Design and Implementation of Multi-partition Paralleled Image Storage Hardware File System for MARS Rover

Yong Xu[1(✉)], Cuilian Wang[1], Lei Zhao[1], Pangfeng Wu[2], and Wenjuan Li[1]

[1] Beijing Institute of Spacecraft System Engineering, Beijing 10094, China
andrexu@163.com
[2] Shandong Aerospace Electro-Technology Institute, Yantai 264670, China

Abstract. Mars rover mission carried many types of cameras, which need to complete complex science exploration and research task, it's hard to meet the demands of image paralleled storage and access operations using the traditional Chang-E's multiplex and storage scheme. Due to the speed and memory size limitation of on-board highly reliable radiation-hardened computer, the ground computer file system cannot be realized. This paper designed a FPGA based hardware paralleled image file system to meet the requirement of image data storage management in the Mars rover explore task. In addition the multi-type and multi-camera data file storage system of MARS rover is implemented in the actual rover's computer which provided multi-partition paralleled reading and writing of multi-image files and on demand addressing and copying functions while ensuring high reliability.

Keywords: MARS rover · Paralleled image storage · Hardware file system

1 Preface

Mars is the closest planet to Earth in the outer orbit of the solar system, and the cost of human exploration of Mars is relatively low. And because of Mars special space position, many of its features are similar to Earth, which makes it most possible for humans to develop a second home in the universe. At the same time, studying Mars can help us understand some of the mysteries of the birth and evolution of the solar system. Among all kinds of information obtained by the Deep Space Exploration Research Institute, image information is the most intuitive and core information. Mars Exploration Rovers (MERs), the Spirit and Opportunity rovers that landed on Mars on January 4 and 25, 2004, respectively, have traveled tens of kilometers on Mars and sent back more than 100,000 photographs, including precious exotic photographs of meteors, sunsets, eclipses and cyclones. Among them, the May 19, 2005 "courage" captured the scene of the sunset of Mars. In January 2007, NASA was named the best picture of Mars in the public election. What is more significant is that they not only found evidence of volcanic eruptions on Mars, but also found silica (May 2007), adding new evidence to the Martian theory of life, and the strongest evidence of water

M. Jia et al. (Eds.): WiSATS 2019, LNICST 280, pp. 371–379, 2019.
https://doi.org/10.1007/978-3-030-19153-5_39

on Mars so far [1]. As a result, deep space probes usually carry multiple sets of cameras. Take the identical designs of Courage and Opportunity for example, each has nine visible-light cameras, while Curiosity has up to 17 cameras, including mast cameras, obstacle avoidance cameras, landing cameras, chemical analysis cameras, and mechanical arm cameras. Machine and so on. Multi-type camera, multi-imaging system, multi-resolution, multi-camera and other factors pose new challenges to the storage and management of Mars Rover image data. Generally speaking, because the onboard computer needs to withstand the examination of long-term harsh space irradiation environment, the performance of the electronic components of the selected space-level computer is rather limited, and it is impossible to deploy the more mature file system similar to NTFS and YAFFS [2] on the ground. Therefore, the traditional remote sensing satellite and the Chang'e Lunar Exploration Series before our country can not be deployed. The on-orbit image data storage of the measuring task is based on the multiplexing storage scheme of FPGA [3]. After multiplexing, the task data is sequentially stored in NandFLASH chip according to VCDU format, and read and write sequentially by recording and playback. Data storage scheme based on multiplexing storage [4] is more suitable for sequential recording and playback of data similar to tape recorders, and can not meet the data management requirements of Mars Rover multi-camera, multi-partition, multi-resolution, multi-system image flexible storage and access and image processing on the Mars surface. Under the premise that it is impossible to improve the performance and memory of the irradiation-resistant computer on board, a hardware file system based on FPGA architecture is designed to meet the requirements of large-capacity storage management, high-speed parallel storage of multi-type camera data, flexible data access and on-orbit data processing.

According to the data characteristics of Mars exploration mission and camera image data, this paper designs a hardware-only image partitioning parallel storage and management file system, which includes a set of bad block table management, partitioning management, image file data storage management and access mechanism of Nand-FLASH high-capacity memory, and designs a scheme based on the system. The hardware file system of Mars Rover image storage is realized. In addition to Nand-FLASH data memory, the MRAM is used as bad block information management and file structure management memory. The file system management mechanism is implemented based on FPGA, and a set of hardware-only partitions is implemented. Parallel, fast and highly reliable Rover image management system.

2 Design of Image File Storage System

In the Mars rover, the image file storage and processing system is located in the onboard computer, which mainly completes the storage and management of image data sent by each camera, the engineering load multiplexing storage, image compression, compressed bit stream organization transmission, telemetry/playback data download, and data communication with the central processing control module. The image file storage and processing system adopts A/B dual-machine cold backup mode. The single component block diagram is shown as follows, mainly including the following functional components (Fig. 1):

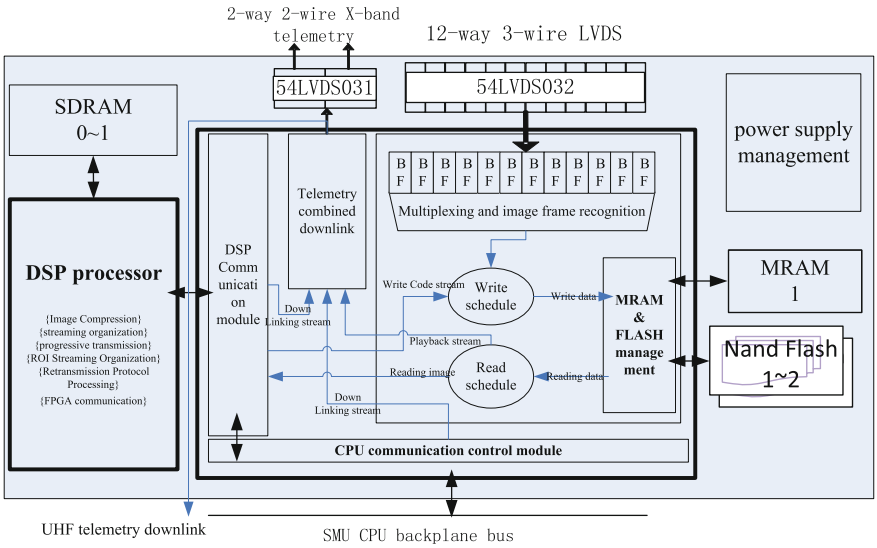

Fig. 1. Block diagram of data storage & processing module

(1) LVDS interface module: completes the function of high-speed differential interface between receiving load data and downloading data to data transmitter.
(2) FPGA chip module: 3 million gate anti-irradiation FPGA is used to realize CPU communication, image data reception and NandFLASH file system, MRAM and NandFLASH interface and redundant fault-tolerant management.
(3) MRAM & FLASH Memory Array Module: MRAM stores all kinds of non-volatile information, including bad block table, read-write address, etc. NandFLASH is used to store large capacity data.
4) DSP & SDRAM processing component: used for image compression algorithm, bit stream organization and downlink control. DSP uses an anti-irradiation space-level DSP processor, and two SDRAM buffers are added to provide the computational buffers needed by the image compression algorithm.
(5) Power management: module power supply, power on reset, voltage conversion and so on, generating 3.3 V, 1.5 V and 1.8 V voltage.

3 File System Data Management Mechanism Design

In order to meet the requirement of miniaturization of the Mars rover, the memory array uses two 3-D stacked NDFLASH chips [5] (eight K9F8G08U0M substrates) with 64 Gbits per chip, so the total on-board physical capacity is 64 Gb * 2 = 128 Gb = 16 GB (Figs. 2 and 3).

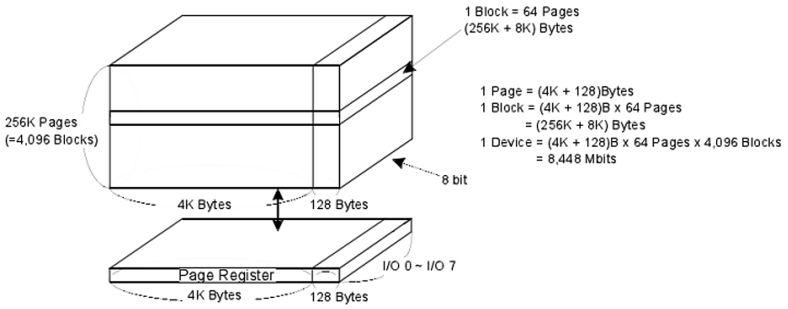

Fig. 2. NandFLASH K9F8G08U0M array organization

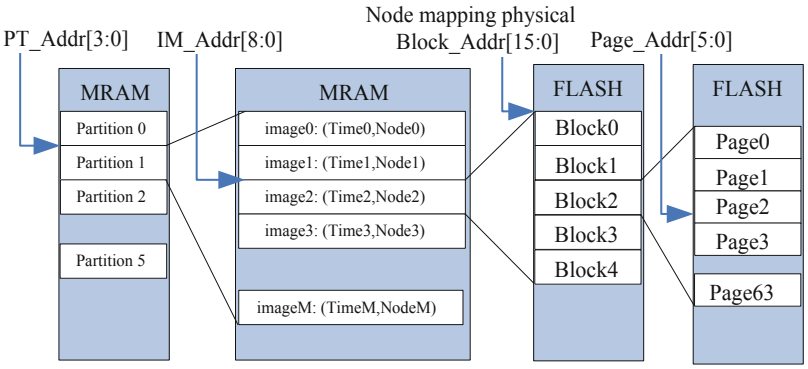

Fig. 3. Organization of panchromatic image data in MRAM

As shown above, each K9F8G08U0M substrate contains 4096 Block, and each Block contains 64 pages, 4096 Byte per page. Therefore, each board has a total storage space of 4194304 pages, 65536 blocks, and block address range 0x0000–0xFFFF. In order to support parallel storage in random access by camera and by image frame, the image storage system is divided into 11 partitions and the storage space is allocated according to the block as shown in the following table (Table 1):

Data organization in MRAM is shown in the figure above. The initial allocation of 32bit * 2K logical space in MRAM stores the bad block information of the whole Nandflash, and then allocates 6K * 32bit space to store the node information of image files in each partition. The node information of each file includes 32bit time and some 32bit node physical address information. To store the above information, a magneto-electric memory (MRAM) with a capacity of 8 Mb (32 * 256K) is mounted on the board. All the data stored in MRAM is stored in three addresses and accessed according to TMR. The logical space is 32bit * 85K. Besides the 32bit * 8K space analyzed above, there is a surplus of 32bit * 77K left for storage others in formation.

Table 1. Image storage partition address allocation table

Partition	Type data	Block address range	Capacity
1	Camera1	0x0000–0x0FFF	512 MB
2	Camera2	0x1000–0x1FFF	512 MB
3	Camera3	0x2000–0x2FFF	512 MB
4	Camera4	0x3000–0x3FFF	512 MB
5	Camera5	0x4000–0x4FFF	512 MB
6	Camera6	0x5000–0x5FFF	512 MB
7	Engineering telemetry	0x6000–0xBFFF	$\geq$5 GB
8	Internal use cache 1	0xC000–0xCFFF	512 MB
9	Internal use cache 2	0xD000–0xDFFF	512 MB
10	Alternate partition 1	0xE000–0xEFFF	512 MB
11	Alternate partition 2	0xF000–0xFFFF	512 MB

4 FPGA Design Implementation of File System

It mainly includes five modules: image receiving and writing module, DSP image access module, image node information initialization module, MRAM controller, FLASH access control module (Fig. 4).

(1) Image Receiving and Writing Module: Create image file: Use the way that the nodes in the partition are sequentially used backward. When the image is created, the MRAM access module reads the current image count n of the corresponding partition; then the newly created file uses n + 1 image and writes the current timestamp; read the corresponding n + 1 node address in MRAM. Return to the image data receiving module; update the partition node count to N + 1, write back the current image count of the corresponding partition in MRAM; write to the image file: image data is received after image creation; every page of data received, call FLASH control module to write to FLASH. Close the image file: After receiving the image file close command, the remaining less than 1 page of data, fill a full page, write to FLASH (Fig. 5).

(2) DSP image access module complete the random access of image files, when we need to read and write the image; take the first logical address of the P partition in MRAM, access the inner block address (physical address), with the physical address to NandFLASH to continuously access five blocks of data content. The first page of the first node of each image is the Affiliated Information page, which stores the relevant time, camera parameters and other parameters. After the first page of the first node is accessed back to a total of 298 data pages, an image data is obtained.

(3) Image node information initialization module establishes the image node information by erasing the partition from the partition start address Bi to the end block address Ei, and updating the failed block information to the bad block table; and searching from the partition start block address to the end block. The first block address of the consecutive block is the starting address of the node; the storage

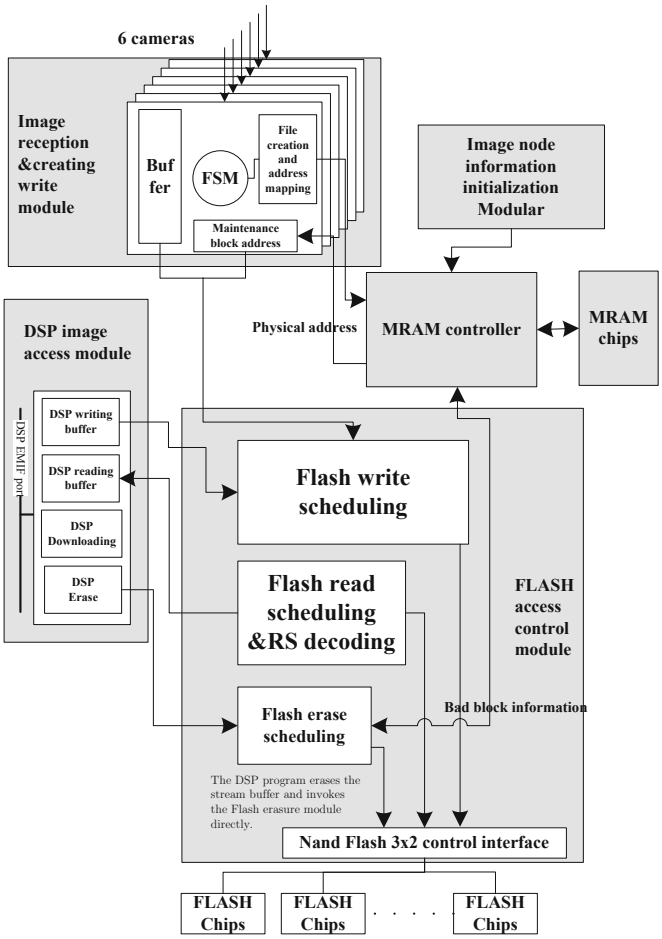

Fig. 4. Top-level architecture design of the FPGA files system

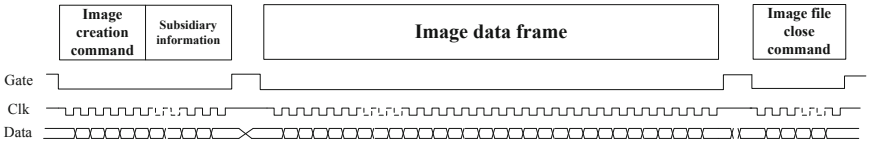

Fig. 5. Image files creation and write operation initialed by camera

mapping is related to the first node address of the partition in MRAM; if enough 512 nodes are established or queried to the end address of the partition. Clear the number of images stored, indicating that the partition is empty.

(4) MRAM controller which manages one piece of MRAM, provides five independent MRAM access ports, and five ports can access MRAM at any time. Three

physical addresses P_A0～2 are converted from one logical address L_A by cyclic priority processing the access requests of each port.

(5) FLASH Access Control Module realize the universal operation control of Nand-FLASH, including the detection of bad block information of FLASH, and control the operation with block as the unit. FLASH erase, erase in block units; FLASH programming, write one page of data per programming unit; FLASH read, read data in page units.

5 Design Result Analysis

In order to adapt to the long-term work of deep space exploration interstellar navigation and the system reliability under harsh space irradiation environment, the file system FPGA is designed and implemented with a space-level anti-irradiation 3 million gate FPGA. The design is based on VHDL language, and its design hierarchy tree is shown in the image below. There are 19 top-level modules (Fig. 6).

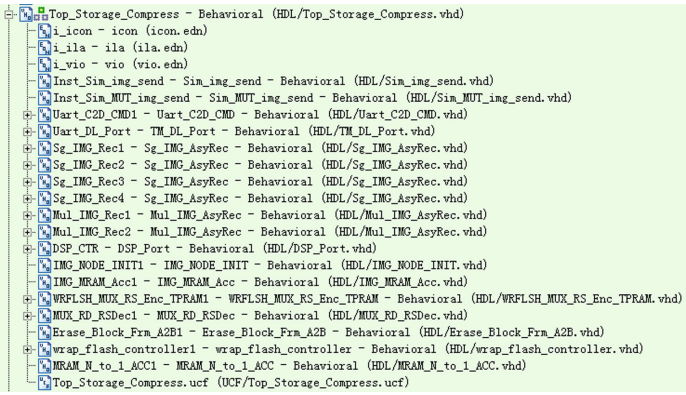

Fig. 6. Hierarchy tree of FPGA design

Each module is implemented by a secure state machine, and three modular redundancy (TMR) design is adopted on key registers and critical control paths, supplemented by [6] on orbit scrubing, which can effectively resist the effects of single particle interruption in space environment and ensure the reliability of the rover in the mission process. The design is finally synthesised by ISE9.2 XST tool. The design resources are shown in the following table (Table 2):

ModelsimSE 6.4 is used to simulate the design. Peripheral camera interface, NandFLASH and other devices test bed and test records are compiled. Image files are created, received and accessed are simulated. Some simulation results are as follows (Fig. 7):

This research designs a 1:1 verification board for the on-board file storage system of the Mars rover. As shown in the following figure, the on-board verification of proposed

Table 2. Resources utility in XQR2V3000 (−4) FPGA

Logical resources	Used	Available	Utilization
Slice	3832	14336	26%
4-LUT	6821	28672	23%
FF register	4216	28672	14%
BRAMs	53	96	55%
Highest frequency		66.878 MHz	

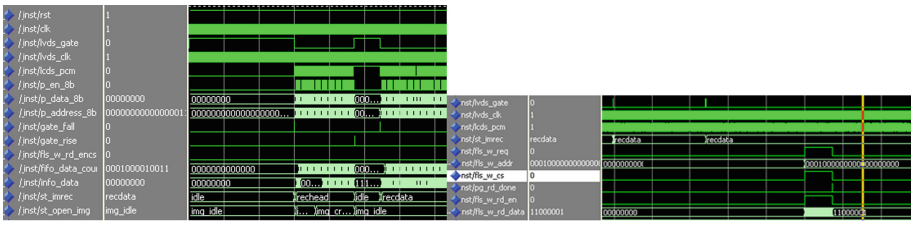

Fig. 7. Simulation waveform of image file creation, receiving and writing operation

file system designed in this paper is completed. The parallel file creation, writing and closing operations of the multi-camera simultaneous operation can be completed, and the flexible storage of the Mars Rover image data can be realized. The file system test master software, as shown in following figure (Fig. 8).

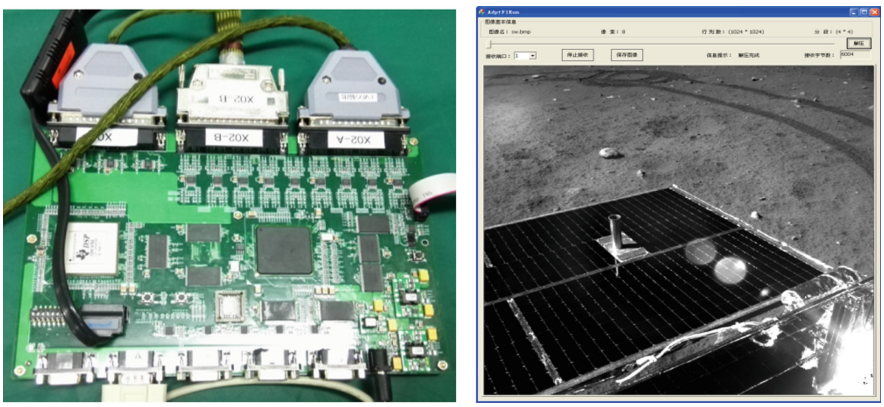

Fig. 8. Evaluation board (left) and Monitor software (right) for image file storage and compression system

6 Conclusion

In Mars exploration mission, Mars Rover carries many kinds of exploration cameras, which need to complete complex scientific exploration and exploration tasks on the surface of Mars. The multiplex storage scheme of traditional satellite can not meet the flexible and parallel storage and access requirements of image data. Due to the highly reliable anti-irradiation satellite-borne computer, the main frequency and memory of the controller are very small, so that the existing computer file system on the ground can not be realized. In this paper, a parallel image file system based on FPGA is designed to meet the requirements of image file storage and management in Mars exploration mission. The parallel image file system is implemented by FPGA, which takes into account the characteristics of high speed, high reliability and high flexibility, and resists by adopting various measures such as RS encoding and so on. Multi-type, multi-camera data file storage and management, while ensuring high reliability, provides multi-partition, multi-image file parallel read-write and on-demand addressing functions, effectively solving the traditional multiplexing storage equipment in the image data parallel storage and Mars surface image processing on-demand access problems.

References

1. Xu, X.: MERs Deep-space Imagery Compression on—board. J. Test Meas. Technol. **21** (2007)
2. Charles Manning: Introducing YAFFS, the first NAND-specific flash file system. http://www.linuxdevice.com/articles/AT9680239525.html
3. Pei, N., Li, K., Zhao, L.: Rate adaptive image extraction method based on FPGA. Modern Electron. Tech. **36**(19) (2013)
4. Wu, Z.: Research of on-board date routing multiplexing and srorage. Harbin Institute of Technology (2016)
5. 3D-Plus Co. Memory Module Flash Nand 4Gx16-SOP 3DFN64G16VS8477, November 2012
6. Yin, M.: SEU-tolerant design of SRAM FPGA for space use. Spacecraft Environ. Eng. **28**(6) (2011)