# exFAT IP for NVMe Demo Instruction

Rev1.3    20-Apr-20

This document describes the instruction to run exFAT IP for NVMe demo on FPGA development board by using the PCIe adapter board (AB16-PCIeXOVR board or AB18-PCIeX16 board). The demo is designed to write and verify data with NVMe SSD. User controls the test operation through NiosII command shell.

## 1   Environment Setup

To run the demo on FPGA development board, please prepare following environment.
1) FPGA development board: Intel Arria10 GX development board
2) PC installing Quartus programmer and NiosII command shell software
3) The PCIe adapter board (AB18-PCIeX16/AB16-PCIeXOVR adapter board) provided by Design Gateway
   https://dgway.com/ABseries_E.html
4) Intel FPGA power adapter for FPGA board
5) ATX power supply for PCIe adapter board
6) NVMe SSD connecting to PCIe female connector on AB16/AB18 board
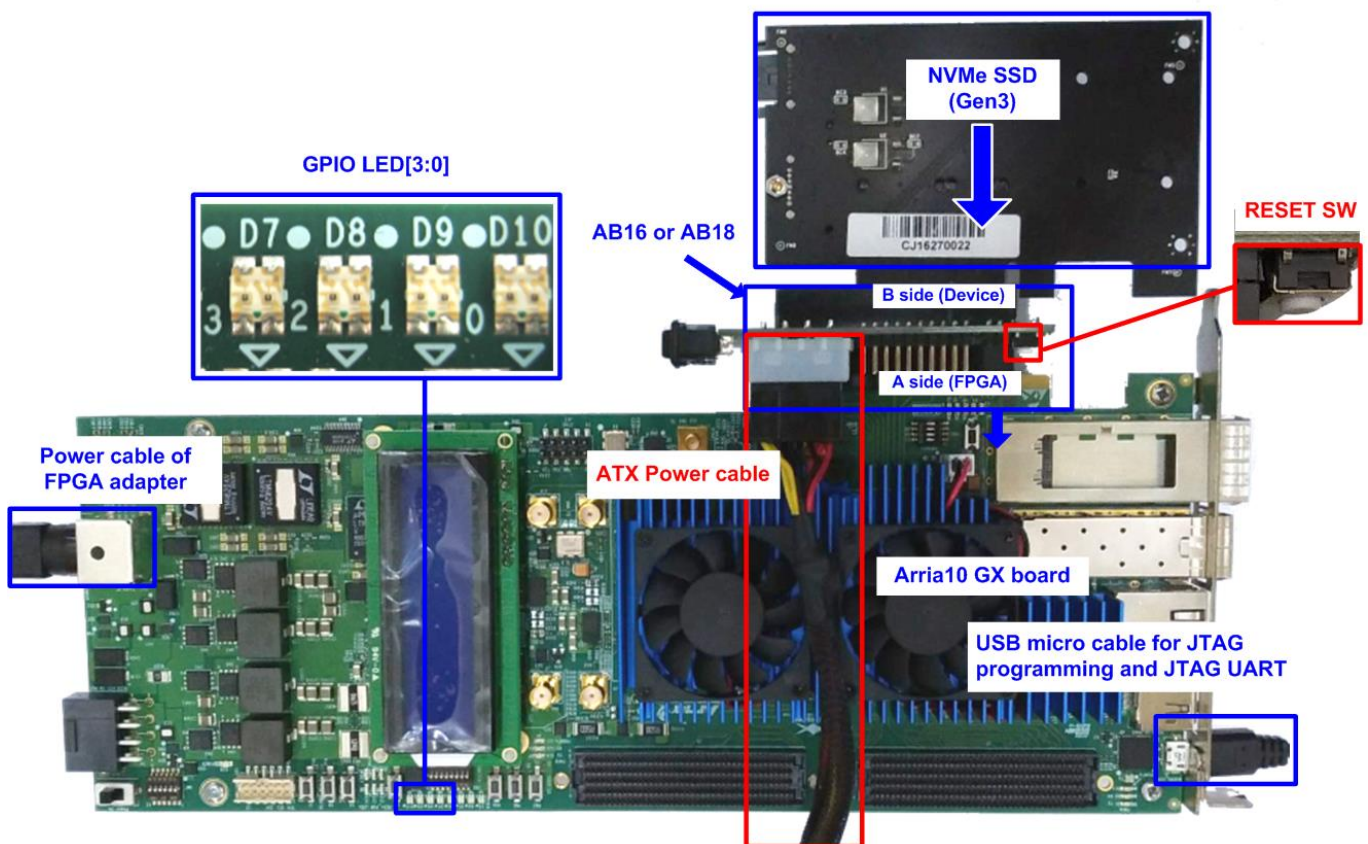7) micro USB cable for programming FPGA and NiosII command shell, connecting between FPGA board and PC



Figure 1-1 exFAT IP for NVMe demo environment setup on Arria10 GX board

## 2 Demo setup

1) Power off system.
2) Insert NVMe SSD to AB16/AB18 and connect AB16/AB18 to FPGA board.
    i.   Confirm that two mini jumpers are inserted at J5 connector on AB16/AB18.
    ii.  Connect ATX power to ATX power connector on AB16/AB18 board.
    iii. Connect FPGA side (A-side) on AB16/AB18 board to PCIe connector on FPGA board.
    iv.  Connect NVMe SSD to device side (B-side) on AB16/AB18 board.
    **Warning: Please confirm that NVMe SSD is inserted in the correct side of AB18/AB16 (B-side, not A-side) before power on system.**
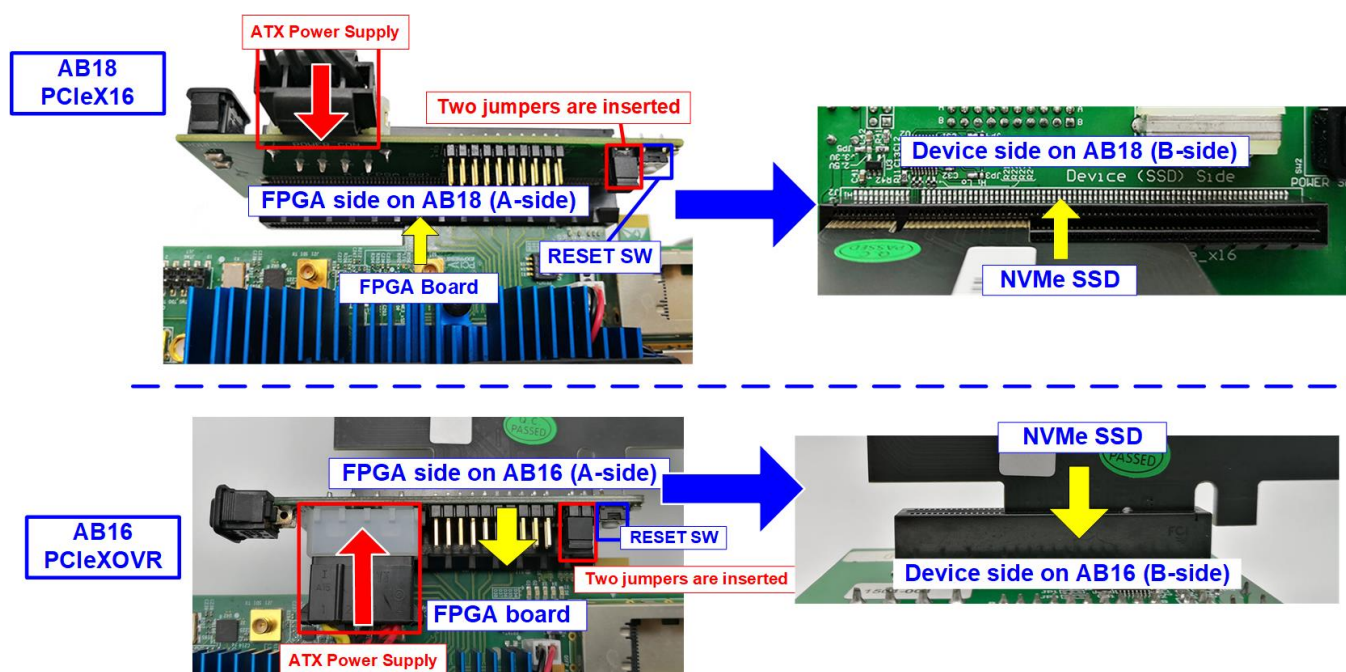


Figure 2-1 Connect adapter board to NVMe SSD and FPGA board

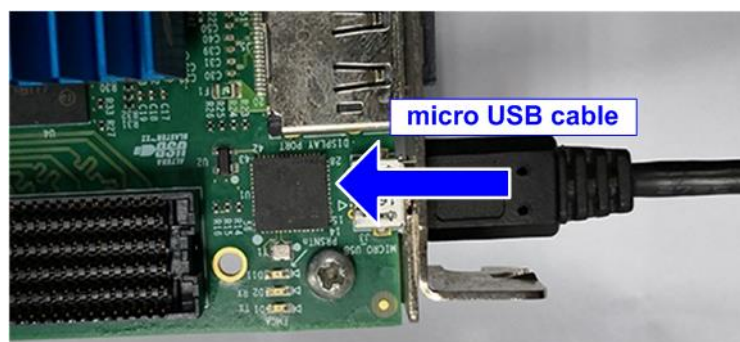3) Connect micro USB cable from FPGA board to PC for JTAG programming and JTAG UART.



Figure 2-2 USB cable connection

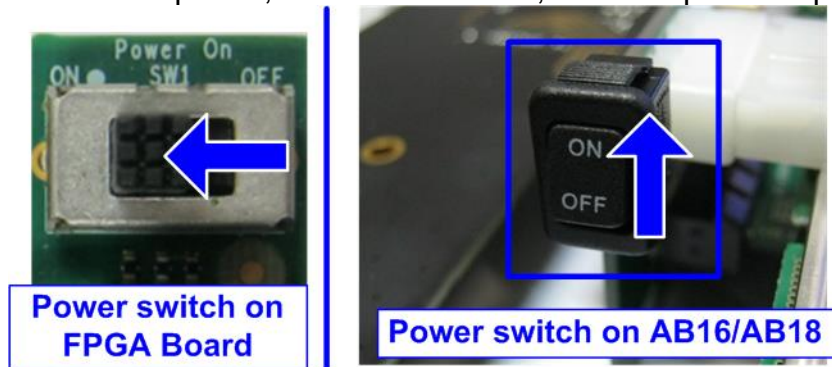4) Power on FPGA development, AB16/AB18 board, and ATX power supply



Figure 2-3 on AB16/AB18 and FPGA board

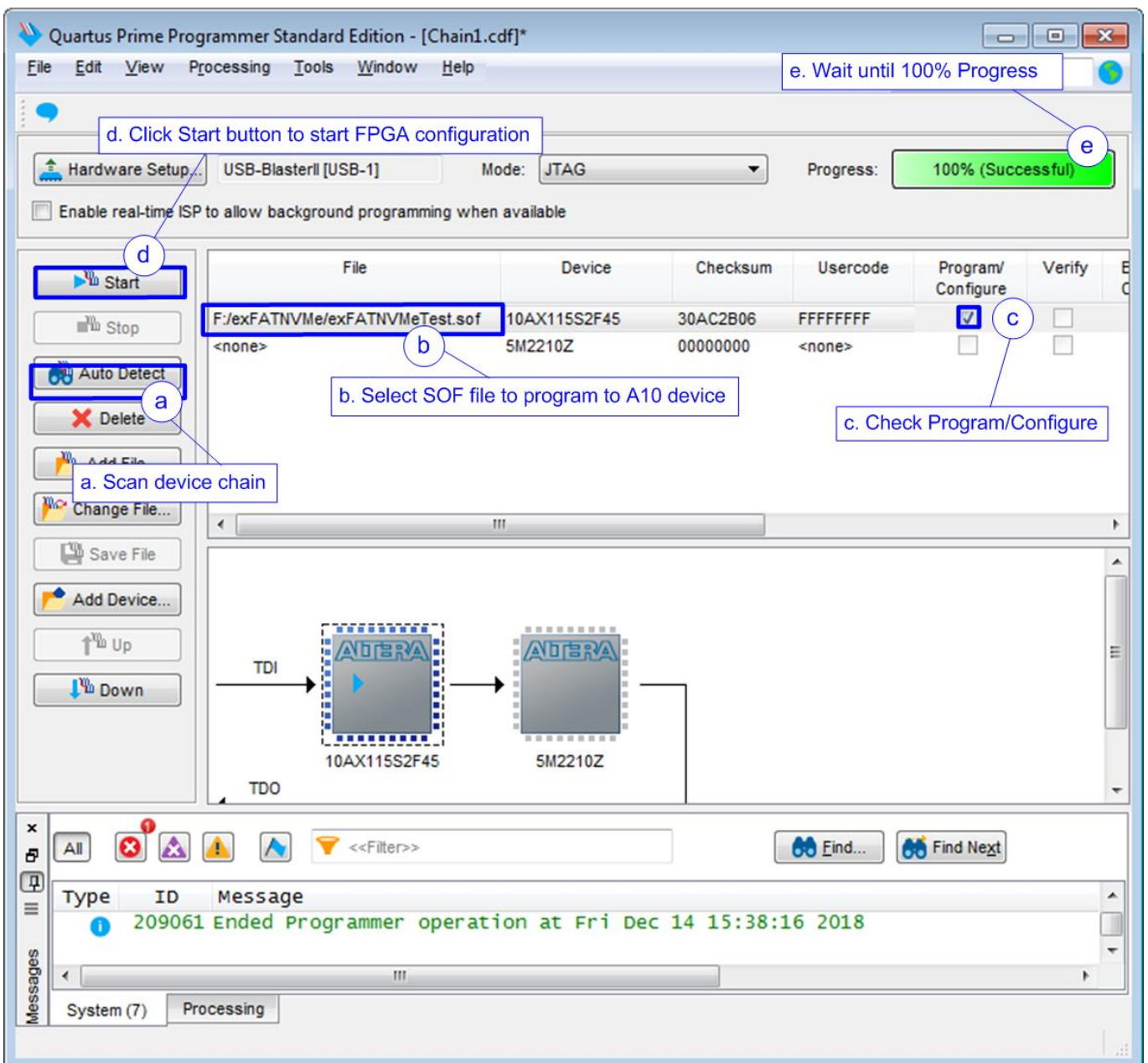5) Use Quartus Programmer to program "exFATNVMeTest.sof" file, as shown in Figure 2-4



Figure 2-4 Programmed by Quartus Programmer

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

| GPIO LED | ON | OFF |
|---|---|---|
| 0 | Normal operation | System is in reset condition |
| 1 | System is busy | Idle status |
| 2 | Error detect | Normal operation |
| 3 | Data verification fail | Normal operation |

Table 2-1 LED Definition

7) After programming complete, LED[0] and LED[1] are ON during running initialization process. Then, LED[1] changes to OFF after finishing exFAT IP initialization.
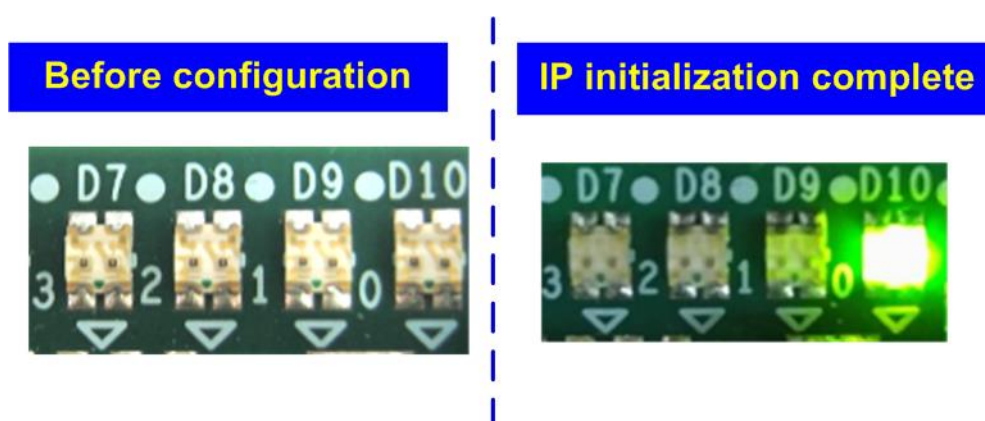


Figure 2-5 LED status

8) Open NiosII Command Shell and run nios2-terminal command. Boot message is displayed.
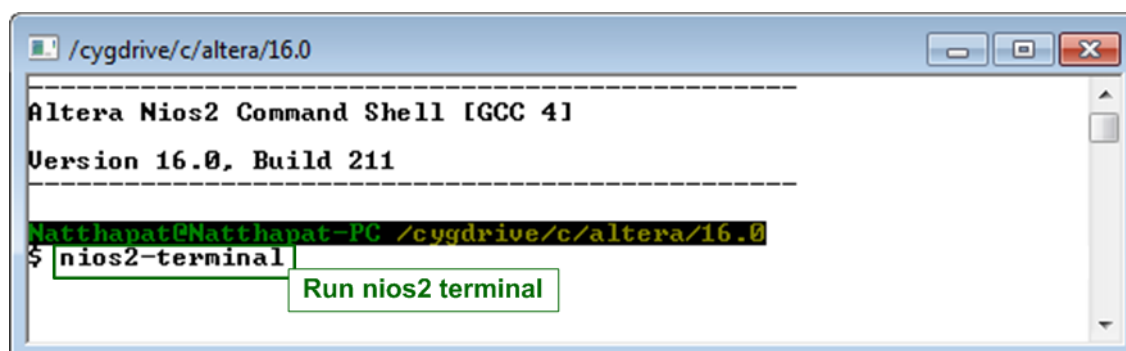


Figure 2-6 NiosII Command Shell

9) On the console, the message is displayed to show current status as follows
   - "Waiting IP initialization" is displayed during initialization sequence.
   - "IP initialization complete" is displayed when IP is idle.
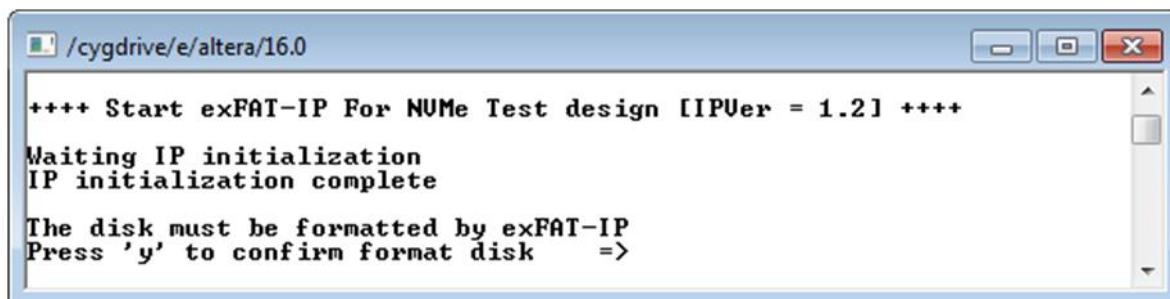   - Before running the test, "The disk must be formatted by exFAT-IP" is displayed.



Figure 2-7 Message when IP runs initialization process

For the new disk which has never been formatted by exFAT-IP, user must enter 'y' to format the disk. Otherwise, user enters other keys to skip format process.

After running Format command, all data in the disk are deleted. The example to boot system with and without Format command is shown in Figure 2-8. The details to run Format operation is described in topic 3.1 (Format Disk).
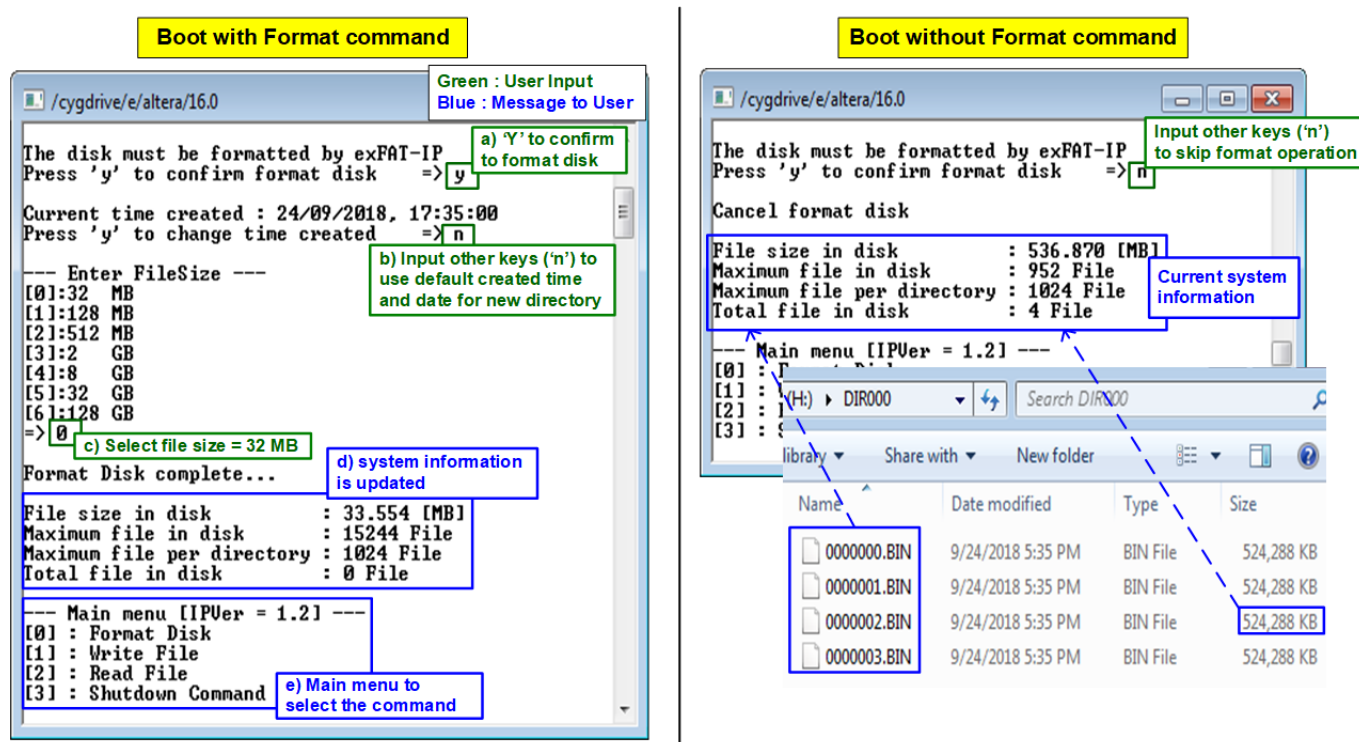


Figure 2-8 System information after initialization process

After that, the system information in the disk are displayed, i.e. current file size, maximum file in the disk, maximum file per directory, and total file in the disk.

If the disk is not empty, total file in the disk is not equal to 0. User can continue the test by writing additional file with the same file size or reading file to verify the data in the disk.

# 3   Test Menu

## 3.1   Format Disk

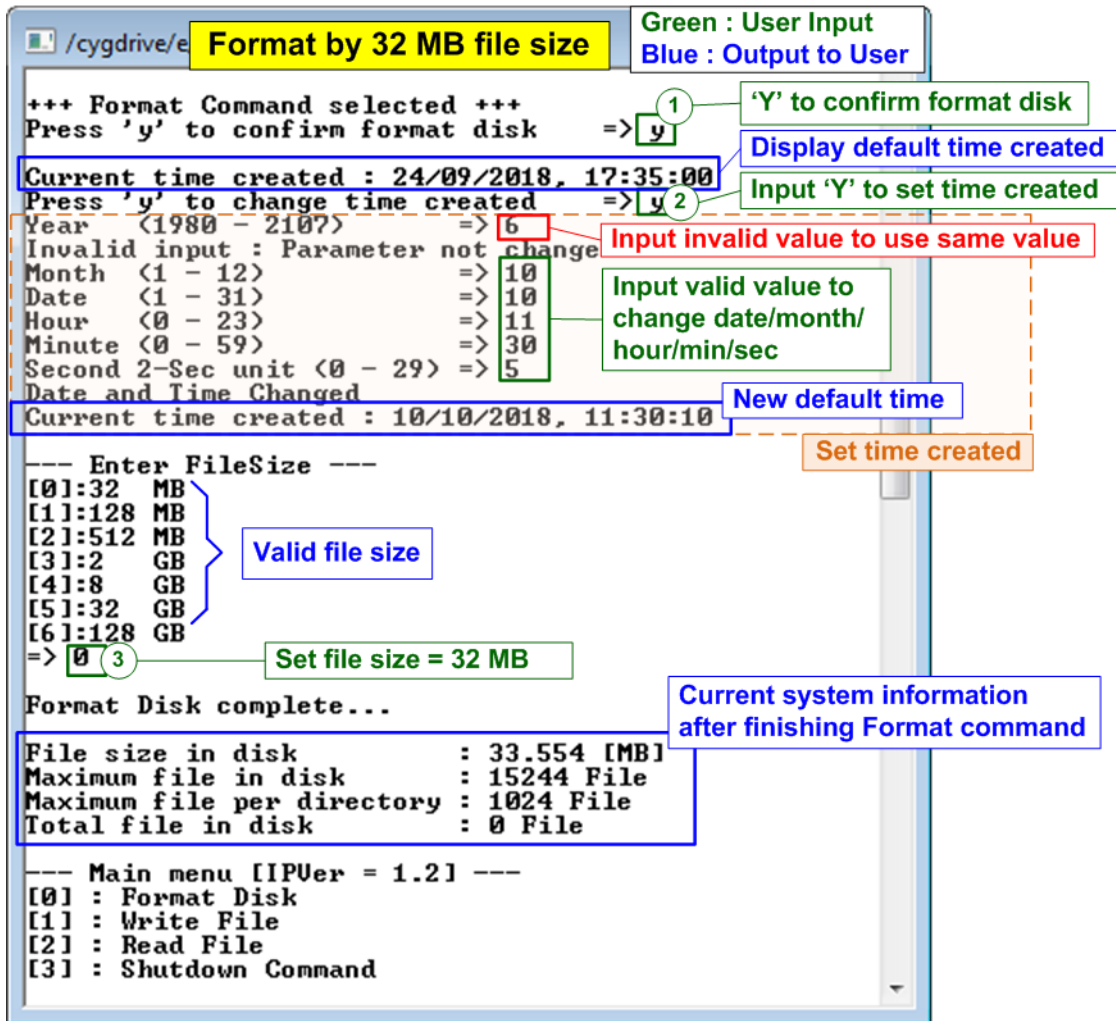Select '0' to send Format disk command to exFAT IP.  The step to run Format command is as follows.



Figure 3-1 Format Disk menu when updating the created time

1) Input 'y' to confirm Format operation or input other keys to cancel the operation. When Format is run, the default value of created date and time is displayed. This value is applied to be created date and time of empty directories, created by Format command.

2) Input 'y' to change created date and time or input other keys to use the default value. There are six parameters to set created date and time for empty directories, i.e.
   a)  Year        – Year of created date. Valid range is 1980 – 2107.
   b)  Month       – Month of created date. Valid range is 1 – 12.
   c)  Date        – Date of created date. Valid range is 1 – 31.
   d)  Hour        – Hour of created time. Valid range is 0 – 23.
   e)  Minute      – Minute of created time. Valid range is 0 – 59.
   f)  Second      – x2 Second of created time. Valid range is 0 – 29.

The input is received as decimal unit. User adds "0x" to be a prefix for hexadecimal unit. If the input is invalid, the parameter does not change (the same value is used). Only the input in the valid range is applied to update the parameter.

As shown in Figure 3-1, year parameter is invalid, so the value does not change (2018 is applied). Month, date, hour, minute, and second are valid, so these parameters are updated to the new default value. After that, "Date and Time changed" and the new created time and date are displayed on the console.
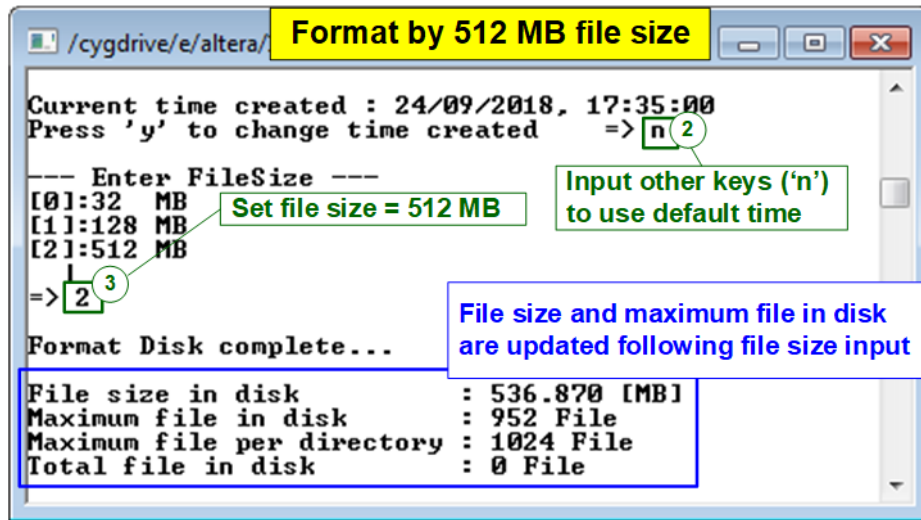


Figure 3-2 Format Disk menu when using the default created time

Figure 3-2 shows the example when running Format command without updating created time.

3) Select file size of the disk which is listed on the console. After receiving file size from user, the IP starts Format operation.

"Format Disk complete" and the updated system information are displayed after finishing Format command.

Figure 3-1 and Figure 3-2 sets the different file size on the same disk capacity. Maximum file in the disk is reduced when file size is increased.

If file size is invalid, the operation is cancelled, as shown in Figure 3-3

*Note: Default created time after system boot up is 24 Sep 2018, 17:35:00. When user sets the new created time in Format menu or Write file menu, the new value is applied to be the new default value.*
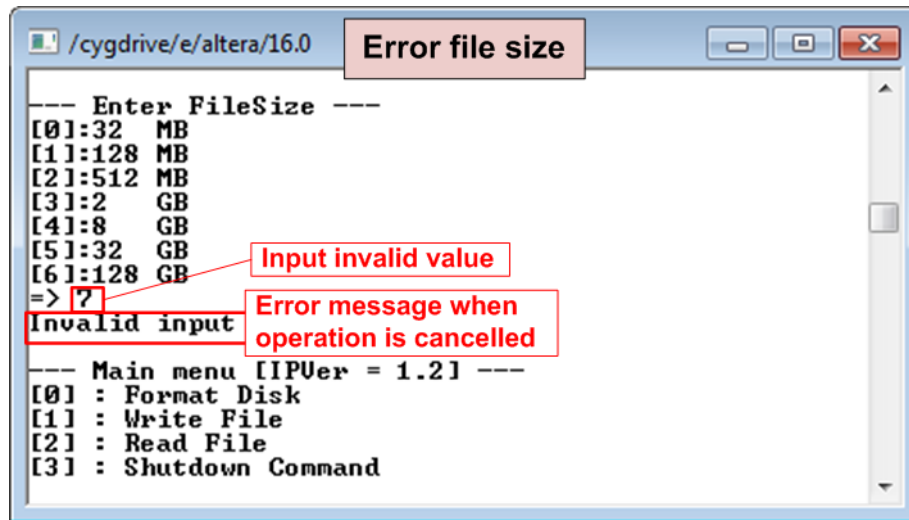
Figure 3-3 Result from Format Disk menu when select invalid file size

When the disk formatted by exFAT IP is connected to PC, DG_exFAT drive is detected with 512 empty directories (DIR000<1st directory> - DIR1FF<512th directory>). Modified date of the empty directories is equal to the created date setting in the test.

*Note: When connecting disk to PC, please do not create, write, or modify data in the disk. If the disk has some modification, the disk must be re-formatted by exFAT IP.*
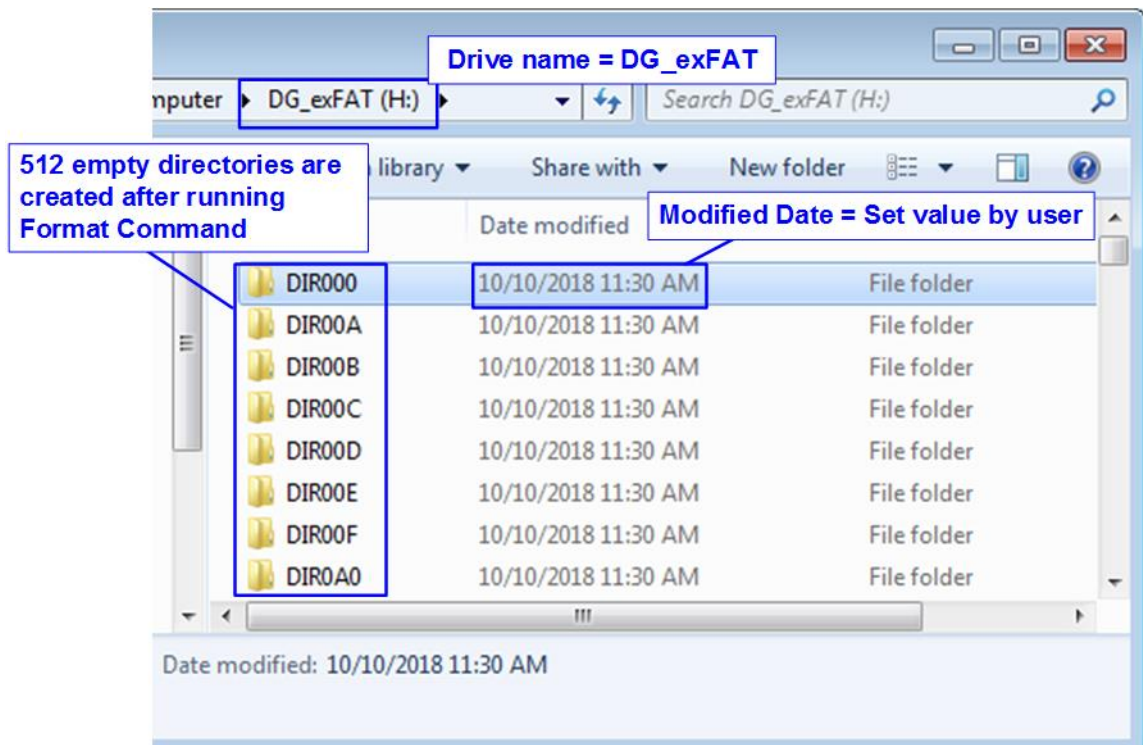


Figure 3-4 512 Empty directories after finishing Format Disk command

## 3.2   Write File

Select '1' to send Write file command to exFAT IP. The step to run Write file command is as follows.



Figure 3-5 Result from Write File menu with setting time created

1) Similar to Format command, input 'y' to change created date and time or input other keys to use the default value. There are six parameters to set created date and time for the new file, i.e.
   a) Year        –Year of created date. Valid range is 1980 – 2107.
   b) Month       – Month of created date. Valid range is 1 – 12.
   c) Date        – Date of created date. Valid range is 1 – 31.
   d) Hour        –Hour of created time. Valid range is 0 – 23.
   e) Minute      – Minute of created time. Valid range is 0 – 59.
   f) Second      – x2 Second of created time. Valid range is 0 – 29.

   The input is received as decimal unit. User adds "0x" to be a prefix for hexadecimal unit. If the input is invalid, the parameter does not change (the same value is used). Only the input in the valid range is applied to update the parameter.

   As shown in Figure 3-5, year parameter is invalid, so the value does not change (2018 is applied). Month, date, hour, minute, and second are valid, so these parameters are updated to be the new default value. After that, "Date and Time changed" and the new created time and date are displayed on the console.

2) Input other keys (not 'y') to use the recommended start file no. The console displays the recommended value which is the next value from the latest write file.
   Note: Input 'y' to change start file no. is allowed for replacing the old file. After running, the data in the old file is replaced by the new one.

3) Input NUM of file – Input total files to transfer in this command. After finishing write file operation, Filename <Start file No>.BIN - Filename <Start file No + NUM of file - 1>.BIN are stored in the device. The input is decimal unit when input only digit number. User can add "0x" to be prefix for hexadecimal unit.

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

After all inputs are valid, total data size (calculated by file size x Num of file) is displayed on the console. Next, Write file command is operated. During writing file, current transfer size is displayed on the console every second to show that system is still alive. Finally, total size, total time usage, and test speed are displayed on the console as test result.

One directory limits the number of files to store, so the new files may store in the different directory. After finishing Write file command, the console displays the 1st file and the last file which has just written with the directory path (calculated by <file name>/<maximum files per directory>). Finally, the console displays the last available file in the disk which may not be equal to the last file in the latest write test when the user runs the write file test for replacing the files.
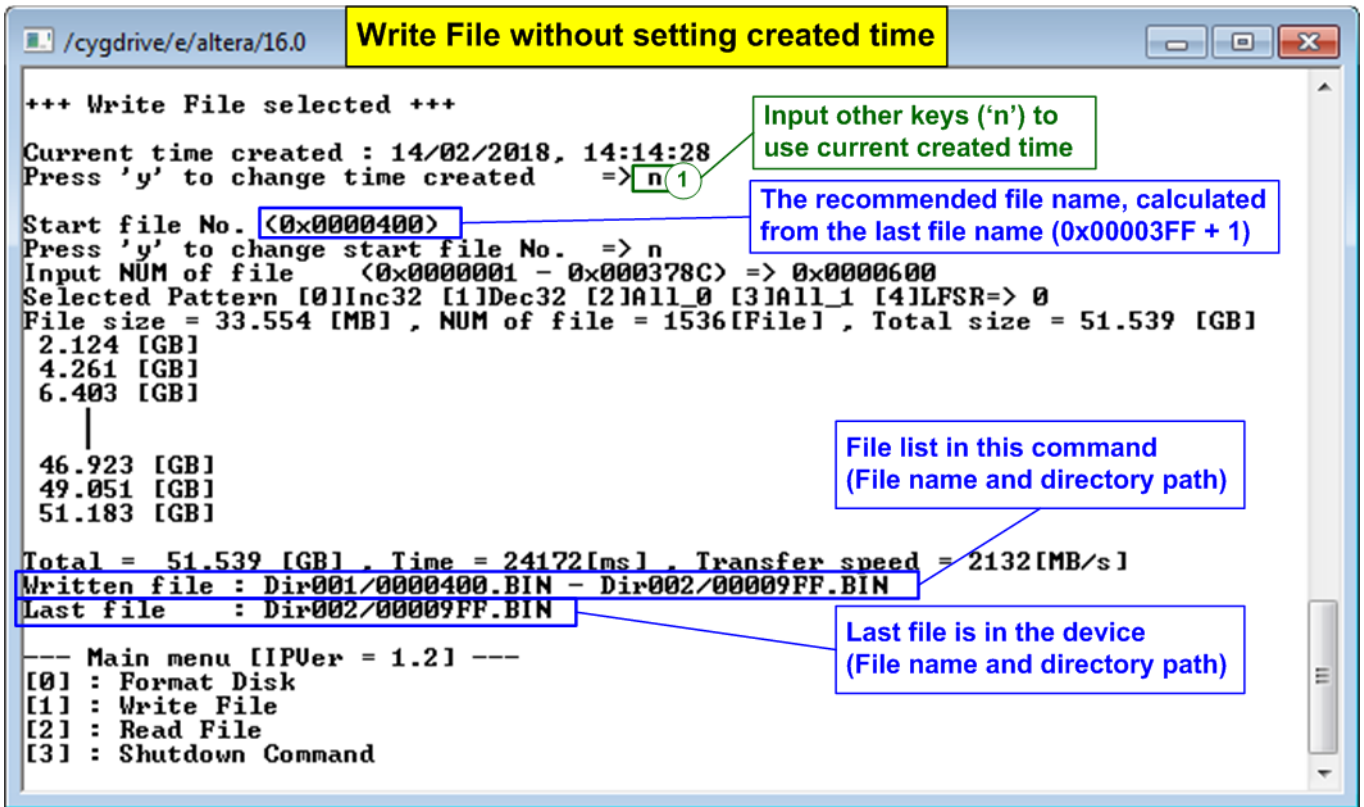
Figure 3-6 Result from Write File by using current created time

Figure 3-6 shows the example to Write File by using current created time. "Start file No" of this test is updated from the previous write test. The previous write test in Figure 3-5 writes file "0000000.BIN" – "00003FF.BIN", so the recommended "Start file No" is 0x0000400 (0x00003FF + 1). After finishing creating 1536 (0x600) file, 0x0000400.BIN – 0x00009FF.BIN are available and the last file in the disk is updated to be 0x00009FF.BIN.
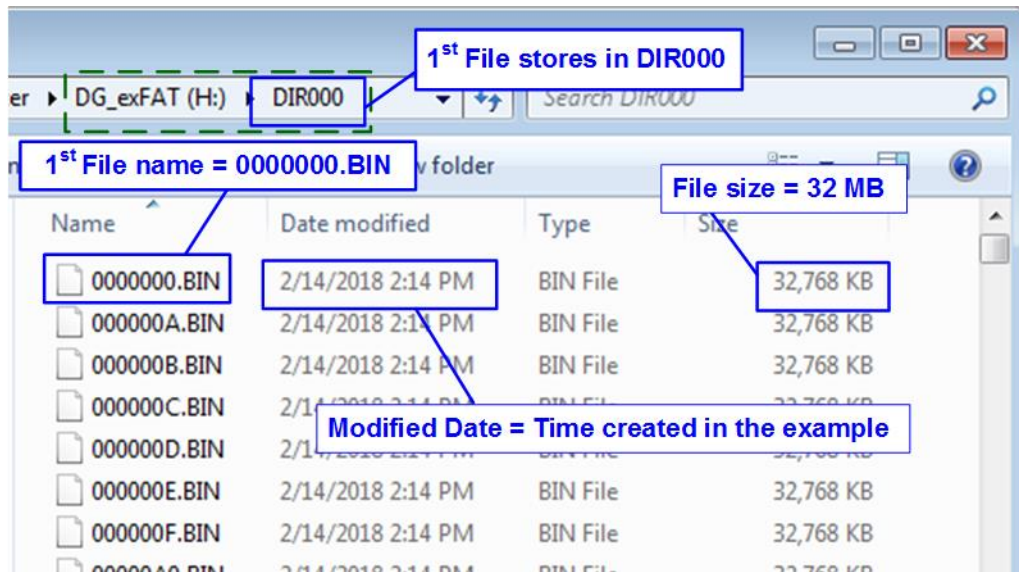
Figure 3-7 Example test files written by Write File command

When plug the device to PC, PC detects the files in subdirectory of DG_exFAT drive. The 1st file in the disk is 0000000.BIN (stored in DIR000). File size and modified date of each file are equal to the created date and created time setting in Write File test.

*Note: When connecting disk to PC, please do not create, write, or modify data in the disk. If the disk has some modification, the disk must be re-formatted by exFAT IP.*

Figure 3-8 Test data in sector#0 - #1 of file "0000000.BIN" by increment/LFSR pattern

Test data is split into 512-byte unit. For incremental, decremental, or LFSR pattern, each 512-byte data has the different 64-bit header which consists of 48-bit address (calculated by <file name x file size> + offset in the file) and 16-bit zero value. 48-bit address is the unique value for each 512-byte data. The unique header is not included when running all-0 or all-1 pattern

The data after 64-bit header is the test pattern, selected by user. The example data in file "0000000.BIN" when writing data by incremental pattern is in the left window of Figure 3-8. The right window of Figure 3-8 shows the example when test pattern is LFSR pattern. The header does not depend on test pattern (incremental, decremental, and LFSR). The test data of test pattern is different after byte#8.
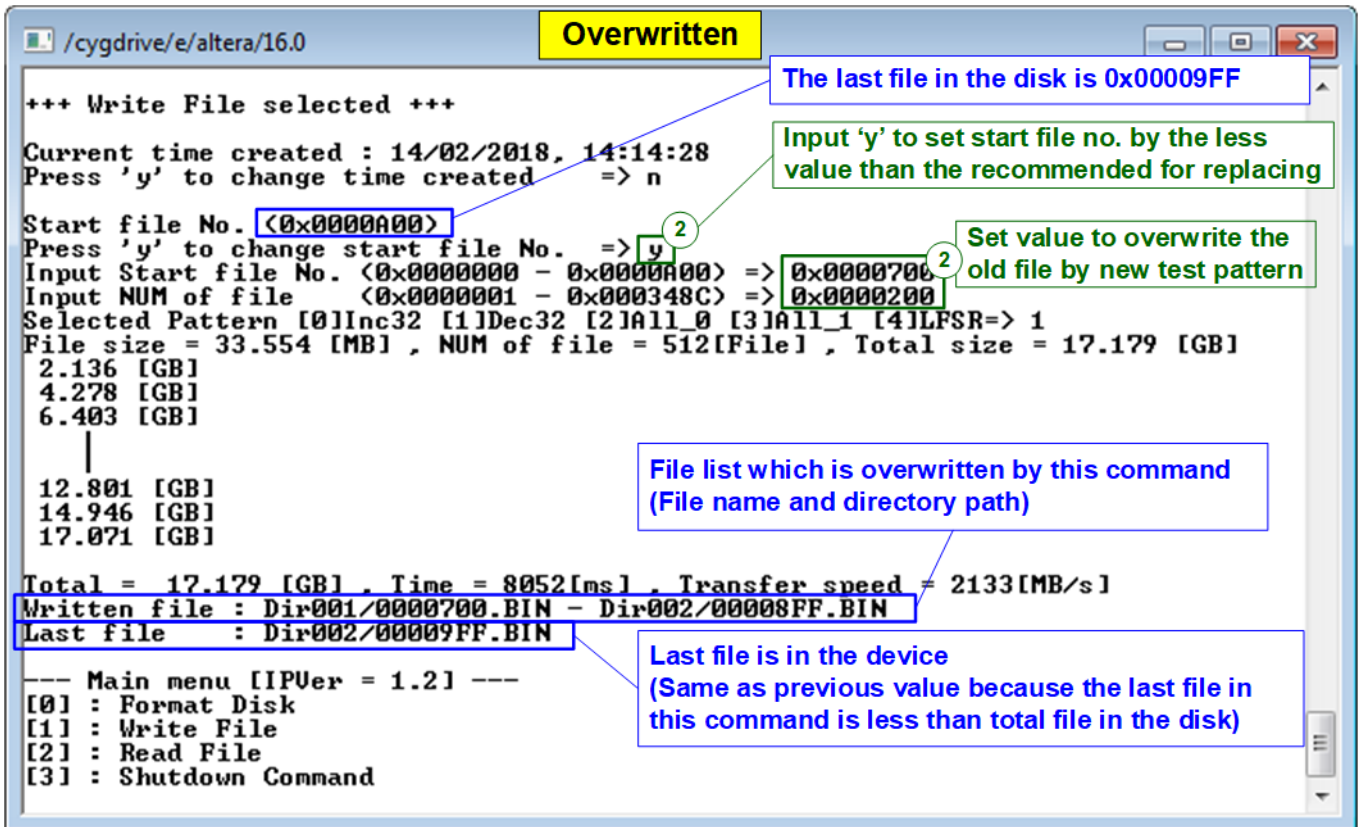
Figure 3-9 Set new start file no.

Figure 3-9 shows the example of the overwritten case by setting Start file No. and NUM of file to be less than the default value. The last file in the disk is 0x00009FF, but the new command writes 0x0000700.BIN – 0x00008FF.BIN by using decremental pattern which is different from Figure 3-6 (0x0000400.BIN – 0x00009FF.BIN are created by using incremental pattern).

After finishing the operation, the data in 0x0000700.BIN – 0x00008FF.BIN is decremental pattern. Because the last file in the new command (0x00008FF.BIN) is less than the last file in the device (0x00009FF.BIN), the last file on the console is not updated. Otherwise, the last file is updated by the last file in this write test.

Figure 3-10 shows the example of error messages when the input is out of the recommended range for each parameter. "Invalid input" message is displayed on the console and then returns to main menu.
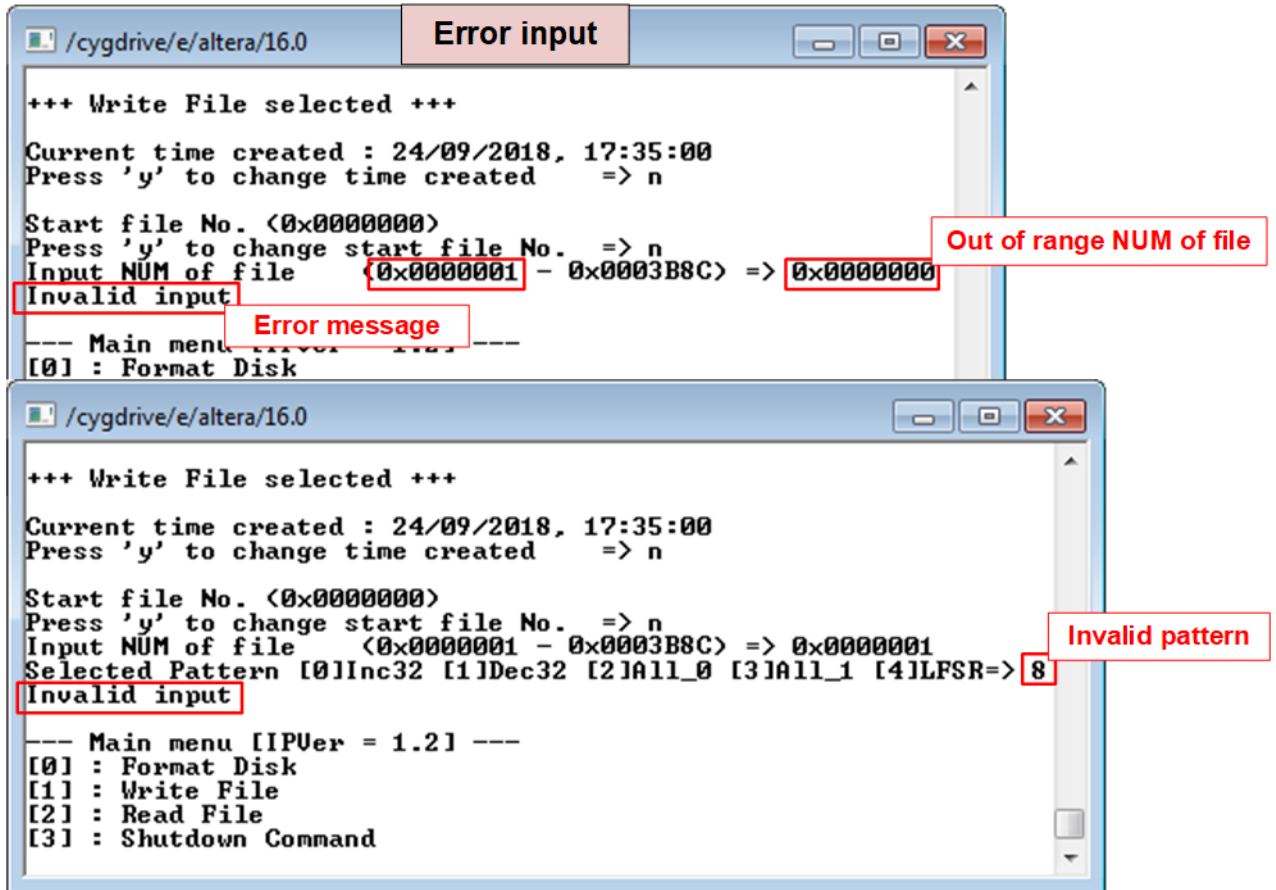


Figure 3-10 Error message from the invalid input

## 3.3   Read File

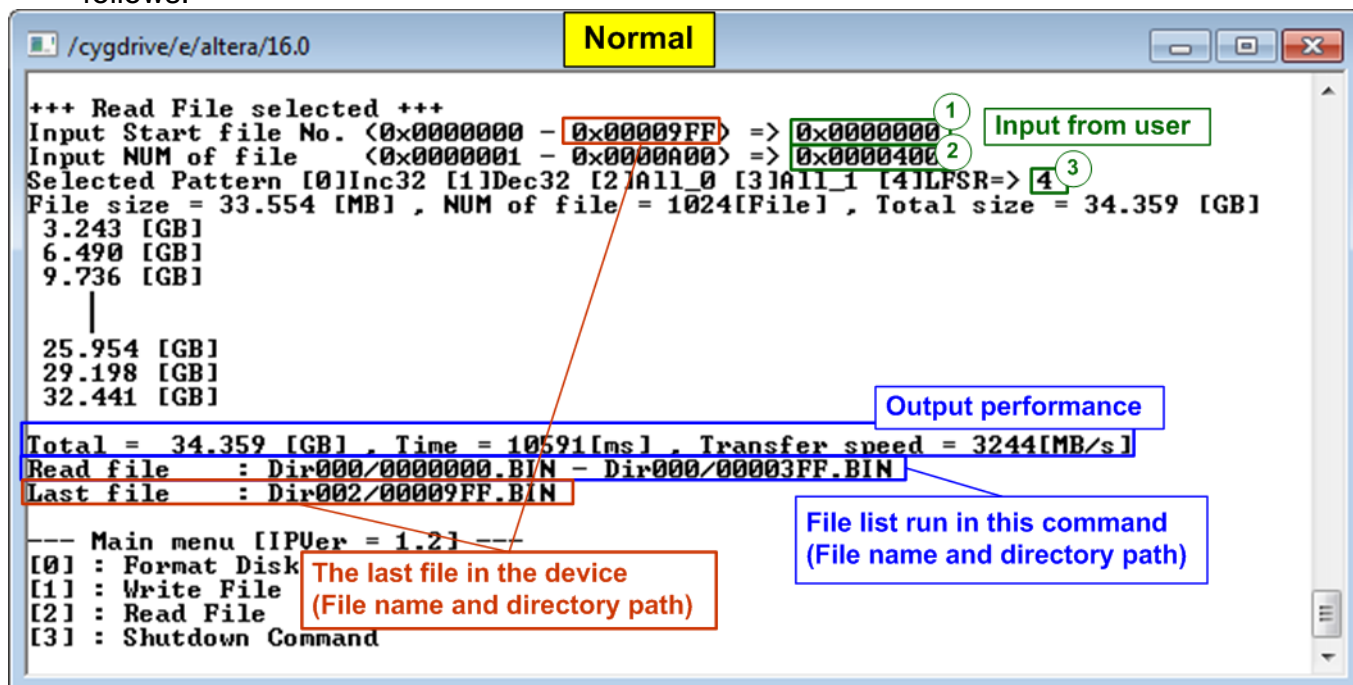Select '2' to send Read file command to exFAT IP. The step to run Read File command is as follows.



Figure 3-11 Read File menu when verification is successful

1) Input Start file No – Input the 1st file name to read. The input is decimal unit when input only digit number. User can add "0x" to be prefix for hexadecimal unit. The maximum value is equal to <total file in the disk – 1>.

2) Input NUM of file – Input total files to transfer in this command. The input is decimal unit when input only digit number. User can add "0x" to be prefix for hexadecimal unit. The maximum value is equal to <total file in the disk – Start file No. input>.

3) Input test pattern – Select 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.

When "Start file No.", "NUM of file", and "Select pattern" are valid, total data size (calculated by file size x NUM of file) are displayed on the console and Read file command is run.

During reading file, current transfer size is displayed on the console every second to show that system is still alive. Finally, total size, total time usage, and test speed are displayed on the console as test result.

When input is out-of-range, "Invalid input" is displayed with operation cancelled.

Similar to write file menu, the console shows the 1st and the last file of the latest command. Finally, the last available file in the disk is displayed although this value is not updated by Read menu and still be equal to the upper range of Start File No. input.

Figure 3-12 shows the example of error message when data verification is failed. "Verify fail" is displayed with the first error file name, the error address of a file, the expected data, and the read data. User can press any key(s) to cancel read operation or wait until the read process is completed.

After the user waits until read command completing, the output performance is displayed to be a test result. The user can continue to run the system test if the error is caused from wrong test pattern input. Otherwise, the command may not complete in the good sequence. It is recommended to power-off/on AB16/AB18 and press "RESET" button to restart system.
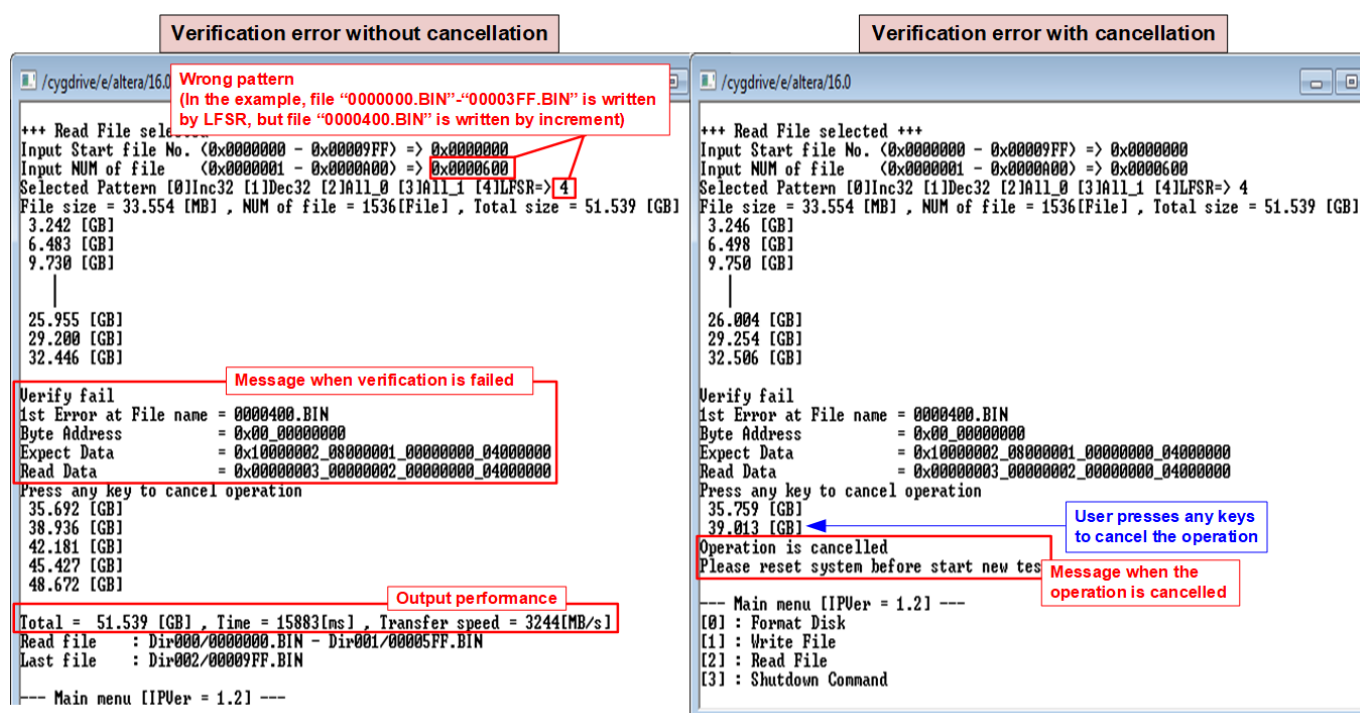


Figure 3-12 Data verification is failed

## 3.4   Shutdown Command

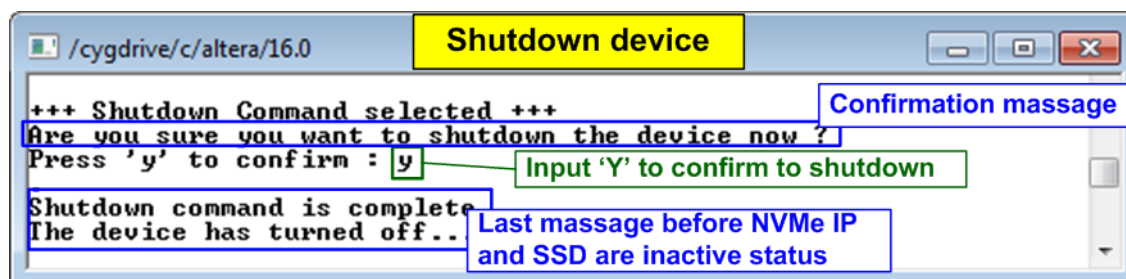Select '3' to send Shutdown command to NVMe device.



Figure 3-13 Shutdown Command with confirmation

The confirmation message is displayed on the console. User inputs 'y' or 'Y' to confirm the operation or other keys for cancellation.

After finishing the operation, the console cannot receive user input. User needs to power off/on test system to start the new test operation.

## 4   Revision History

| Revision | Date | Description |
|----------|------|-------------|
| 1.0 | 23-Jan-19 | Initial version release |
| 1.1 | 22-Mar-19 | Add file size and total file information |
| 1.2 | 16-May-19 | Correct directory name and update overwritten feature |
| 1.3 | 20-Apr-20 | Remove power adapter cable from AB18 |