

# FAT32-IP for SATA Demo Instruction

Rev1.1 6-Jul-23

1	En	vironment Setup	2
		emo setup	
3	Tes	st Menu	8
		Format Disk	
		Write File	
		Read File	
	3.4	Change File Size	
4	Rev	evision History	
•	1.0		



## FAT32-IP for SATA Demo Instruction

Rev1.1 6-Jul-23

This document describes the instruction to run FAT32-IP for SATA demo on FPGA development board by using AB09-FMCRAID board. The demo is designed to write and verify data with SATA-II or 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: KC705/ZC706
- 2) PC installing Xilinx programmer software (Vivado) and Serial console software such as HyperTerminal
- 3) AB09-FMCRAID board provided by Design Gateway
- 4) SATA-II or SATA-III device connecting to AB09 board
- 5) Xilinx Power adapter for FPGA board and ATX power supply for SATA device
- 6) micro USB cable for programming FPGA, connecting between FPGA board and PC
- 7) mini USB cable for Serial console, connecting between FPGA board and PC

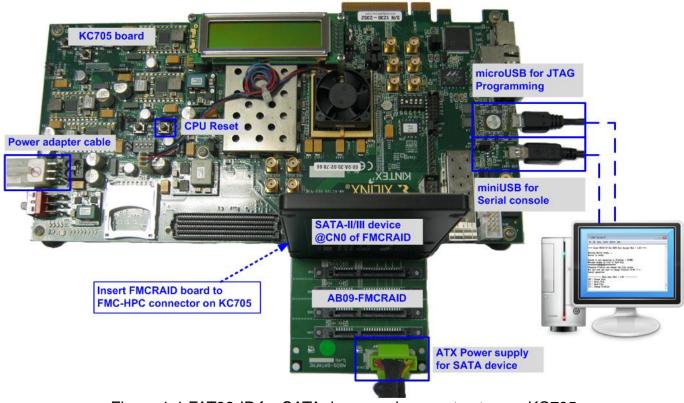
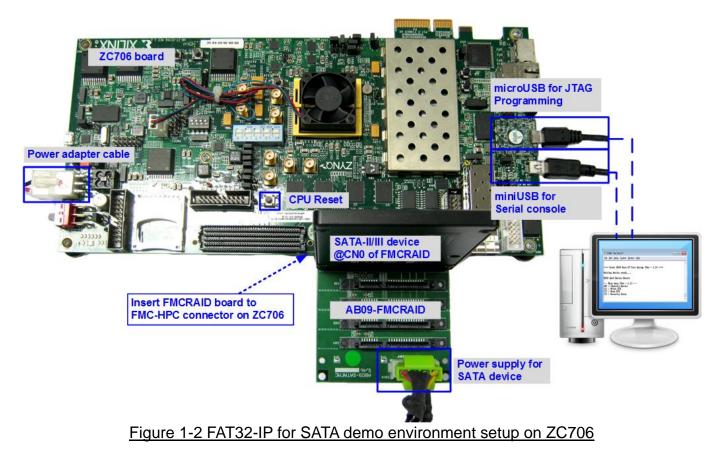


Figure 1-1 FAT32-IP for SATA demo environment setup on KC705







### 2 Demo setup

- 1) Power off system.
- 2) Set up board option for ZC706 board only,
  - i. Set SW11="00000" to configure PS from JTAG, as shown in Figure 2-1.
  - ii. Set SW4="01" to connect JTAG with USB-to-JTAG interface, as shown in Figure 2-2.

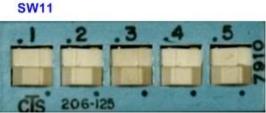


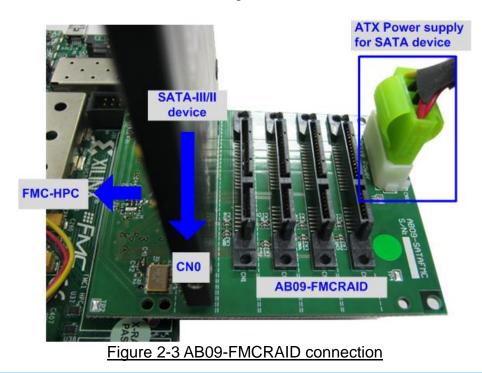
Figure 2-1 SW11 setting to configure PS from JTAG on ZC706 board



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

- 3) Connect the SATA-II/III device to the board by the following steps.
  - i. Connect AB09-FMCRAID board to FMC-HPC connector on Xilinx development board.
  - ii. Connect SATA-II/III device to CN0 on FMCRAID board.
  - iii. Connect ATX power to power connector on FMCRAID board.

The connection on AB09 is shown in Figure 2-3.





4) Connect micro USB cable between FPGA board and PC for JTAG programming, and connect mini USB cable between FPGA board and PC for Serial console.



Mini USB for Serial console Figure 2-4 USB cable connection

- 5) Power on FPGA board and power supply for SATA device.
- 6) Open Serial console such as TeraTerm, HyperTerminal. Set Baud rate=115,200 Data=8 bit Non-Parity Stop=1.
- 7) Download bit file or bat file to configure FPGA and firmware.
  - a) For KC705 board, use Vivado tool to download configuration file which includes CPU firmware, as shown in Figure 2-5.

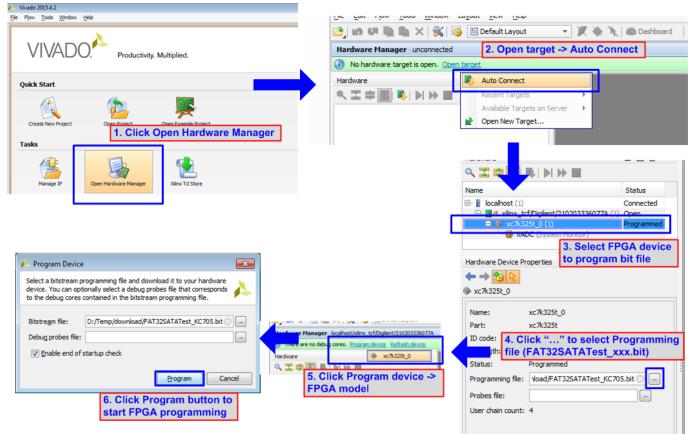


Figure 2-5 Download configuration file by Vivado tool for KC705



b) For ZC706 board, open Vivado TCL shell. Change current directory to download. Next, run FAT32SataTest\_ZC706.bat as shown in Figure 2-6.



Figure 2-6 Command script for download demo file to ZC706 by Vivado TCL Shell

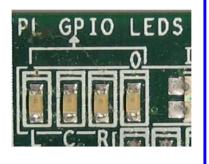
8) Check LED status on FPGA board. The description of LED is as follows.

<b>GPIO LED</b>	ON	OFF		
0	Normal operation	System is in reset condition		
1/R	System is busy	Idle status		
2/C	IP Error is detected	Normal operation		
3/L	Data verification fail	Normal operation		
Table 1 LED Definition				

Table 1 LED Definition

ZC706

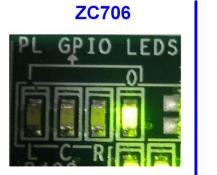
KC705



3 2 1 0 A I I I I I I I I I I I I I I I I I

Figure 2-7 4-bit LED Status for user output

9) After programming completely, LED[0] and LED[1] are ON to show that FAT32-IP is in initialization process. LED[1] changes to OFF after FAT32-IP completes initialization process and system is ready to receive command from user. Finally, file size setting menu is displayed.



KC705

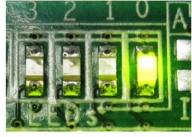


Figure 2-8 LED status after program configuration file and FAT32-IP initialization complete



10)In file size setting menu

- a) Input 'y' or 'Y' to set file size. Seven file sizes could be set in the submenu.
- b) If user input other values, default value will be used (32 MB).

After that, boot menu is displayed on the console, as shown in Figure 2-9 and Figure 2-10

🔟 COM10 - 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					
++++ Start FAT32-IP For SATA Test design [Ver = 1.1] Green : User Input Blue : Message to Use	r Ì				
Waiting IP initialization IP initialization complete Wait until FAT32-IP is not busy					
System is now operating at FileSize = 32[MB] Maximum number of file is 7629 File ************************************					
File size changed System is now operating at FileSize = 256[MB] Maximum number of file is 953 File					
Please format disk before use Press 'y' to confirm format disk =>y 'Y' to format disk Format Disk complete					
Main menu [Ver = 1.1] [0] : Format Disk [1] : Write File [2] : Read File [3] : Change FileSize	~				
Figure 2-9 Main menu when confirm to set file size					

💻 COM10 - Tera Term VT Х File Edit Setup Control Window <u>H</u>elp Green : User Input ++++ Start FAT32-IP For SATA Test design [Ver = 1.1] Blue : Message to User Waiting IP initialization IP initialization complete Wait until FAT32-IP is not busy System is now operating at FileSize = 32[MB] Maximum number of file is 7629 File \* Changing FileSize may damage the file system Are you sure you want to change FileSize [Y/N] =>n (not 'y') to use Cancel operation Input other values default file size --- Main menu [Ver = 1.1] ---[0] : Format Disk [1] : Write File [2] : Read File Main menu to select the command [3] : Change FileSize Š Figure 2-10 Main menu when skip to set file size

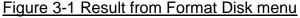


#### 3 Test Menu

#### 3.1 Format Disk

Select '0' to send Format disk command to FAT32-IP. When operation is completed, the system shows current operation file size and maximum file, as shown in Figure 3-1.

💆 COM10 - Tera Term VT 🦳 🗌	$\times$					
<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						
+++ Format Disk selected +++ Press 'y' to confirm format disk =>y 'Y' to confirm format disk Format Disk complete System is now operating at FileSize = 32[MB] Current file size and Maximum number of file is 7629 File maximum file						



It is recommended to run this menu when one of following conditions is found. 1) The disk is formatted by other system which is not FAT32-IP.

- 2) The disk is not formatted by FAT32-IP.
- 3) User needs to change file size value in the test.
- 4) User needs to delete file in the disk.



#### 3.2 Write File

Select '1' to send Write file command to FAT32-IP. After that, users are asked to set created time of file. Users can input 'y' or 'Y' to change created time or input others to use default created time.

<u>Note</u>: Default created time after system boot up is 22 Sep 2017, 15:31:18. When user sets new created time in Write file menu, the value is used to be created time of current Write file operation. The default created time is updated by the latest created time.

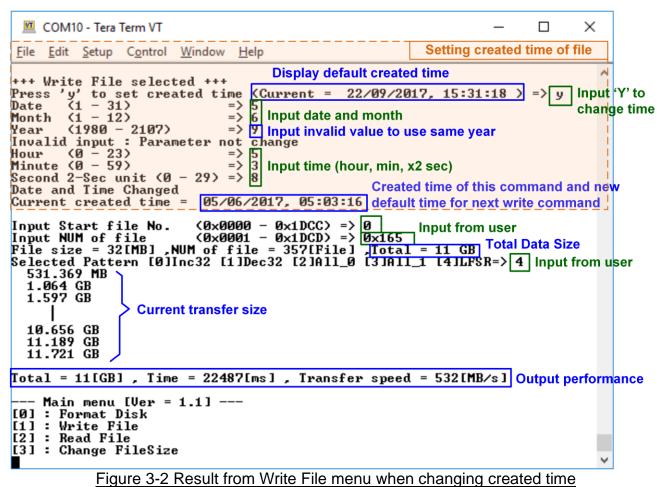


Figure 3-2 shows the example when running Write File menu with setting created time of file.

There are six values to set, i.e.

- a) Date Input created date of file as decimal unit. Valid range is 1 31.
- b) Month Input created month of file as decimal unit. Valid range is 1 12.
- c) Year Input created year of file as decimal unit. Valid range is 1980 2107.
- d) Hour Input created hour of file as decimal unit. Valid range is 0 23.
- e) Minute Input created minute of file as decimal unit. Valid range is 0 59.
- f) Second Input created x2 second of file as decimal unit. Valid range is 0 29. The created second is equal to input value x 2.

If the input is invalid, the parameter will not change and default value will be used. Only parameter that valid is changed. Finally the new created time is displayed on the console.



After setting created time process is done, three parameters are required to write file, i.e.

- 1) Start file No Input file name of the first file as a number. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
- 2) NUM of file Input total number of file to transfer. The first file name is FILE<Start file No>.BIN and the last file name is FILE<Start file No + NUM of file - 1>.BIN. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.

If "Start file No" and "NUM of file" are valid, total data size (calculated from File size x NUM of file) will be displayed on the console.

3) Test pattern – Select pattern of test data in written file. Five patterns can be set, i.e. 32-bit increment, 32-bit decrement, all 0, all 1, and 32-bit LFSR counter.

If all inputs are valid, the operation will be started. During writing file, current transfer size is displayed on the console to show that system still be alive every second. Finally, test performance, total size, and total time usage are displayed on the console as test result.

Figure 3-3 shows the example to Write File without setting created date. In this example, default created time is updated by the value in previous Write File test. It is recommended to input "Start file No" value by the next value which continues from previous test. Previous test in Figure 3-2 writes FILE0000.BIN – FILE0164.BIN, so "Start file No" should be 0x165 (0x164 + 1).

🔟 COM10 - Tera Term VT	– 🗆 ×
<u>Eile Edit Setup Cont</u> Default created time = the latest setting	y value - Input other keys ('n') to ski created time of file setting
+++ Write File selected +++ Press 'y' to set created time (Current = 05/06/2 Input Start file No. (0x0000 - 0x1DCC) => 0x165 Input NUM of file (0x0001 - 0x1C68) => 0x200 File size = 32[MB],NUM of file = 512[File],Tota Selected Pattern [0]Inc32 [1]Dec32 [2]All_0 [3]Al 531.359 MB	
1.064 GB 1.597 GB	Next file name from previous Write file command
15.985 GB 16.518 GB 17.050 GB	
otal = 17[GB] , Time = 32249[ms] , Transfer spee	ed = 532[MB/s] Output performanc
Main menu [Ver = 1.1] [0] : Format Disk [1] : Write File [2] : Read File [3] : Change FileSize	~
Figure 3-3 Result from Write File menu when	n changing created time



After completing Write File command, user can plug the disk to PC which supports FAT32 File format. On PC, the new disk "DG\_FAT32" is detected. Inside the disk, FILE0000.BIN – FILE<Last file No>.BIN are stored. Also, File size and modified date must be matched to the value set in Write File test. Four numbers in File name is file number in hexadecimal unit.

<u>Note</u>: When connecting disk to PC, please do not write or modify a file in the disk. If file in disk has some modification, disk must be formatted by FAT32-IP.

🗸 🗢 🗢 Compute	r 🕨 DG_FAT32 (E:)	Modified Date = Creat	ed Time in th	ne example	
Organize 👻 📄 1 <sup>st</sup> Fil	e name = FILE0000.BIN	/	File	size = 32 MB	(
🚖 Favorites	Name	Date modified	Туре	Size	T
🧮 Desktop	FILE0000.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	1
鷆 Downloads	FILE000A.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
🗓 Recent Places	FILE000B.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
	FILE000C.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
词 Libraries	FILE000D.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
Documents	FILE000E.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
🚽 Music	FILE000F.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
📔 Pictures	FILE00A0.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
🛃 Videos	FILE00A1.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
	FILE00A2.BIN	6/5/2017 5:03 AM	BIN File	32,768 KB	
FILE0000.BIN D	Date modified: 6/5/2017 5 Size: 32.0 MB	:03 AM Date cr	reated: 6/5/20	017 5:03 AM	

Figure 3-4 Example test files written by Write File command



•	Test data of 32-bit increment pattern	Test data of 32-bit LFSR pattern
	←64-bit header of each sector→	←64-bit header of each sector→
	bit unique value = Test data ne x File size) + offset 0x00000000 (32-bit increment)	Test data32-bit unique value0x00000000(32-bit LFSR)
Offset	0 1 4 2 3 4 5 6 7 8 9 A B C D E	F 0 1 2 3 4 5 6 7 8 9 A B C D E F
00000000		
00000010		
00000020		0 49 00 00 00 92 00 00 00 24 01 00 00 49 02 00 00
00000030		0 92 04 00 00 24 09 00 00 49 12 00 00 92 24 00 00
00000040	10 00 00 00 11 00 00 00 12 00 00 00 13 00 00 0	00 24 49 00 00 49 92 00 00 92 24 01 00 24 49 02 00
00000050	14 00 00 00 15 00 00 00 16 00 00 00 17 00 00 0	00 49 92 04 00 92 24 09 00 24 49 12 00 49 92 24 00
00000060	18 00 00 00 19 00 00 00 1A 00 00 00 1B 00 00 0	DO 93 24 49 00 27 49 92 00 4F 92 24 01 9E 24 49 02
00000070	1C 00 00 00 1D 00 00 00 1E 00 00 00 1F 00 00 0	DO 3C 49 92 04 79 92 24 09 F3 24 49 12 E7 49 92 24
00000080	20 00 00 00 21 00 00 00 22 00 00 00 23 00 00 0	DO CF 93 24 49 9E 27 49 92 3D 4F 92 24 7A 9E 24 49
00000090	24 00 00 00 25 00 00 00 26 00 00 00 27 00 00 0	DO F5 3C 49 92 EB 79 92 24 D7 F3 24 49 AE E7 49 92
0A000000		DO 5D CF 93 24 BA 9E 27 49 75 3D 4F 92 EB 7A 9E 24
000000B0		DO D7 F5 3C 49 AE EB 79 92 5C D7 F3 24 B8 AE E7 49
000000000		00 70 5D CF 93 E0 BA 9E 27 C1 75 3D 4F 83 EB 7A 9E
000000D0		00 07 D7 F5 3C 0E AE EB 79 1D 5C D7 F3 3B B8 AE E7
000000E0		00 77 70 5D CF EE EO BA 9E DC C1 75 3D B8 83 EB 7A
000000F0		00 70 07 D7 F5 E0 0E AE EB C1 1D 5C D7 83 3B B8 AE
00000100		00 07 77 70 5D 0E EE E0 BA 1C DC C1 75 39 B8 83 EB
00000110		00 73 70 07 D7 E6 E0 0E AE CD C1 1D 5C 9A 83 3B B8
00000120		00 34 07 77 70 68 0E EE E0 D1 1C DC C1 A3 39 B8 83
00000130		00 47 73 70 07 8E E6 E0 0E 1D CD C1 1D 3A 9A 83 3B
00000140		00 74 34 07 77 E9 68 0E EE D3 D1 1C DC A6 A3 39 B8
00000150		00 4C 47 73 70 98 8E E6 E0 31 1D CD C1 63 3A 9A 83
00000160		00 C6 74 34 07 8D E9 68 0E 1B D3 D1 1C 37 A6 A3 39 00 6E 4C 47 73 DC 98 8E E6 B8 31 1D CD 70 63 3A 9A
00000170		00 6E 4C 47 73 DC 98 8E E6 B8 31 1D CD 70 63 3A 9A 00 E1 C6 74 34 C3 8D E9 68 86 1B D3 D1 0D 37 A6 A3
00000190		00 1A 6E 4C 47 34 DC 98 8E 68 B8 31 1D D0 70 63 3A
000001A0		00 A0 E1 C6 74 41 C3 8D E9 83 86 1B D3 06 0D 37 A6
000001B0		00 OC 1A 6E 4C 18 34 DC 98 30 68 B8 31 60 DO 70 63
000001C0		DO CO AO E1 C6 81 41 C3 8D 03 83 86 1B 07 06 0D 37
000001D0		00 OF OC 1A 6E 1F 18 34 DC 3F 30 68 B8 7F 60 D0 70
000001E0		DO FF CO AO E1 FF 81 41 C3 FE 03 83 86 FD 07 06 0D
000001F0		DO FA OF OC 1A F4 1F 18 34 E9 3F 30 68 D3 7F 60 D0
00000200		
00000210		
00000220		
	64-bit header	64-bit header
	Figure 3-5 Test data in sector#0 - #1 of F	II E0000 BIN by increment/LESR pattern
	$\frac{1}{1}$ guile 0.0 Test data in Sector#0 - #1.011	

ī

Figure 3-5 shows the example of the data in FILE0000.BIN when writing data by increment (left window) and LFSR pattern (right window). 64-bit header is inserted every 512-byte test data. The header is unique value in one disk. The first 32-bit header is calculated by (file name x file size) + offset in the file. The remaining 32-bit data is 0. The offset value is started from 0 and increased by 1 every 512-byte data. The remaining data of one sector is test data which the pattern is defined from user.



Figure 3-6 – Figure 3-8 show error messages when user input is invalid. "Invalid input" message is displayed on the console, and then returns to main menu to receive new command.

COM10 - Tera Term VT  $\times$ File Edit Setup Control Window <u>H</u>elp ~ +++ Write File selected +++ Press 'y' to set created time (Current = 05/06/2017, 05:03:16 ) => n Input Start file No. (0x0000 - 0x1DCC) => 0xFFFF Out-of-range Start file No. Invalid input Error message Main menu [Ver = 1.1] ---[0] : Format Disk [1] : Write File [2] : Read File [3] : Change FileSize Figure 3-6 Invalid Start file no. input COM10 - Tera Term VT × File Edit Setup Control Window Help ~ +++ Write File selected +++ Press 'y' to set created time (Current = 05/06/2017, 05:03:16 > => n Input Start file No. ( $0 \times 0000 - 0 \times 10$ CC> => 0 Input NUM of file ( $0 \times 0001 - 0 \times 10$ CD> => 0 Out-of-range NUM of file Invalid input Error message --- Main menu [Ver = 1.1] ---[0] : Format Disk [1] : Write File [2] : Read File [3] : Change FileSize Figure 3-7 Invalid Num of file input × 💻 COM10 - Tera Term VT File Edit Setup Control Window <u>H</u>elp ~ +++ Write File selected +++ Press 'y' to set created time (Current = 05/06/2017, 05:03:16 ) => n Input Start file No. (0x0000 - 0x1DCC) => 0 Input NUM of file (0x0001 - 0x1DCD) => 1 File size = 32[MB],NUM of file = 1[File],Total = 32 MB Selected Pattern [0]Inc32 [1]Dec32 [2]All\_0 [3]All\_1 [4]LFSR=>9 Invalid input Error message **Out-of-range pattern** --- Main menu [Ver = 1.1] ---[0] : Format Disk [1] : Write File

#### Figure 3-8 Invalid Test pattern input

[2] : Read File [3] : Change FileSize



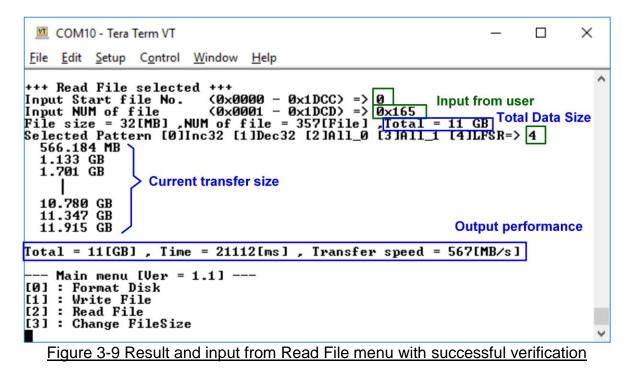
#### 3.3 Read File

Select '2' to send Read file command to FAT32-IP. Similar to Write File menu, three parameters are required to read file, i.e.

- 1) Start file No Input a number of the first read file. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
- 2) NUM of file Input total number of file to transfer. The first file name is FILE<Start file No>.BIN and the last file name is FILE<Start file No + NUM of file - 1>.BIN. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.

If "Start file No" and "NUM of file" are valid, total data size (calculated from File size x NUM of file) will be displayed on the console.

3) Select pattern – Select test pattern to verify data in the file. Test pattern must be matched with the test pattern using in Write File menu. Five patterns can be set, i.e. 32-bit increment, 32-bit decrement, all 0, all 1, and 32-bit LFSR counter.



Similar to Write File menu, if all inputs are valid, the operation will be started. During reading file, current transfer size is displayed on the console to show that system still be alive every second. Finally, test performance, total size, and total time usage are displayed on the console as test result. "Invalid input" will be displayed if some inputs are out-of-range.

Figure 3-10 and Figure 3-11 show the error message when data verification is failed. "Verify fail" message is displayed with the first file name which has error, error address of a file, expected data, and read data. The operation is cancelled when user presses any keys. If Read file is completed, output performance from Read file process will be displayed.



💻 COM10 - Tera Term VT	– n ×				
	Wrong pattern				
<u>File Edit Setup Control Window Help</u>	(In the example, FILE0000-0164 is written by				
+++ Read File selected +++	LFSR, but FILE0165 is written by increment)				
Input Start file No. (0x0000 - 0)	<1DCC) => 0				
Input Start IIIE No. $(0 \times 0000 = 0)$	(1000) = 0 (300)				
Input NUM of file (0x0001 - 0x File size = 32[MB] .NUM of file = 3	768[Fi]a] Total = 24 CB				
Selected Pattern [0]Inc32 [1]Dec32	[21011 0 [31011 1 [4]]ESR=> 4				
1.133 GB					
1.701 GB					
10.780 GB					
11.347 GB					
11.915 GB Verify fail mess	200				
verny fail files	sage				
Verify fail					
1st Error at File name = FILE0165.J	bin				
Byte Address = 0x0000008	3				
Expect Data = 0x02CA0000	0				
Read Data = $0 \times B2800002$	2				
Press any key to cancel operation					
12.482 GB					
13.050 GB					
13.617 GB					
Verify fail without					
and a second second second second					
24.966 GB					
25.533 GB	Output performance				
Total = 25[GB] , Time = 45[s] , Tra					
10tal - 251GDJ , 11Me - 451SJ , 1ra	ansfer speed - 567(mb/s)				
Main menu [Ver = 1.1]					
[0] : Format Disk					
[1] : Write File					
[2] : Read File					
[3] : Change FileSize					
151 · Ondinge LITEOISE	~				
Figure 3-10 Data verification i	<u>s failed but wait until read complete</u>				



<pre>M COM10 - Tera Term VT - C X File Edit Setup Control Window Help +++ Read File selected +++ Input Start file No. (0x0000 - 0x1DCC) =&gt; 0 Input NUM of file (0x0001 - 0x1DCD) =&gt; 0x300 File size = 32[MB],NUM of file = 768[File],Total = 24 GB Selected Pattern [0]Inc32 [1]Dec32 [2]All_0 [3]All_1 [4]LFSR=&gt; 4 566.183 MB 1.133 GB 1.701 GB </pre>	<b>^</b>
+++ Read File selected +++ Input Start file No. (0x0000 - 0x1DCC) => 0 Input NUM of file (0x0001 - 0x1DCD) => 0x300 File size = 32[MB] ,NUM of file = 768[File] ,Total = 24 GB Selected Pattern [0]Inc32 [1]Dec32 [2]All_0 [3]All_1 [4]LFSR=> 4 566.183 MB 1.133 GB	^
Input Start file No. (0x0000 - 0x1DCC) => 0 Input NUM of file (0x0001 - 0x1DCD) => 0x300 File size = 32[MB],NUM of file = 768[File],Total = 24 GB Selected Pattern [0]Inc32 [1]Dec32 [2]All_0 [3]All_1 [4]LFSR=> 4 566.183 MB 1.133 GB	~
10.780 GB 11.347 GB 11.914 GB Verify fail message	
Verify fail 1st Error at File name = FILE0165.bin Byte Address = 0x0000008 Expect Data = 0x02CA0000 Read Data = 0xB2800002	
Press any key to cancel operation 12.482 GB	
Operation is cancelled Verify fail with	
Please reset system before start new test cancel operation	
Main menu [Ver = 1.1] [0] : Format Disk [1] : Write File [2] : Read File [3] : Change FileSize	~
Figure 3-11 Data verification is failed and press any keys to cancel operation	

D·G

dg\_fat32ip\_sata\_instruction\_en

#### 3.4 Change File Size

Select '3' to change file size. After selecting menu, current file size and warning message are displayed on the console. Then, user input 'y' or 'Y' to confirm to change file size or input others to cancel the operation.

💻 COM10 - Tera Term VT			_		$\times$
<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				
+++ Change FileSize sele	cted +++		Setting	new file	size∧
System is now operating Maximum number of file i ************************************	s 7629 File	Current fil	e size		
Changing FileSize may da Are you sure you want to	mage the file system				
Enter FileSize [0] : 32 MB [1] : 64 MB [2] : 128 MB [3] : 256 MB [4] : 512 MB [5] : 1024 MB [6] : 2048 MB => 2	Selecting file size menu	cn	lange file	SIZE	
Fil <u>e size changed</u> System is now operating Maximum number of file i	at FileSize = 128[MB] s 1907 File		ration file n number		d
Please format disk befor Press 'y' to confirm for Format Disk complete	e use mat disk => y 'Y' to con	firm to forn	nat disk	Format	disk
Main menu [Ver = 1.1 [0] : Format Disk [1] : Write File [2] : Read File [3] : Change FileSize	]				~
Figure 3-12 Result from Change FileSize command					

After confirming to change file size, the selecting file size menu is displayed. There are seven file sizes to select, i.e. 32 MB, 64 MB, 128 MB, 256 MB, 512 MB, 1 GB, and 2 GB. After setting new file size, updated file size and maximum number of file are displayed on the

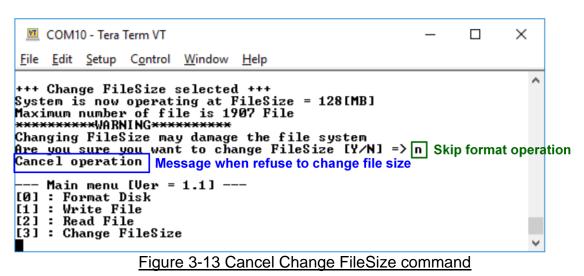
After that, a request to format disk message is asked. User inputs 'y' or 'Y' to accept format disk operation or inputs other keys to refuse.

<u>Note</u>: Changing file size may damage file system. Therefore, user should format disk after complete file size changing menu to delete all files in the device.

The example to refuse changing file size by input others (not 'y') is shown in Figure 3-13.

console.







### 4 Revision History

Revision	Date	Description
1.0	4-Oct-17	Initial version release
1.1	5-Oct-18	Update firmware version and description