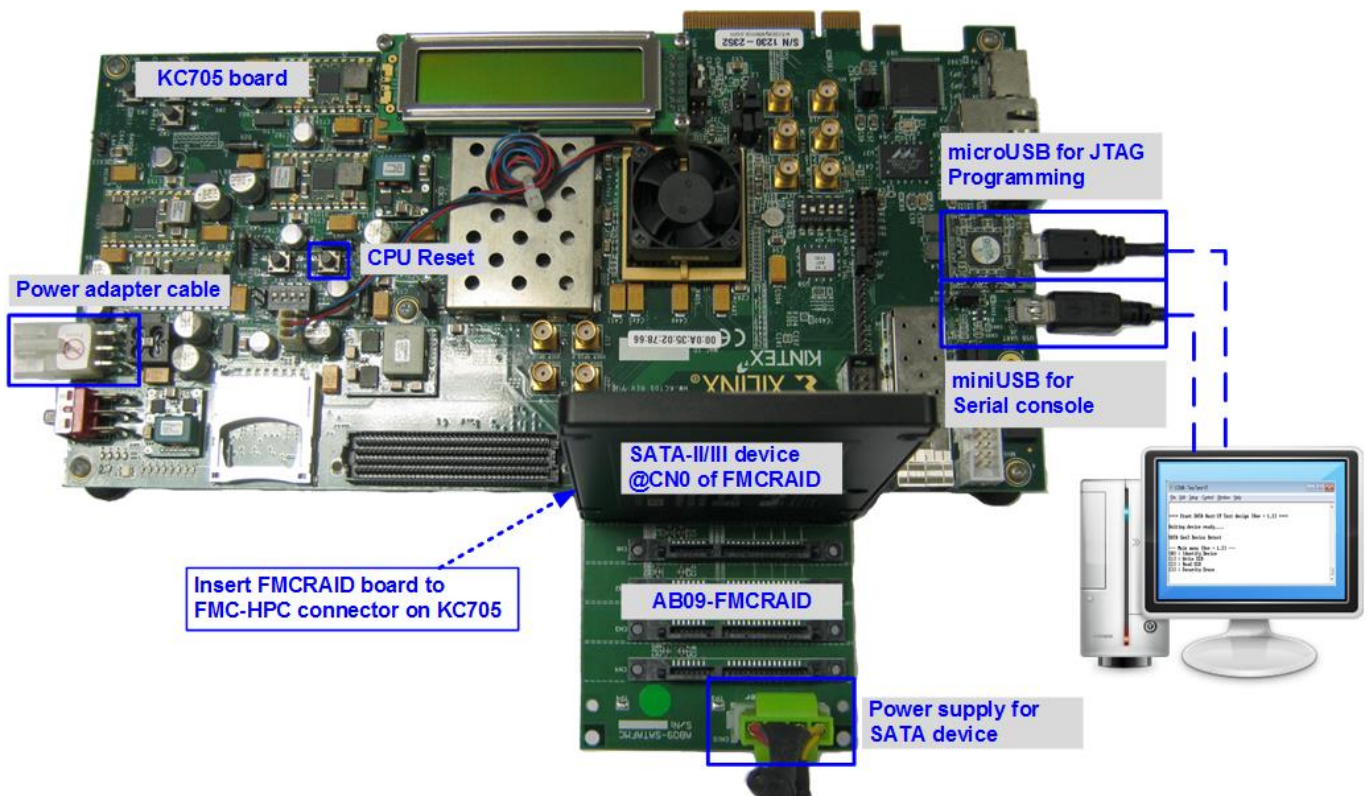


Figure 1-2 HCTL-IP demo setup on KC705

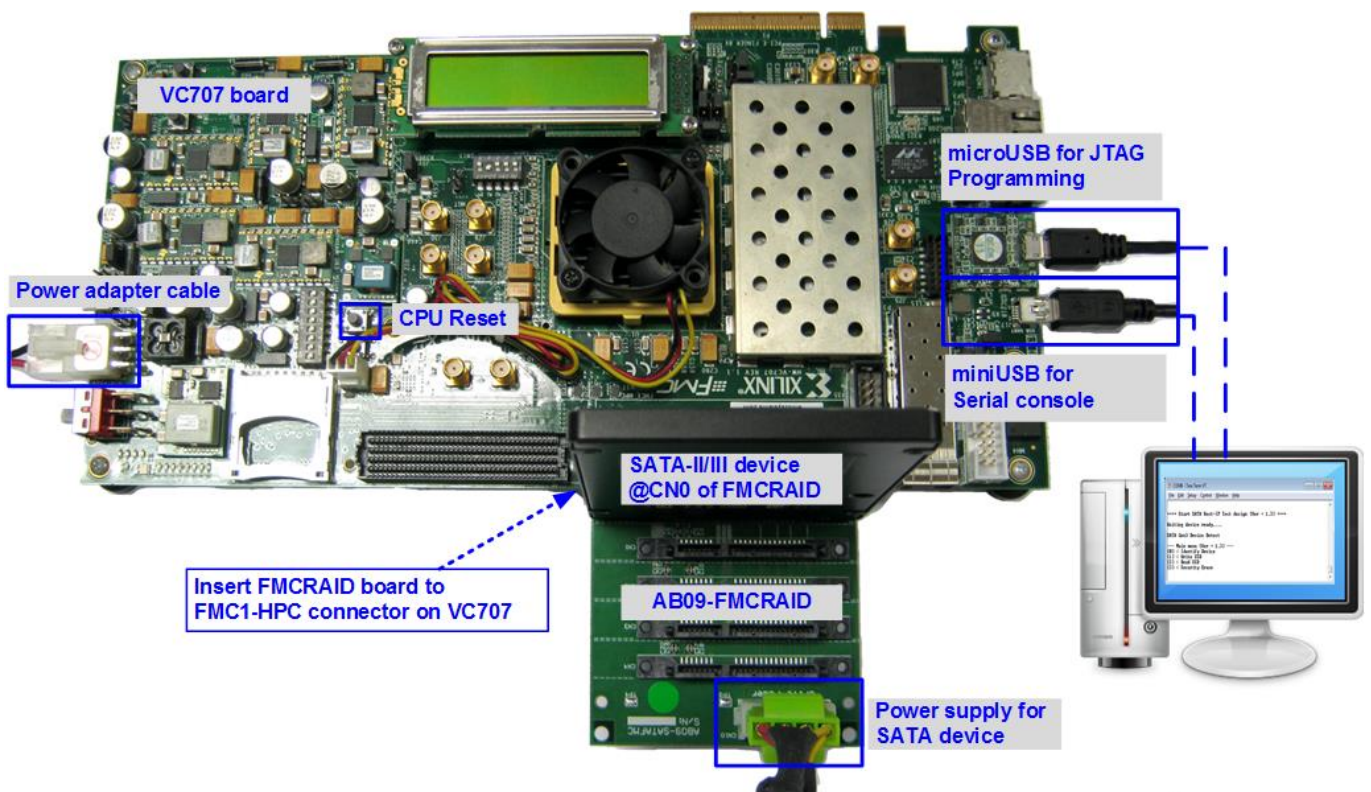


Figure 1-3 HCTL-IP demo setup on VC707

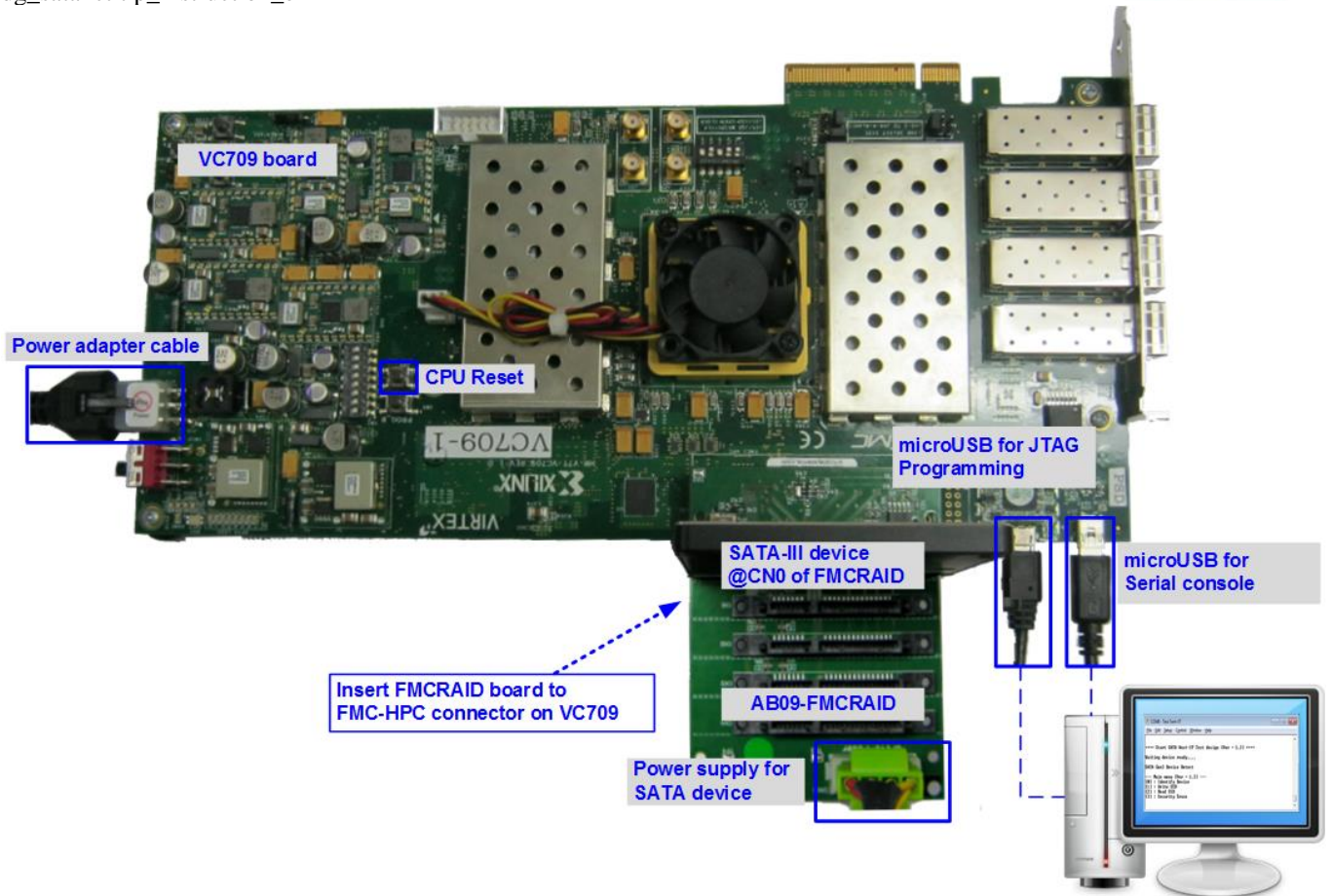


Figure 1-4 HCTL-IP demo setup on VC709

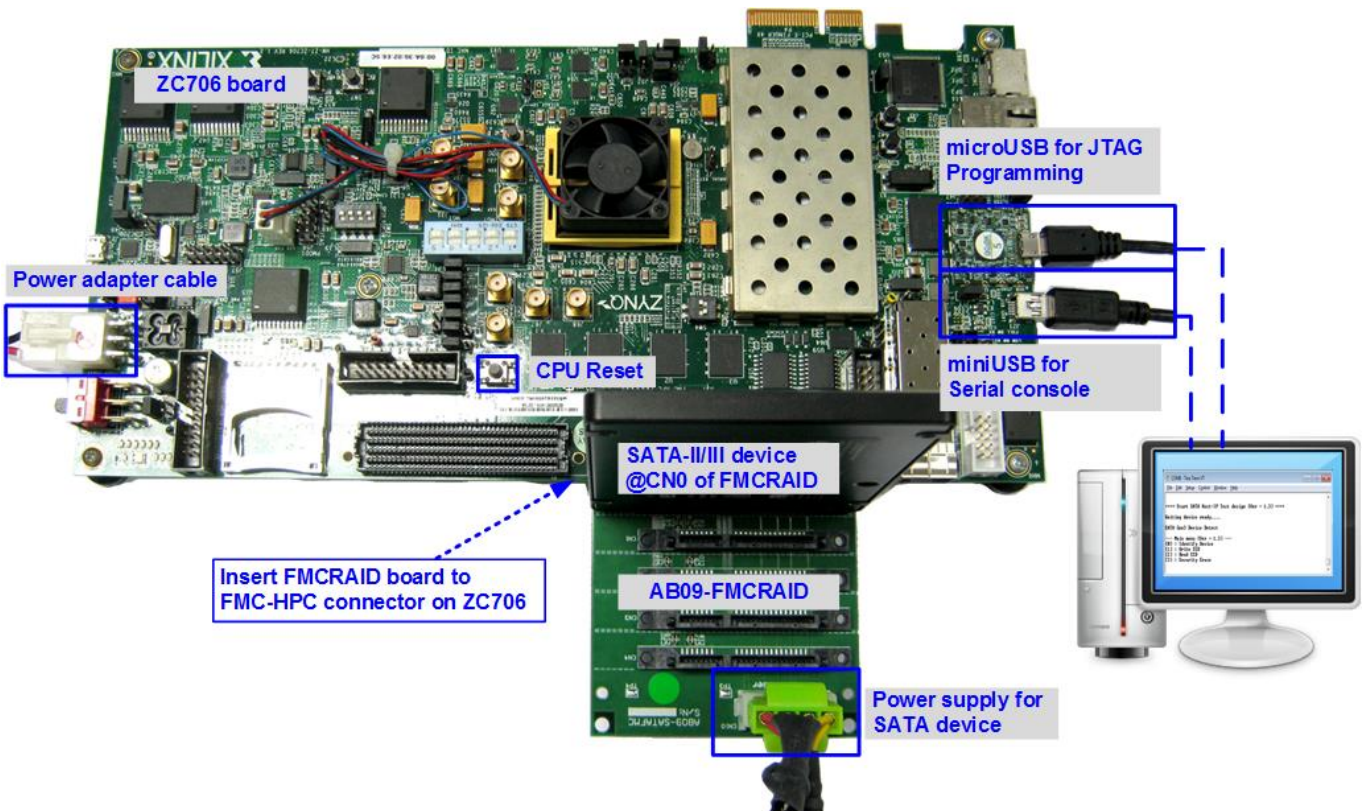


Figure 1-5 HCTL-IP demo setup on ZC706

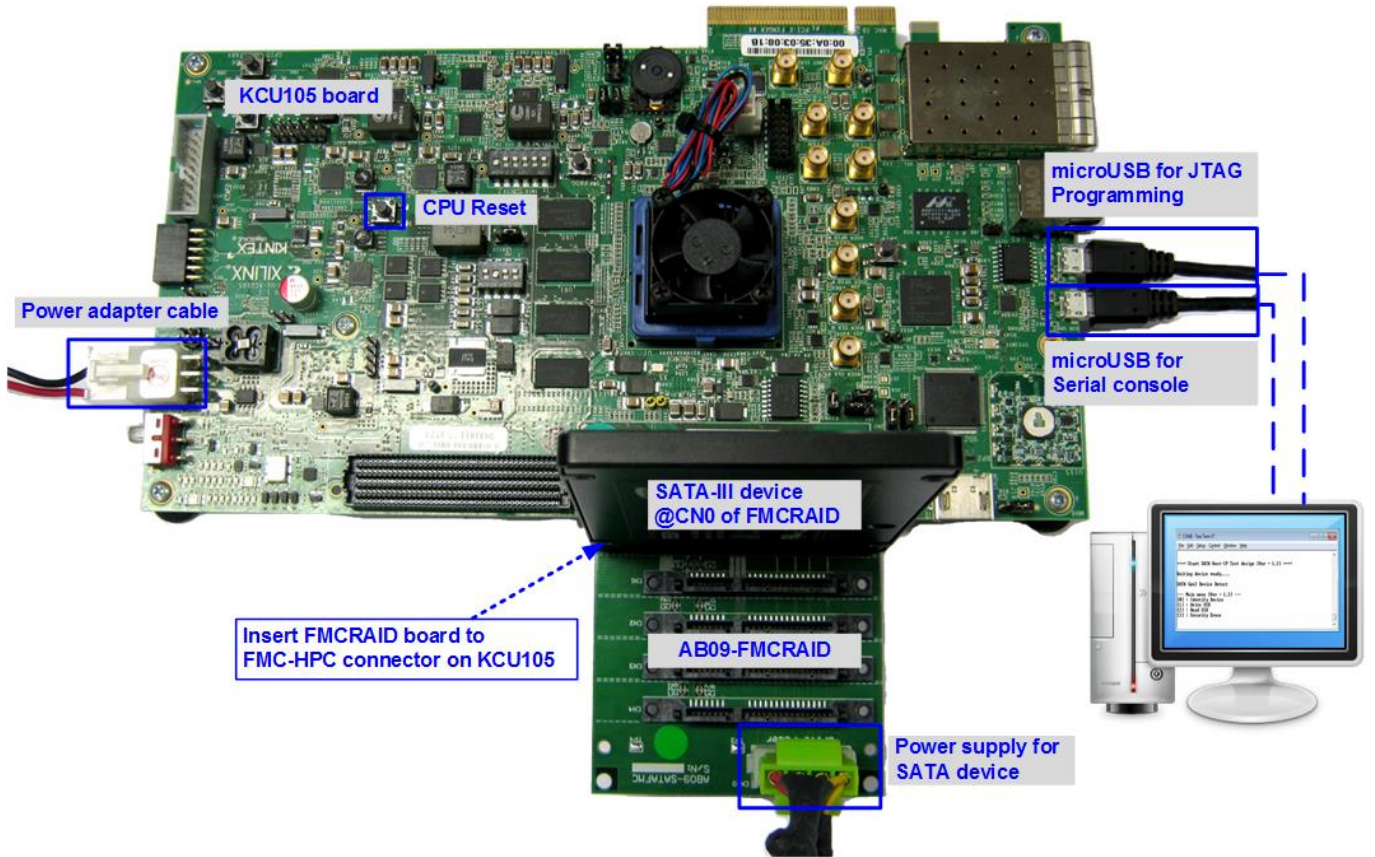


Figure 1-6 HCTL-IP demo setup on KCU105

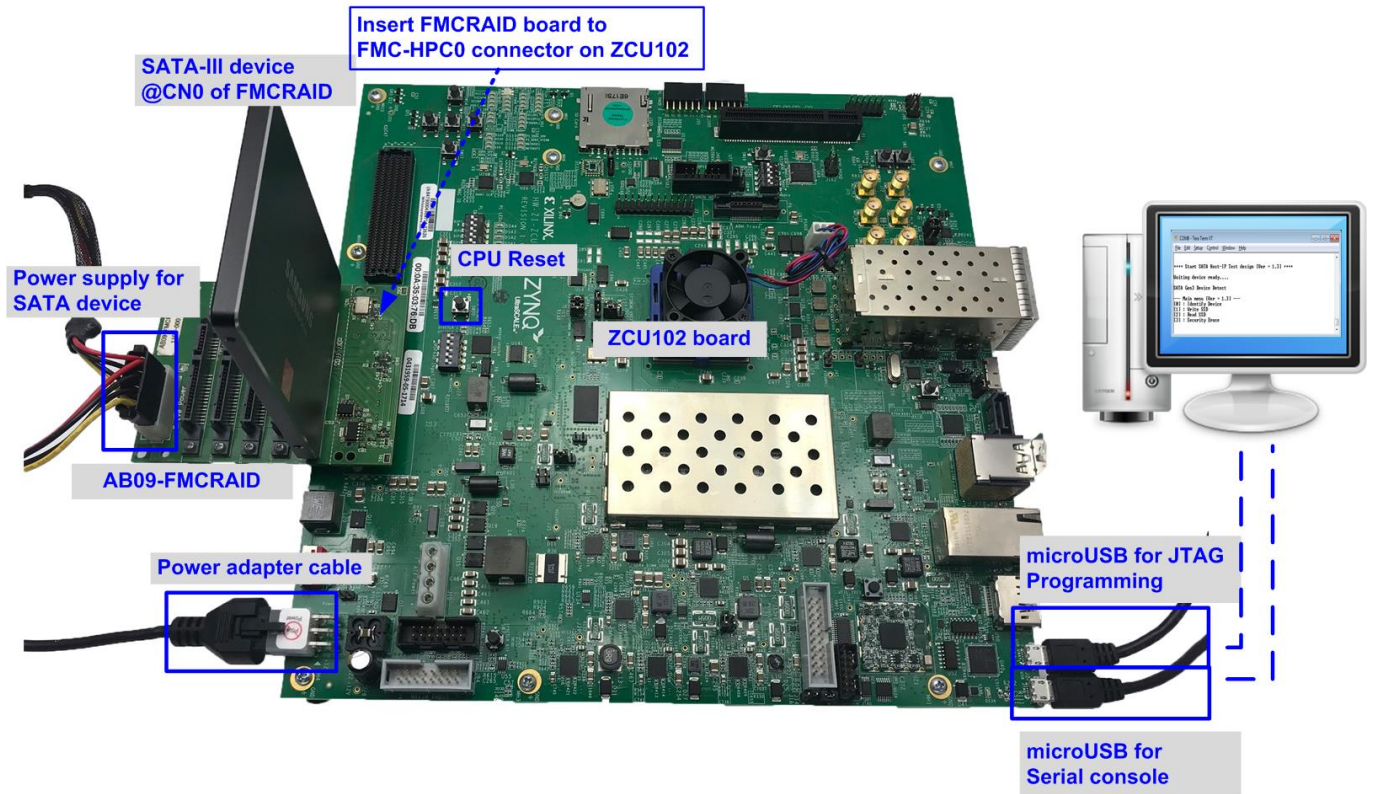


Figure 1-7 HCTL-IP demo setup on ZCU102

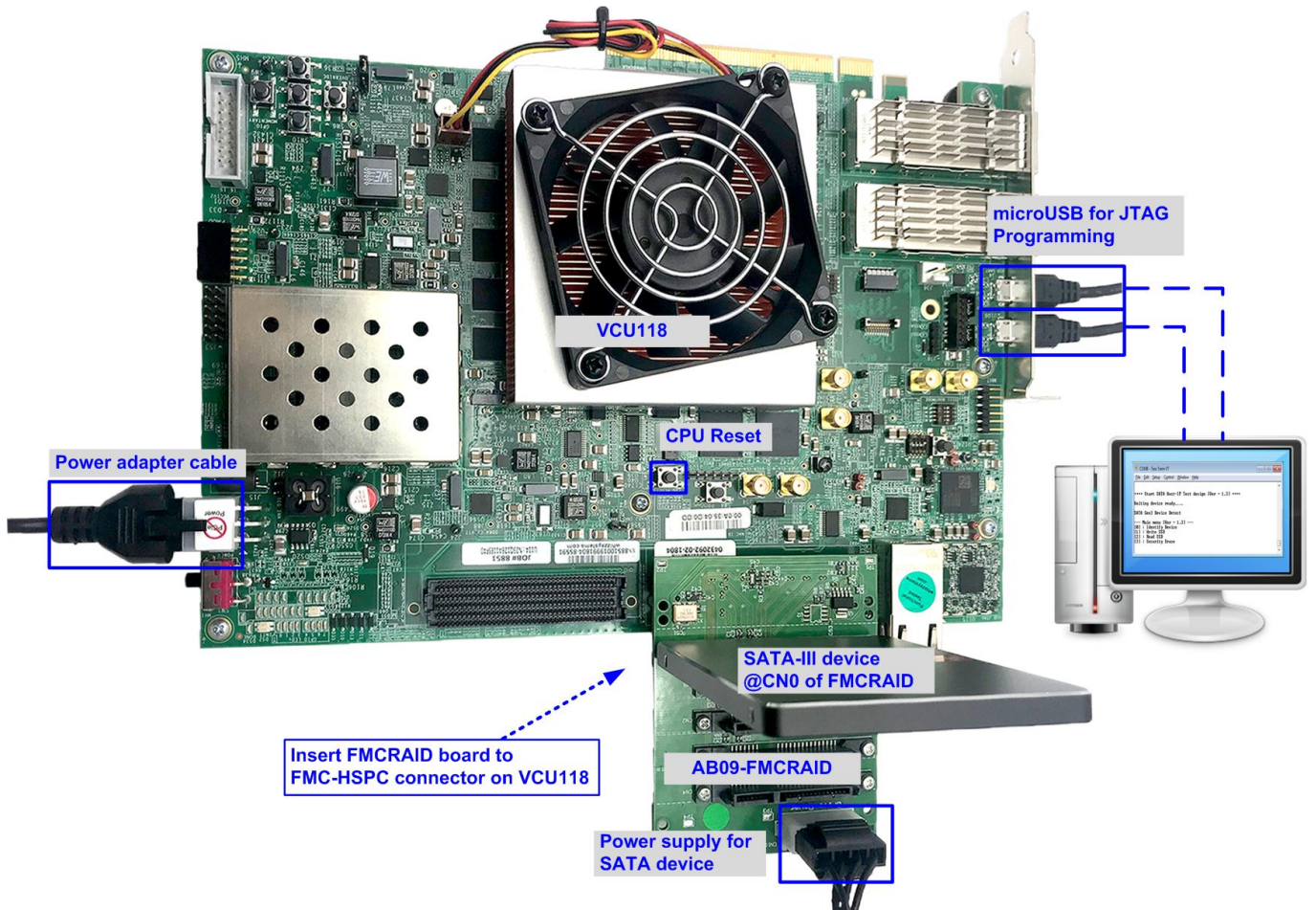


Figure 1-8 HCTL-IP demo setup on VCU118

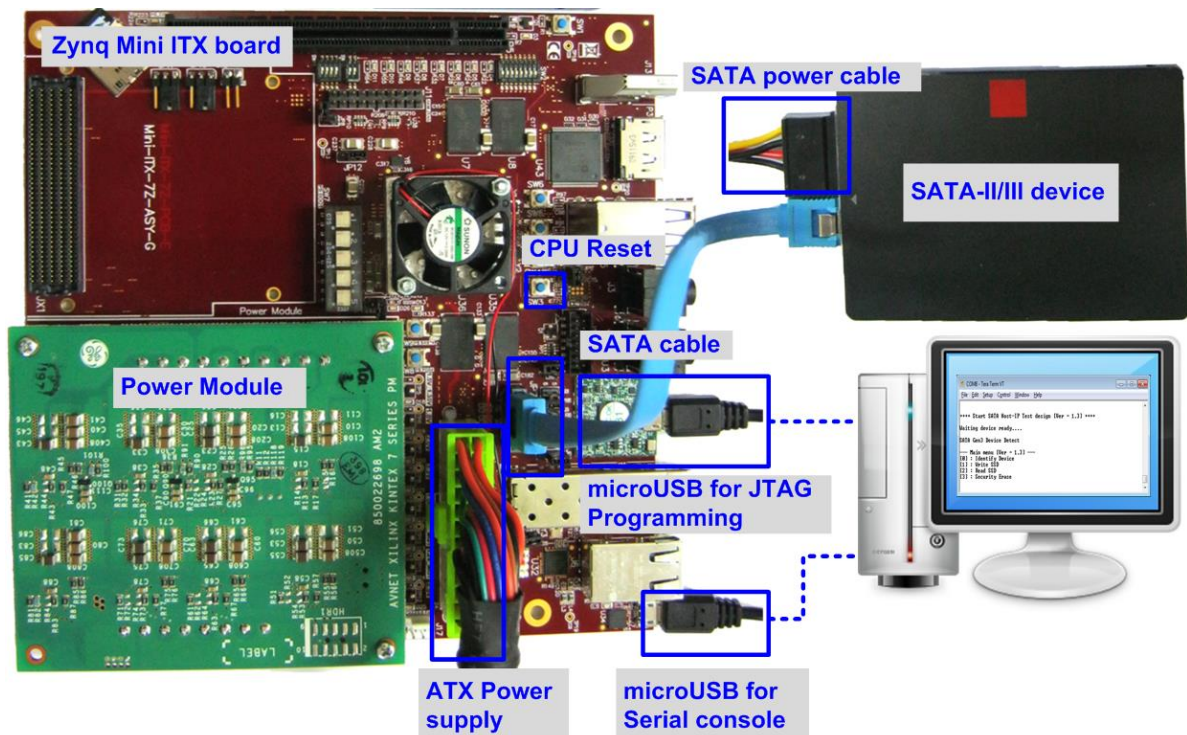


Figure 1-9 HCTL-IP demo setup on Zynq Mini-ITX

2 Demo setup

- 1) Power off system.
- 2) Set up board option.
 - a) For AC701 board, connect CLKSMA board to the board.



Figure 2-1 Connect CLKSMA board to AC701

- b) For ZC706 board,
 - i. Set SW11="00000" to configure PS from JTAG, as shown in Figure 2-2.
 - ii. Set SW4="01" to connect JTAG with USB-to-JTAG interface, as shown in Figure 2-3.



Figure 2-2 SW11 setting to configure PS from JTAG on ZC706 board



Figure 2-3 SW4 setting to use USB-to-JTAG on ZC706 board

- c) For Zynq Mini-ITX board only,
 - i. Set SW7="00000" to configure PS from JTAG, as shown in Figure 2-4.
 - ii. As shown in Figure 2-5, install a jumper on JP1 pins 1-2 to enable JTAG chain.
 - iii. Install the power module onto the board via J8, J9, J10 connectors.
 - iv. Connect ATX power cable to FPGA board via P2 connector.

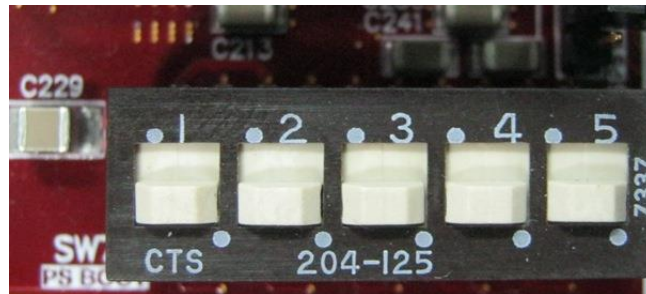


Figure 2-4 SW7 setting to configure PS from JTAG on Zynq Mini-ITX

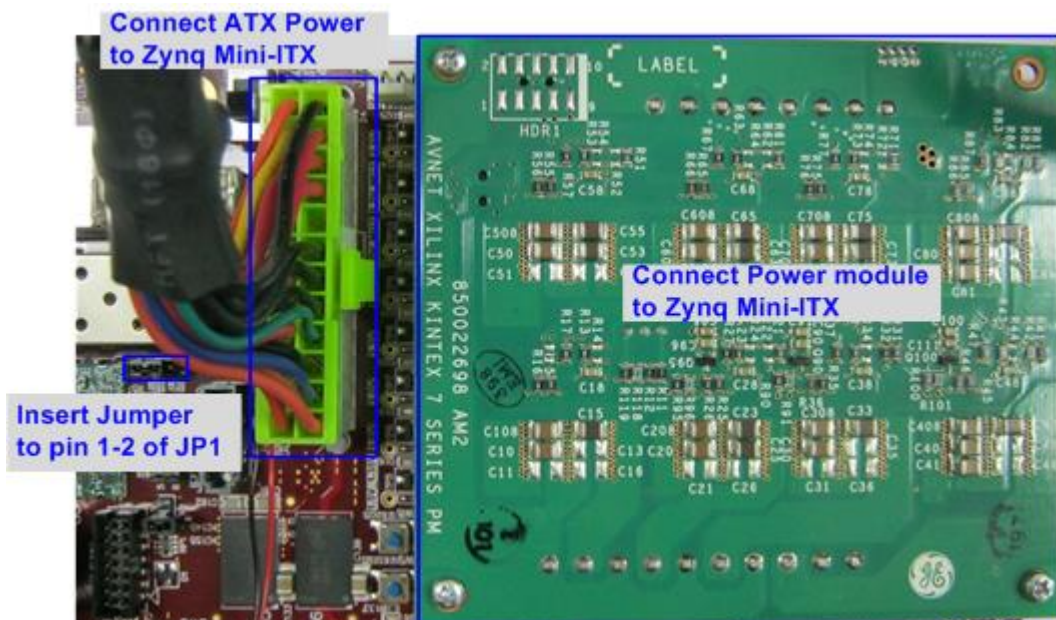


Figure 2-5 The power module installed onto the board

- d) For ZCU102, set SW6="0000" (SW = ON) to configure PS from JTAG, as shown in Figure 2-6.



Figure 2-6 SW6 setting to configure PS from JTAG on ZCU102

- 3) Connect the SATA-II/III device to the board by the following steps.
 - a) For Xilinx board,
 - i. Connect AB09-FMCRAID board to FMC connector on Xilinx board following Figure 1-1 - Figure 1-9.
 - ii. Connect SATA-II/III device to CN0 on FMCRAID board.
Note: KCU105/ZCU102/VCU118 supports only SATA-III device.
 - iii. Connect power to power connector on FMCRAID board.
- The connection on AB09 is shown in Figure 2-7.

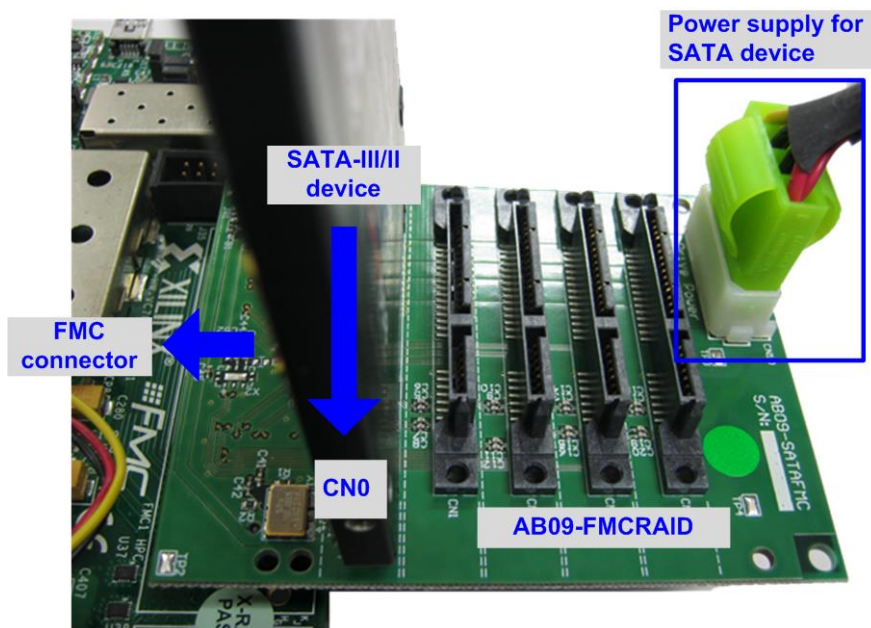


Figure 2-7 AB09-FMCRAID connection

- b) For Zynq Mini-ITX board,
 - i. Connect the device to the SATA connector (J12) on the board using SATA cable.
 - ii. Connect SATA Power cable to the device.
- The connection on Zynq Mini-ITX is shown in Figure 2-8.

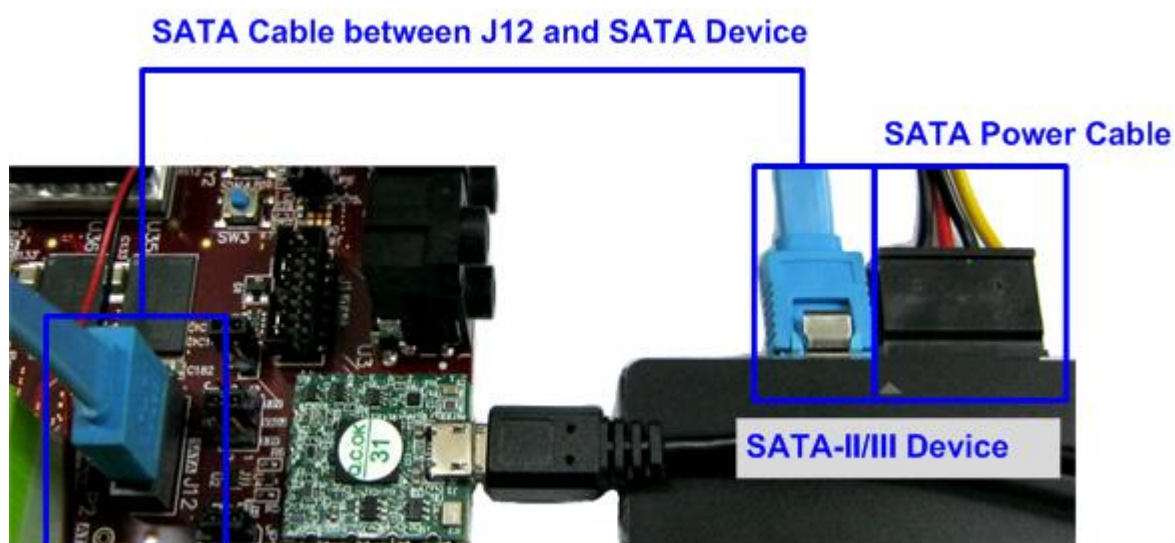


Figure 2-8 SATA-II/III device connection for Zynq Mini-ITX

- 4) Connect micro USB cable from FPGA board to PC for JTAG programming
- 5) Connect mini/micro USB cable from FPGA board to PC for Serial console.



Figure 2-9 USB cable connection

- 6) Power on FPGA board and power supply for SATA device.
- 7) Open Serial console such as TeraTerm, HyperTerminal. Set Buad rate=115,200 Data=8 bit Non-Parity Stop=1.
- 8) a) For Zynq device board (Zynq Mini-ITX/ZC706/ZCU102), open Vivado TCL shell. Change current directory to ready_for_download. Next, run BoardName_HSataIPTest.bat (zc706_HSataIPTest.bat, zcu102_HSataIPTest.bat, MiniITX_7z045_HSataIPTest.bat), as shown in Figure 2-10

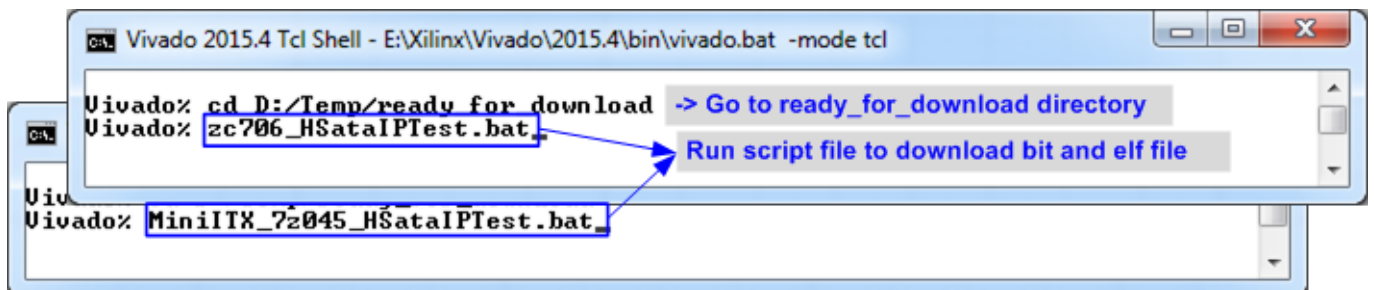


Figure 2-10 Command script for download demo file on Zynq device board

b) For AC701/KC705/VC707/VC709/KCU105/VCU118 board, use Vivado tool to download configuration file which includes CPU firmware, as shown in Figure 2-11.

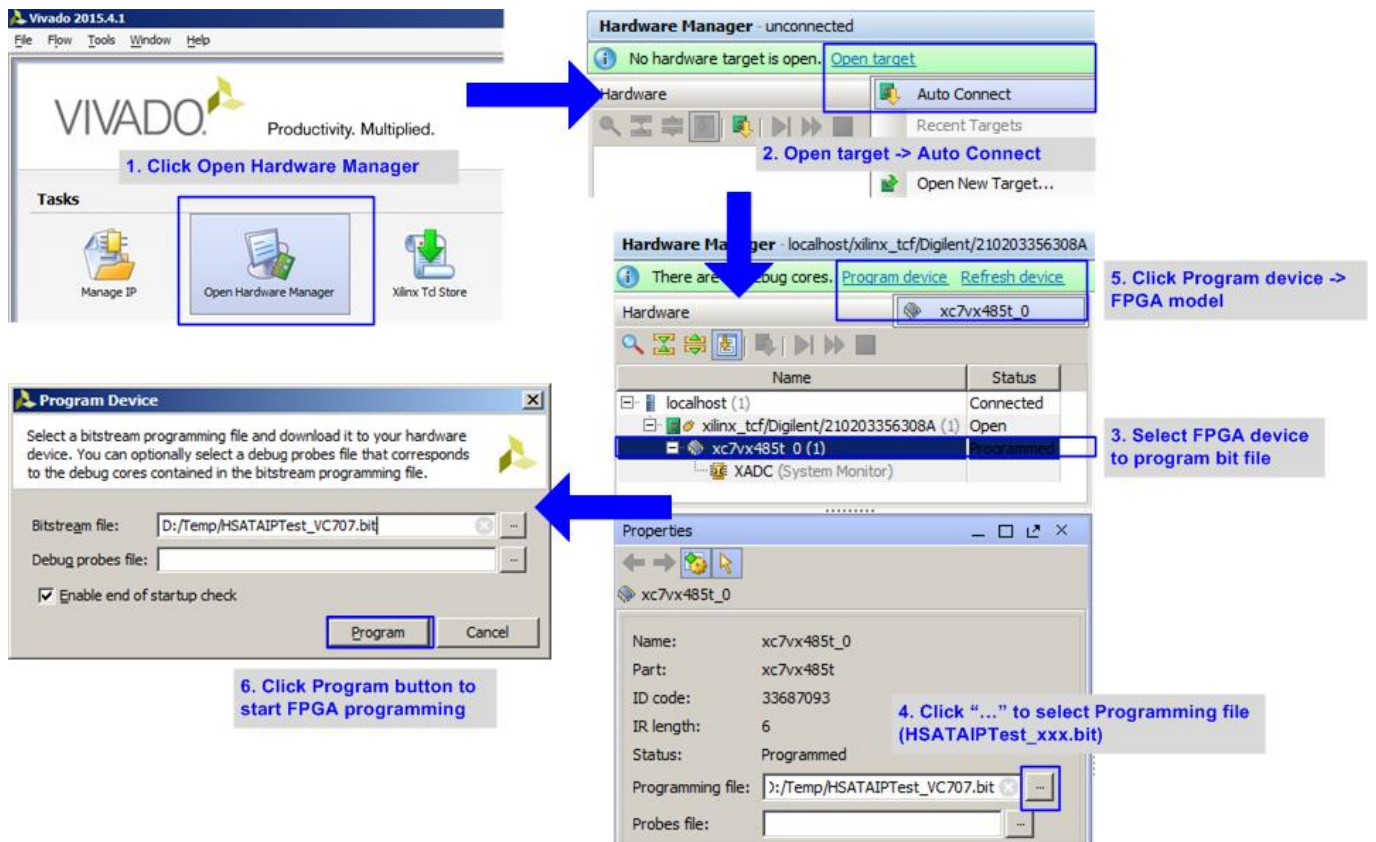


Figure 2-11 Programmed by Vivado

9) Check LED status on FPGA board. The description of LED is as follows.

GPIO LED	ON	OFF
0/D4	Normal operation	System is in reset condition
1/R/D5	System is busy	Idle status
2/C/D6	Error detect	Normal operation
3/L/D7	Data verification fail	Normal operation

Table 1 LED Definition

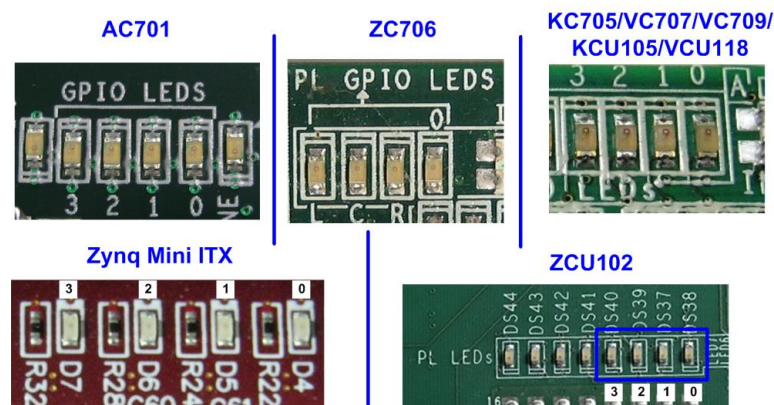


Figure 2-12 4-bit LED Status for user output

10) After programming completely, LED[0] and LED[1] are ON during SATA initialization process. Then, LED[1] changes to OFF after SATA completes initialization process and system is ready to receive command from user. After that, main menu is displayed, as shown in Figure 2-14.

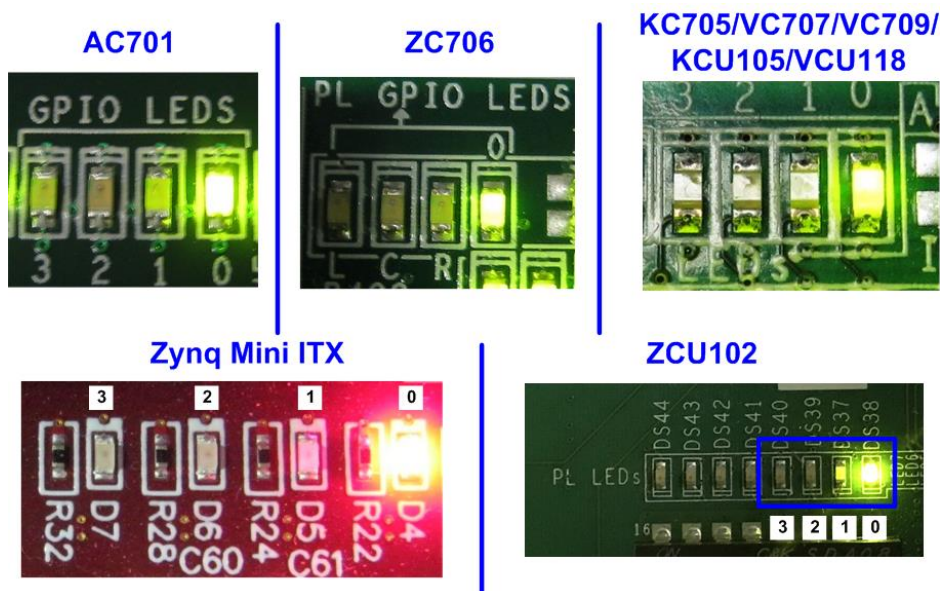


Figure 2-13 LED status after program configuration file and SATA initialization complete

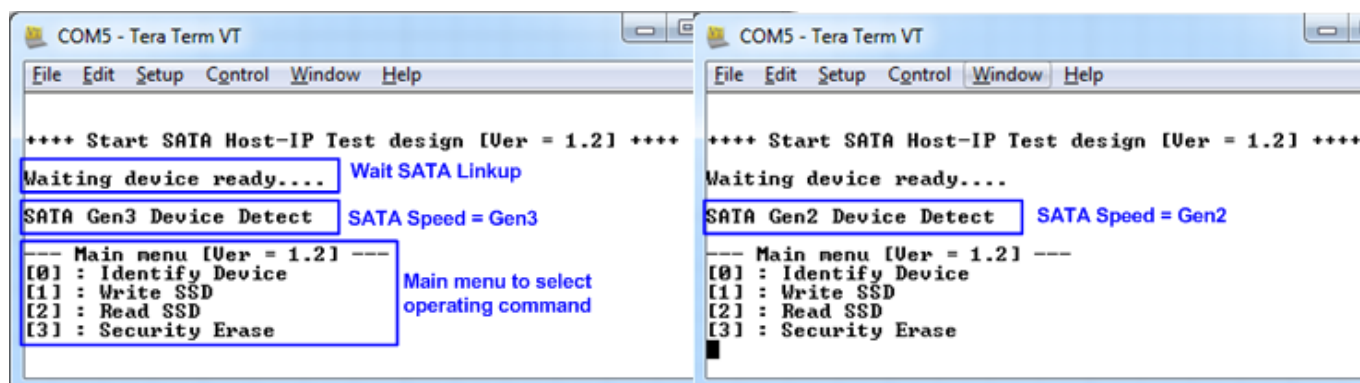


Figure 2-14 Main menu after program configuration file and SATA initialization complete

3 Test Menu

3.1 Identify Device

Select '0' to send Identify device command to SATA device. When operation is completed, all information is displayed on the console, i.e.

- 1) Device Model number
- 2) Security feature set is supported or not. If device is not supported, user must not run menu 3 to test Security erase command.
- 3) Normal Erase Mode Time: This is estimation time to complete Security erase command. Minimum valid value is 2 minutes. This information is displayed when the device supports Security feature set.
- 4) Device capacity which is output from HCTL-IP.

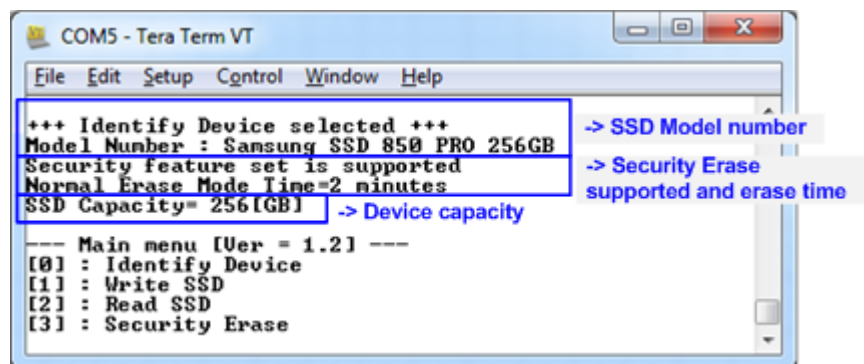


Figure 3-1 Result from Identify Device menu

3.2 Write Device

Select '1' to send Write command to SATA device. Three inputs are required for this menu.

- 1) Start LBA: Input start address of SATA device in sector unit. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
- 2) Sector Count: Input total transfer size in sector unit. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
- 3) Test pattern: Select test pattern of test data for writing to SATA device. Five patterns can be selected, i.e. 32-bit increment, 32-bit decrement, all 0, all 1, and 32-bit LFSR counter.

As shown in Figure 3-2, if all inputs are valid, the operation will be started. During writing data, current transfer size is displayed on the console to show that system still run. Finally, test performance, total size, and total time usage are displayed on the console as test result.

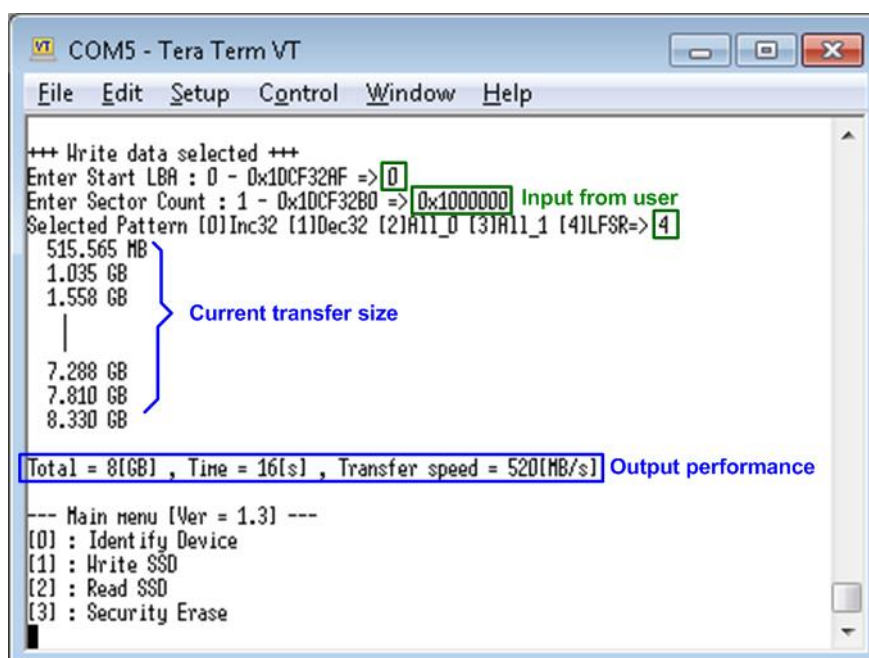


Figure 3-2 Input and result of Write Device menu

Offset	←64-bit header of each sector→															
	48-bit LBA Address								0x0000		32-bit increment data					
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000000000	00	00	00	00	00	00	00	00	02	00	00	00	03	00	00	00
0000000010	04	00	00	00	05	00	00	00	06	00	00	00	07	00	00	00
0000000020	08	00	00	00	09	00	00	00	0A	00	00	00	0B	00	00	00
0000000030	0C	00	00	00	0D	00	00	00	0E	00	00	00	0F	00	00	00
0000000040	10	00	00	00	11	00	00	00	12	00	00	00	13	00	00	00
0000000050	14	00	00	00	15	00	00	00	16	00	00	00	17	00	00	00
0000000060	18	00	00	00	19	00	00	00	1A	00	00	00	1B	00	00	00
0000000070	1C	00	00	00	1D	00	00	00	1E	00	00	00	1F	00	00	00
0000000080	20	00	00	00	21	00	00	00	22	00	00	00	23	00	00	00
0000000090	24	00	00	00	25	00	00	00	26	00	00	00	27	00	00	00
00000000A0	28	00	00	00	29	00	00	00	2A	00	00	00	2B	00	00	00
00000000B0	2C	00	00	00	2D	00	00	00	2E	00	00	00	2F	00	00	00
00000000C0	30	00	00	00	31	00	00	00	32	00	00	00	33	00	00	00
00000000D0	34	00	00	00	35	00	00	00	36	00	00	00	37	00	00	00
00000000E0	38	00	00	00	39	00	00	00	3A	00	00	00	3B	00	00	00
00000000F0	3C	00	00	00	3D	00	00	00	3E	00	00	00	3F	00	00	00
0000000100	40	00	00	00	41	00	00	00	42	00	00	00	43	00	00	00
0000000110	44	00	00	00	45	00	00	00	46	00	00	00	47	00	00	00
0000000120	48	00	00	00	49	00	00	00	4A	00	00	00	4B	00	00	00
0000000130	4C	00	00	00	4D	00	00	00	4E	00	00	00	4F	00	00	00
0000000140	50	00	00	00	51	00	00	00	52	00	00	00	53	00	00	00
0000000150	54	00	00	00	55	00	00	00	56	00	00	00	57	00	00	00
0000000160	58	00	00	00	59	00	00	00	5A	00	00	00	5B	00	00	00
0000000170	5C	00	00	00	5D	00	00	00	5E	00	00	00	5F	00	00	00
0000000180	60	00	00	00	61	00	00	00	62	00	00	00	63	00	00	00
0000000190	64	00	00	00	65	00	00	00	66	00	00	00	67	00	00	00
00000001A0	68	00	00	00	69	00	00	00	6A	00	00	00	6B	00	00	00
00000001B0	6C	00	00	00	6D	00	00	00	6E	00	00	00	6F	00	00	00
00000001C0	70	00	00	00	71	00	00	00	72	00	00	00	73	00	00	00
00000001D0	74	00	00	00	75	00	00	00	76	00	00	00	77	00	00	00
00000001E0	78	00	00	00	79	00	00	00	7A	00	00	00	7B	00	00	00
00000001F0	7C	00	00	00	7D	00	00	00	7E	00	00	00	7F	00	00	00
0000000200	01	00	00	00	00	00	00	00	82	00	00	00	83	00	00	00
0000000210	84	00	00	00	85	00	00	00	86	00	00	00	87	00	00	00
0000000220	88	00	00	00	89	00	00	00	8A	00	00	00	8B	00	00	00
0000000230	8C	00	00	00	8D	00	00	00	8E	00	00	00	8F	00	00	00

Offset	←64-bit header of each sector→															
	48-bit LBA Address								0x0000		32-bit LFSR pattern					
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000000000	00	00	00	00	00	00	00	00	01	00	00	00	02	00	00	00
0000000010	04	00	00	00	09	00	00	00	12	00	00	00	24	00	00	00
0000000020	08	00	00	00	09	00	00	00	24	01	00	00	49	02	00	00
0000000030	0C	00	00	00	09	00	00	00	49	12	00	00	92	24	00	00
0000000040	10	00	00	00	09	00	00	00	92	24	01	00	24	49	02	00
0000000050	14	00	00	00	09	00	00	00	24	49	12	00	49	92	24	00
0000000060	18	00	00	00	09	00	00	00	49	92	00	00	4F	92	24	01
0000000070	1C	00	00	00	09	00	00	00	F3	24	49	12	E7	49	92	24
0000000080	20	00	00	00	09	00	00	00	3C	49	92	04	79	92	24	09
0000000090	24	00	00	00	09	00	00	00	CF	93	24	49	9E	27	49	92
00000000A0	28	00	00	00	09	00	00	00	F5	3C	49	92	EB	79	92	24
00000000B0	2C	00	00	00	09	00	00	00	D7	F3	24	49	AE	E7	49	92
00000000C0	30	00	00	00	09	00	00	00	5D	CF	93	24	BA	9E	27	49
00000000D0	34	00	00	00	09	00	00	00	D7	F5	3C	49	AE	EB	79	92
00000000E0	38	00	00	00	09	00	00	00	77	70	5D	CF	EE	E0	BA	9E
00000000F0	3C	00	00	00	09	00	00	00	70	07	D7	F5	E0	0E	AE	EB
0000000100	40	00	00	00	09	00	00	00	07	77	70	5D	0E	EE	E0	BA
0000000110	44	00	00	00	09	00	00	00	73	70	07	D7	E6	E0	0E	AE
0000000120	48	00	00	00	09	00	00	00	34	07	77	70	68	0E	EE	E0
0000000130	4C	00	00	00	09	00	00	00	47	73	70	07	8E	E6	E0	0E
0000000140	50	00	00	00	09	00	00	00	74	34	07	77	E9	68	0E	EE
0000000150	54	00	00	00	09	00	00	00	4C	47	73	70	98	8E	E6	E0
0000000160	58	00	00	00	09	00	00	00	C6	74	34	07	8D	E9	68	0E
0000000170	5C	00	00	00	09	00	00	00	6E	4C	47	73	DC	98	8E	E6
0000000180	60	00	00	00	09	00	00	00	E1	C6	74	34	C3	8D	E9	68
0000000190	64	00	00	00	09	00	00	00	1A	6E	4C	47	34	DC	98	8E
00000001A0	68	00	00	00	09	00	00	00	A0	E1	C6	74	41	C3	8D	E9
00000001B0	6C	00	00	00	09	00	00	00	0C	1A	6E	4C	18	34	DC	98
00000001C0	70	00	00	00	09	00	00	00	C0	A0	E1	C6	81	41	C3	8D
00000001D0	74	00	00	00	09	00	00	00	0F	0C	1A	6E	1F	18	34	DC
00000001E0	78	00	00	00	09	00	00	00	FF	C0	A0	E1	FF	81	41	C3
00000001F0	7C	00	00	00	09	00	00	00	FA	0F	0C	1A	F4	1F	18	34
0000000200	01	00	00	00	00	00	00	00	01	00	00	00	00	00	00	00
0000000210	84	00	00	00	85	00	00	00	09	00	00	00	12	00	00	00
0000000220	88	00	00	00	89	00	00	00	92	00	00	00	24	01	00	00
0000000230	8C	00	00	00	8D	00	00	00	24	09	00	00	49	12	00	00

Figure 3-3 Example Test data in sector#0/#1 by increment/LFSR pattern

Test data of each sector has different 64-bit header which consists of 48-bit LBA address and 16-bit of zero value. 48-bit LBA address is unique value for each sector. The data after 64-bit header is the test pattern which is selected by user. The example of test pattern is shown in Figure 3-3. 32-bit increment pattern is in left window and 32-bit LFSR pattern is in right window.

Figure 3-4 – Figure 3-6 show error message when user input is invalid. “Invalid input” message is displayed on the console. Then, it returns to main menu to receive new command.

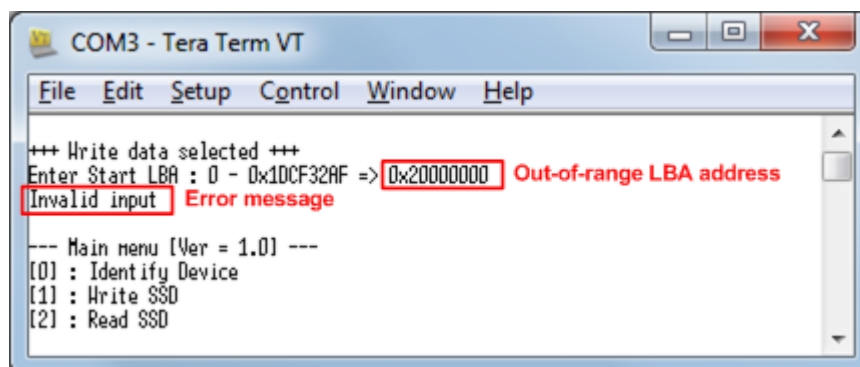


Figure 3-4 Invalid Start LBA input

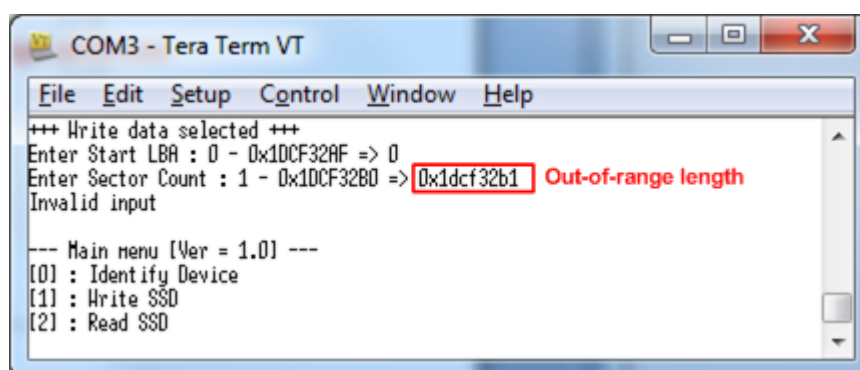


Figure 3-5 Invalid Sector count input

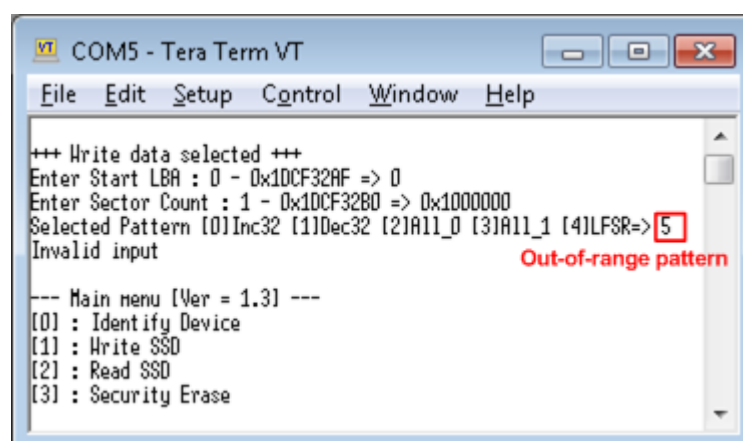


Figure 3-6 Invalid Test pattern input

3.3 Read Device

Select '2' to send Read command to SATA device. Three inputs are required for this menu.

- 1) Start LBA: Input start address of SATA device in sector unit. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
- 2) Sector Count: Input total transfer size in sector unit. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
- 3) Test pattern: Select test pattern to verify data from SATA device. Test pattern must be matched with the test pattern using in Write Command menu. Five patterns can be selected, i.e. 32-bit increment, 32-bit decrement, all 0, all 1, and 32-bit LFSR counter.

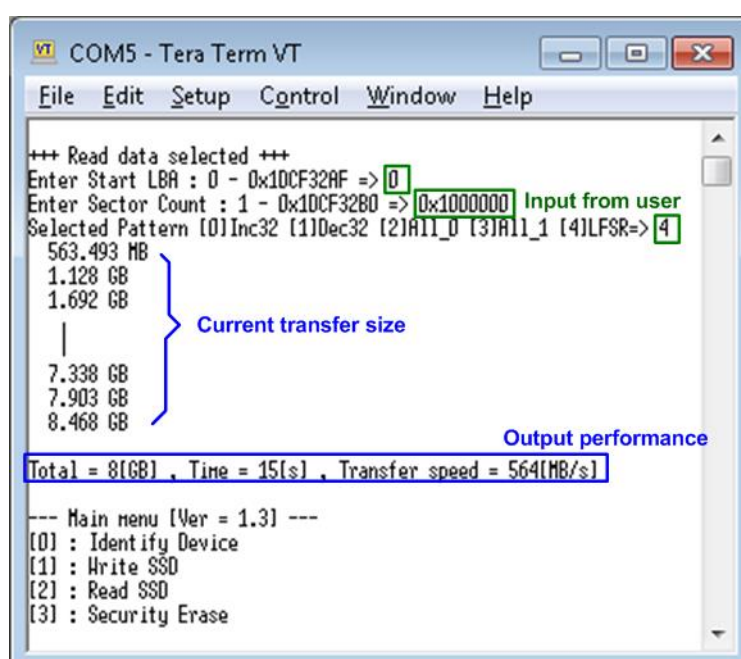


Figure 3-7 Input and result of Read Device menu

Similar to Write Command menu, if all inputs are valid, test system will read data from SATA device. Test performance, total size, and total time usage are displayed after end of transfer. "Invalid input" will be displayed if some inputs are out-of-range.

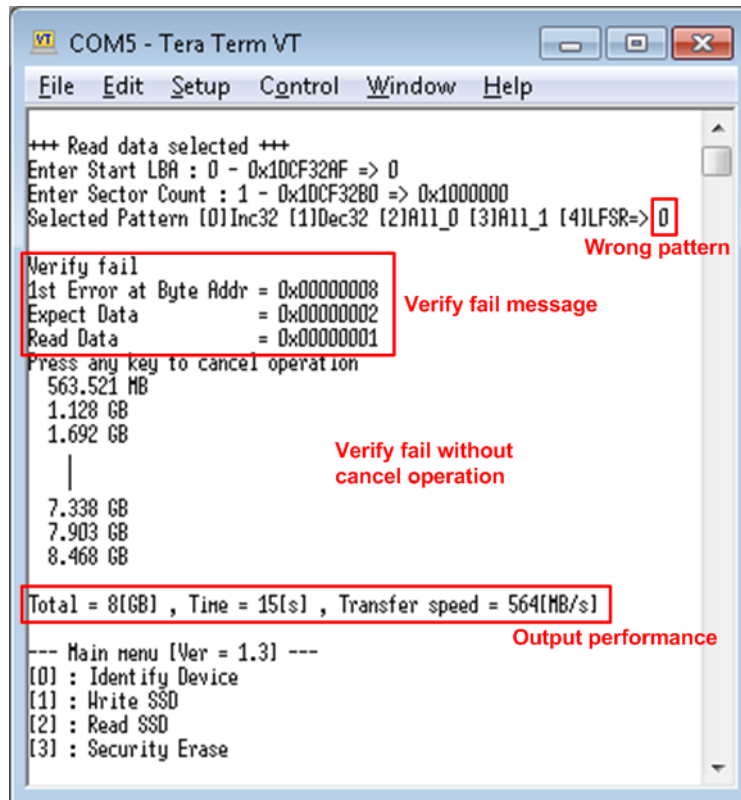


Figure 3-8 Data verification is failed but wait until read complete

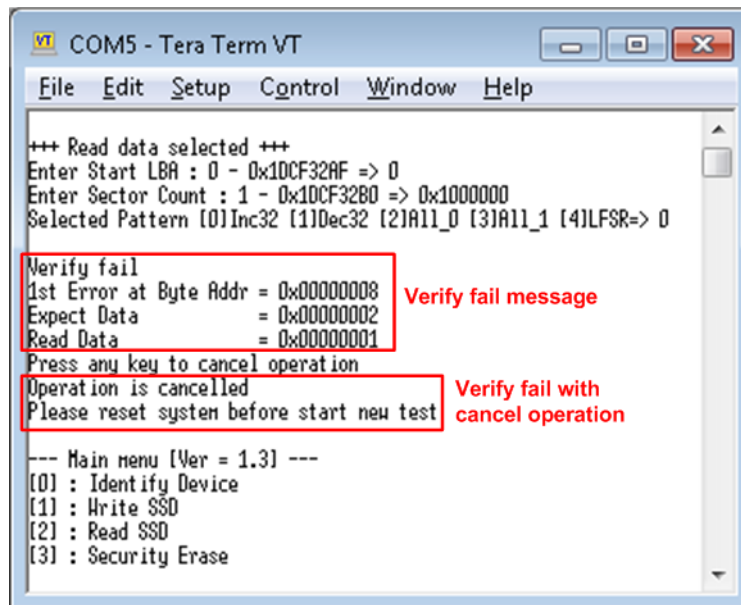


Figure 3-9 Data verification is failed and press any key to cancel operation

Figure 3-8 and Figure 3-9 show the error message when data verification is failed. “Verify fail” is displayed with error address, expected data, and read data. User can press any key to cancel read operation or wait until all read process complete.

In case of cancel operation, the previous command does not complete in good sequence. It is recommended to power-off/on SATA device and press “RESET” button to restart system.

3.4 Security erase

Select '3' to send Security Erase command to SATA device. Please confirm that SATA device supports Security Erase feature by using Identify device menu. The estimation time of security erase is also displayed in Identify device menu.

After selecting the menu, confirmation message is displayed on the console. User inputs 'y' or 'Y' to continue Security erase operation or input other keys to cancel operation.

Number 0-9 is displayed on the console every second to show that system still run. After complete the operation, total time usage is displayed as shown in Figure 3-10.

Figure 3-11 shows the example when user inputs other keys to cancel the command.

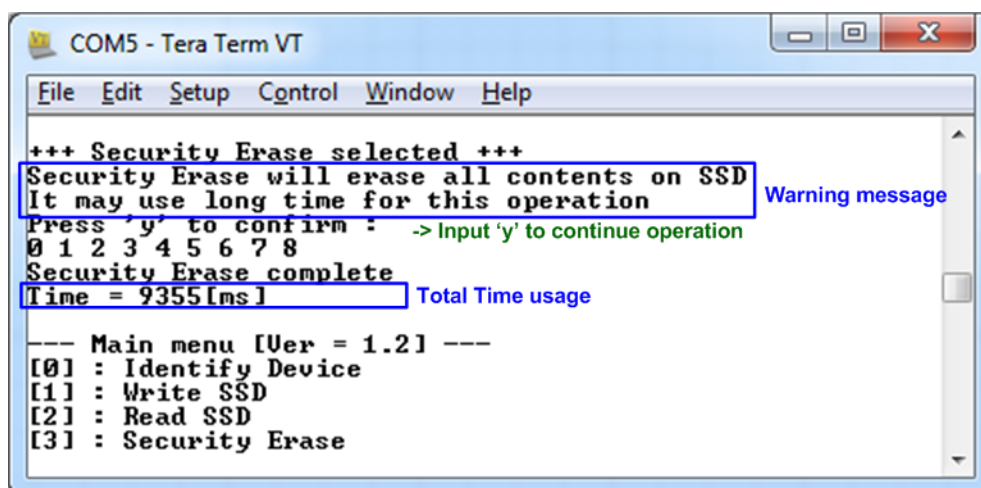


Figure 3-10 Result from Security Erase command

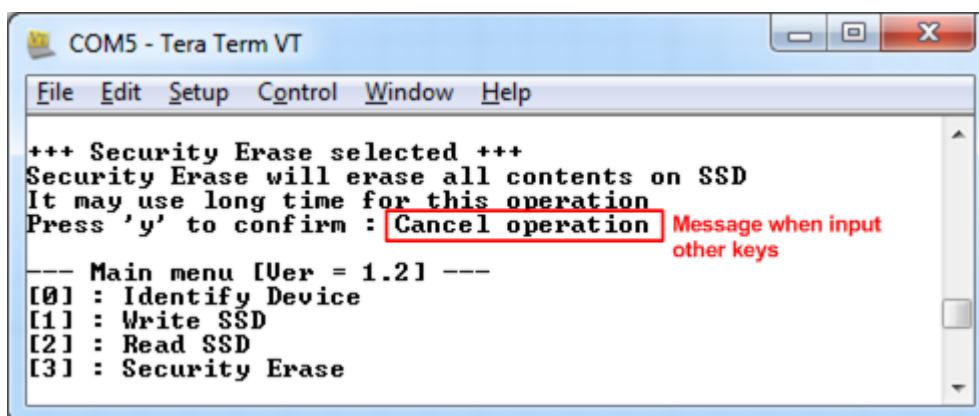


Figure 3-11 Cancel Security Erase command

4 Revision History

Revision	Date	Description
1.0	15-Oct-14	Initial version release
1.1	3-Apr-15	Add ZC706 support
1.2	30-Aug-16	Add CPU and support KCU105 board
1.3	29-Sep-16	Add Zynq Mini-ITX support
1.4	9-Nov-16	Add Security erase command and VC709 support
1.5	1-Aug-17	Add LFSR pattern
1.6	15-Jan-18	Add ZCU102 support
1.7	26-Apr-18	Add VCU118 support