

HIGH RELIABILITY FLASH FILE SYSTEM USED INTO DATA UNIT OF FLIGHT DATA RECORDER

Przemysław Kordowski, Marcin Chodnicki, Mirosław Nowakowski

*Air Force Institute of Technology
Airplanes and Helicopters Division
Ks. Bolesława Street 6, 01-494 Warsaw, Poland
tel.: +48 261638489, fax: +48 261364603
e-mail: przemyslaw.kordowski@itwl.pl
marcin.chodnicki@itwl.pl, mirosław.nowakowski@itwl.pl*

Wiesław Cuper

*43. „Oksywska” Baza Lotnictwa Morskiego
Zielona Street, 81-197 Gdynia, Poland
tel.: +48 261268004, fax: +48 261364666
e-mail: seaspider@wp.pl*

Abstract

Since the appearance of the solid-state flight data recorders, semiconductor memories are still modernized and equipped with new features and higher capacities. This process allows developing new functionality for FDR's but entails effort to design reliable memory management algorithms. In order to provide usage the entire memory area of modern semiconductor memories, is required to assure flash file system utilities inter alia wear levelling, error corrections and bad blocks management. There are some open source file systems for flash memories but any can face strict requirements of flight data recording data unit. Due to this reasons Air Force Institute of Technology has been developed flash file system adapted to meet the requirements. The most significant issue was to ensure of stable work after sudden power cut in any moment. To cover the highest level of assurance, additional memory is held in idle mode. Recording system is implemented into ARM Cortex-M3 microcontroller, which handles two nand flash memory chips. Whole code is written in C language so it can be effortless migrated into processor architectures. Article explains source of strict necessities and difficulties of creating high reliability flash file system and presents main features of it. In addition, there are explanations of selected algorithms in order to clarify developed software.

Keywords: *transport, vehicles, flight data recorder, file system, semiconductor memory*

1. Introduction

The most significant advance in evolution of flight recorder technology was held at late 1980s, which was appearance of the solid-state flight data recorders. Since occurrence at aircrafts, semiconductor memories are experiencing dynamic growth. This development allowed for registration more and more parameters from aircraft, integrating cockpit voice recorder and flight data recorder into one single block and finally in nearest future equipping this block with feature to record video from pilots cabin. This entails increase amount of required memory space and enhance recording speed capabilities. There is variety of semiconductor memories in presently market, which can undertake these demands. The nand flash memories are used everywhere there is necessity to provide fast and high capacity of non-volatile memory space. Despite the application of the highest quality memory chips there are some disadvantages and difficulties, which have to be carried out. The first step is to provide all memory handling functions including bad blocks management, error checking and correction algorithms, wear levelling etc. Another important issue is to organize specific records into memory space. The ideal solution would be to

apply commonly used files system. There is some open source flash file systems specialized for use on flash memories but not anyone can face strict requirements. The most significant issue was to ensure of stable work after sudden power cut in any moment. The solution undertaken in Air Force Institute of Technology was to implement own flash file system named recording system. The paper describes main features and some algorithms of designed recording system

2. Flight data recorder software layers

Flight data recorder or quick access recorder is a device which only task is to record reduced data to the appropriate arrangement and format comes from flight data acquisition unit (FDAU). Because of strict and tough requirements, software was divided into three layers. Layer closest to a memory chip is responsible to communication with nand flash, bad block management, usage of on die error correction coding and wear levelling. Recording system takes over some wear-levelling task but mainly it is responsible for providing suitable arrangement of records and manages writing and reading procedures. Main program controls receiving procedures of data and takes control over entirety. There is one not mentioned but important software part supervising proper functioning of whole device, called diagnostic system. Structure of flight data recorder is presented on Fig. 1.

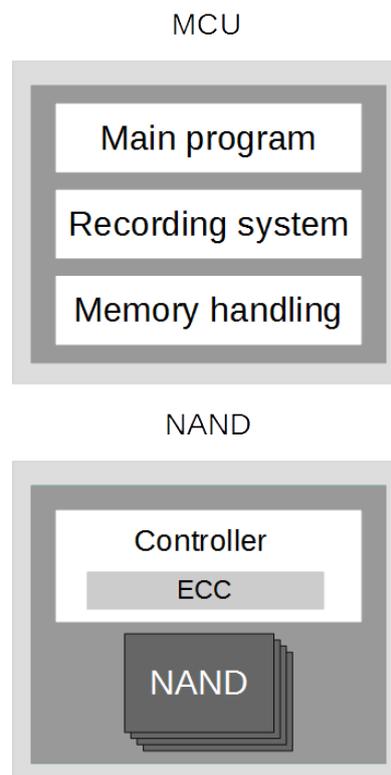


Fig. 1. FDR software structure

2. Requirements of flash file system compared to designed recording system

From the definition file system controls how data is stored and retrieved on other hand, flash file system is designed to meet these requirements on flash memory-based storage devices. Both are strongly related but flash file system takes into account all issues caused by usage flash memories, e.g. write amplification avoidance.

Memory of modern FDR can be presented as digitalized magnetic tape recorder where newest data taking place of the oldest record. Such organization allows applying comparatively

lightweight structure of stored data. In order to ensure low latency data access, recording table was implemented which holds 127 last records pointers.

Key features of every flash file system are:

- a. Special blocks erasing – flash memory blocks have to be explicitly erased before they can be written to. The time taken to erase blocks can be significant, thus it is beneficial to erase unused blocks while the device is idle,
- b. Wear levelling – flash memory devices tend to wear out when a single block is repeatedly overwritten; flash file systems are designed to spread out writes evenly,
- c. Write amplification protection – prevention from multiple rewriting of the amount of physical information mostly caused by wear levelling and garbage collection.

Regarding to sequential character of stored information there is no need to make any erasing operations until new data has been arrived. Wear levelling algorithms are easily achievable by the same reason. Because of unique and simplified arrangement of stored records, there is no need to apply garbage collection algorithms. It follows that there is no need to take any write amplification protection algorithm. Every disadvantage of usage nand flash memory chip is prevented by recording system conducted with memory handling software layer. Procedures of writing data and records arrangement will be described in next chapter.

3. Recording system implementation

Memory handling layer has taking effort of bad block management and it is closely associated with recording system layer, which allows simplification of both. Standard bad block handling memory organization is presented on Fig. 2.

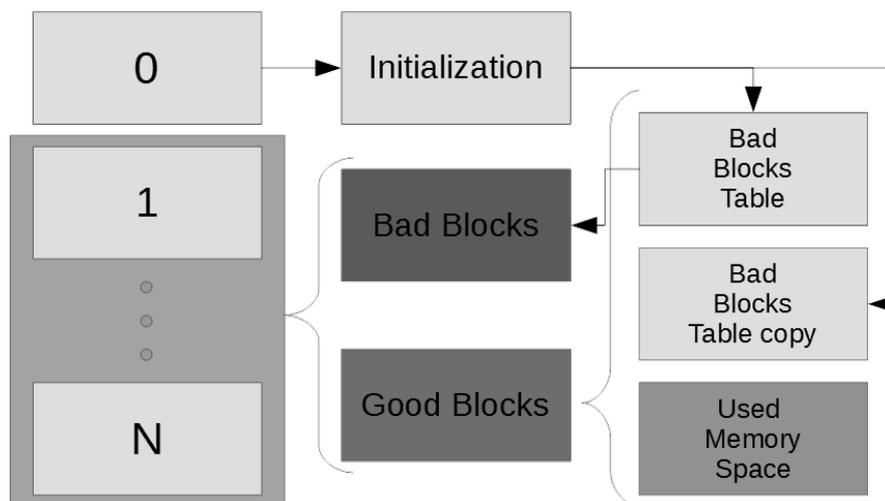


Fig. 2. Recording system memory space organization

Important feature of used memory chips is that even factory fresh unit may contain bad memory spaces. Brand new silicon can take about two percent of bad block and it is recognized as valid in case if every bad block is properly marked. Block number 0 is always valid and there are stored pointers to bad block table and its copy. Bad block table contains information which blocks are excluded from memory used space. At initialization, pointers from block 0 are read and then information from bad block table is copied to RAM memory. After that recording system can be launched

Recording system software layer taking place between main program and memory handling layer. His main task is to arrange data as efficiently as possible simultaneously providing all features of nand flash file system. Recording system takes information about bad blocks locations

and location of table of records. Bad Blocks are excluded and pointer to last record is loaded in order to find where the new one is to take place. In this time procedure to close last record has been started. Any data left into temporary area becomes attached to last record and then it is closed. After this place new record is started. Structure of memory space organization is presented on Fig. 3.

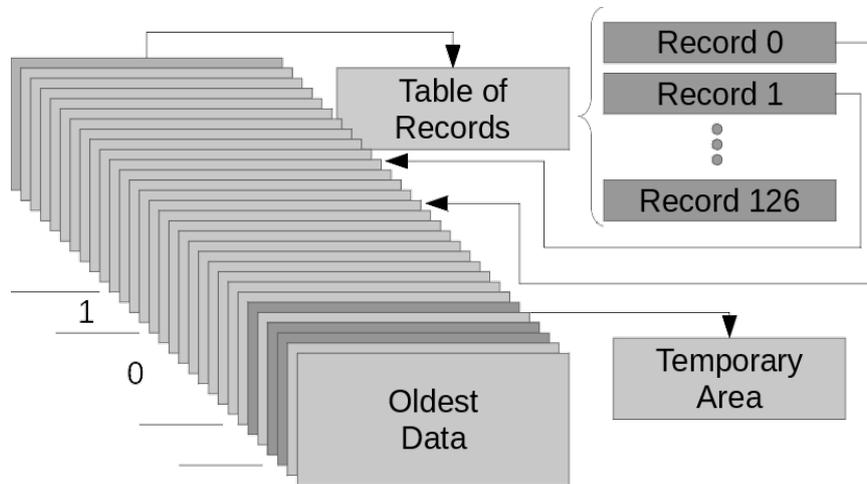


Fig. 3. Recording system memory space organization

Almost all memory space is used for data arrived from Flight Data Acquisition Unit. Table of records, temporary area and four cleared blocks at the end of actual record are used for other purposes. They are needed to provide ability of continuous recording. If block, which is actually written, reached its capacity there is one cleared to take additional data. There is no need to loose time to clearing at this moment. In case of table of records, loss there is a procedure to restore based on markers placed into additional memory space. Interesting is fact that restoring procedure is never lunched in recording mode because last record pointer can be figured out faster by cleared blocks placed at the end of last record. Another and most important issue after procedure of closing last record is procedure of writing data to memory. Nand flash memory is storage device, which does not provide full random access. The smallest portion of data to write is one page and the smallest portion to clear is whole block. This feature brings both profit and loss. In used memory chips there is for example 2048 blocks divided to 64 pages each. This is the point of usage nand flash file systems to ensure that every important aspect will be covered. Any of open source nand flash file systems do not provide full assurance of stable work after sudden power loss. In this case, such as simplification of algorithms allows to movement of closing procedures to initialization process.

Writing method is presented on Fig. 4. The main issue handled by this procedure is to maximize area used for data storing purposes. Every single byte incoming from FDAU is stored into read buffer. When writing interruption is triggered data from read buffer is added to existing data into writing buffer. If the minimum amount of data to write is not achieved that information is stored only into temporary area. It is obvious that temporary area not uses memory space efficiently but it allows storing newest arrived data into non-volatile memory. If power cut would occurred at this time, data remained into temporary area would be available. Similar situation occurs when write buffer holds more data (Fig. 2b). The only difference is writing data into both recording area and temporary area. Such method brings another advantage: complete usage of recording area. Temporary area block is cleared around 30 times until recording area block have been fully written. To provide wear levelling temporary area block changes its location together with recording area block which is clearly presented on Fig. 1. It should be noted that usage of two circular FIFO buffers makes last state of write buffer identical to temporary area writing.

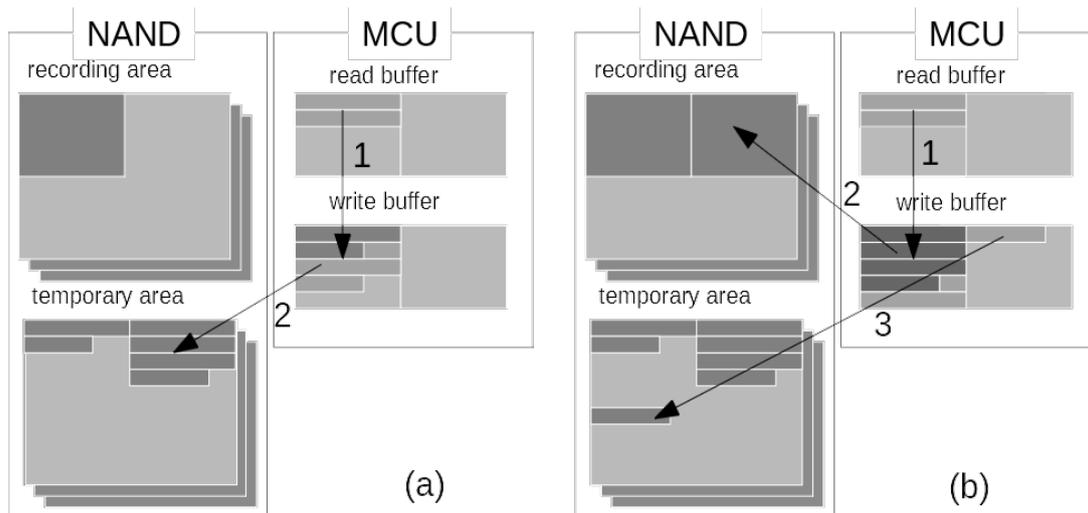


Fig. 4. Recording system writing method

- a. Writing only into temporary area:
 1. Copying data from read buffer to write buffer,
 2. Writing data from write buffer to temporary area.
- b. Writing to recording area and temporary area:
 1. Copying data from read buffer to write buffer,
 2. Writing minimum amount of data to write from write buffer to temporary area,
 3. Writing remaining data to temporary area.

4. Conclusion

Recording System is complete software layer allowing to appliance modern on-die ecc nand flash chip into Flight Data Recorder. An algorithm was precisely adapted to FDR needs, which follows clarity and reliability of designed software. Choosing of such as solution solved many difficulties of trustworthy handling memory chips. Software was applied on Cortex-M3 microcontroller clocked by 84 MHz oscillator. Maximal speed of writing exceeds expectations and reached 6 MB/s. Produced devices are characterized by memory space from 256 MB to 4 GB what allows to store any parametric data and digital voice record.

References

- [1] EUROCAE, Norma ED-112, Minimum Operational Performance Specification For Crash Protected Airborne Recorder Systems, 2004.
- [2] MICRON, TLC MLC and SLC Devices, 2015.
- [3] MICRON, NOR | NAND Flash Guide: Selecting a Flash Memory Solution for Embedded Applications, 2015.
- [4] MICRON, NAND Flash Memory MT29F2G08ABAEAH4, MT29F2G08ABAEAWP, MT29F2G08ABBEAH4, MT29F2G08ABBEAHC, MT29F2G16ABAEAWP, MT29F2G16ABBEAH4, MT29F2G16ABBEAHC, 2014.
- [5] Technical Note Error Correction Code (ECC) in Micron® Single-Level Cell (SLC) NAND, 2011.
- [6] BEA, Flight Data Recorder Read-Out Technical and Regulatory Aspects, France 2005.

