Research on Reliability Design of Data Storage for Embedded System

Liu Chao, Hu Fang
School of Information Science and Technology, Jiu Jiang University, Jiu Jiang, China

Abstract

With the increasing development and extensive application of embedded technology, the reliability of data storage for embedded systems increasingly become issues of concern for developers and users. Many methods to ensure the reliability of data storage in all aspects of the system design process based on the characteristics of embedded systems will be proposed in this paper and have certain significance on the reliability design of data storage in embedded system design.

1. Introduction

With the rapid development of embedded technology, embedded systems have been widely infiltrated into science, engineering, military technology and other areas of daily life, no matter what field of embedded systems is the reliability of data storage a widely concerned issue, which affects product performance, stability and security. Therefore, the reliability of embedded systems technology of data storage has become a new hotspot, and may affect the future direction of development of embedded systems.

2. Data storage reliability

What is the reliability of data storage? Reliability refers to the data storage system is to ensure reliable data storage to ensure data integrity and effectiveness, to ensure data will not abort system has been damaged or lost. The reliability of data storage by many factors, power failure, memory failure, system instability, human error, etc. are a direct result of data storage reliability problems [1]. Therefore, the
reliability of data storage is a matter of hardware, storage technology, operating systems and other areas of the comprehensive study.

3. Storage Reliability of Embedded Systems

The traditional method of ensuring the reliability of data storage is to create the memory image, in different physical hardware for data backup, update data in different image on the need to also update the data, there are two deficiencies in this way, you first need to double the storage capacity, followed by the writing of data loss need more time, for embedded systems, a simple method of using memory image is clearly not meet the majority of real-time embedded systems, streamlining the demand.

The embedded system design, in order to improve system reliability, ensure data storage security, in addition to select good quality hardware platforms and storage devices, there is also a key consideration is the system of anti-power-down capability. We know that when the system was a sudden power-down situations, timely and correct the operating system can not quit, often resulting in the loss of real-time data, file system damage. How to improve anti-power-down capabilities of embedded systems, improve anti-jamming performance of software, usually available from hardware power supply design, the choice of storage devices, file system design are considered, the following terms for talk related to these the reliability of data storage technology.

3.1 Hardware Power Supply

Hardware reliability is the basis of the entire system reliability, hardware reliability design of power supply design is used to achieve an important means of power-off protection, the so-called reliability design is "a system in a certain environment, in the given scheduled within the time required to complete certain functions of probability ", so the" reliability "is relatively certain working conditions and scope, if present in the system failure, not affecting the normal function of the completion, the system is still reliable. Reliability of the system depends on many factors, such as system design, component quality, operational environment.

In general, power monitoring through a power protection circuit to achieve [2], the DC power source in parallel on a micro-capacitor, when the power failure after a very short period of time, due to electrical capacitance of the storage function, can in a very short time for the system power supply monitoring circuit detects when the power-down event, send out signals at the same time cause CPU interrupts, power-off protection to protect the scene immediately interrupt program, shut down the system work, such work must be the power supply voltage drop to the threshold to complete. This way you can effectively protect the system from being damaged, the data is not lost. Capacitor size can be calculated, for example, generally speaking, 3.3V down to 3.1V power supply when the power supply to ensure that the system is still normal, but lower than 3.1V can not be guaranteed, so the power supply voltage threshold should be reduced to the normal exit before the operating system, assuming that shut down the system time is 0.5 seconds, current is 1A in the case of the size of the required reserve of 2.5 farad capacitor.

3.2 Storage options

Another key consideration is the choice of storage devices, storage devices, including volatile storage device (VRAM) and non-volatile storage (NVRAM), we know the dynamic random access memory (DRAM) and static random access memory (SRAM) are all volatile storage device, which is characterized by small size, access speed, but lost power after the data. In recent years the traditional EEPROM and Flash RAM (memory) is non-volatile storage device, characterized by storing the data will not be lost down and the external interference. As the mainstream of current Flash memory NVRAM as a safe, fast memory bank, because of its small size, high capacity, low cost, power-down data is not lost, and a series of advantages, has become embedded in the main carrier of data and programs [3].
For many of the core of embedded data storage security products are available in both Flash memory, which can be a read-only, used to store the kernel and core data, and the other Flash generated during operation for storing data. This power-down events in case of Flash on only the second data may be destroyed, this time by the first Flash to be recovered or repaired. It should be noted that the current situation can not be different in a Flash partition to protect the files read-only attribute set, since the underlying Flash drive is distributed throughout the space, the operating system level of the partition and can not really protect the file system. According to relevant technical reports, the next generation of Flash will fill this gap.

There is also a practical embedded system design approach often adopted, that is, Flash, stored in a compressed form of the kernel file backup, when every time you start the Flash file to extract the kernel and associated RAM, and from the RAM, guidance system, because every time you start off as a clean file system, file system, so this approach will not be power-down events. The need to store data in the case of Flash, this is an excellent approach.

Because Flash memory has a certain life, and their life is to erase the number of terms, such as the largest NOR Flash endurance in the hundreds of thousands of times, NAND Flash can reach the maximum endurance million times, so take the right, good method is also used to ensure the reliability of a data storage means, such as to minimize the number of erase. Flash is a block (block) as a unit for management and access, therefore, should try to erase the average number of all blocks to achieve the "loss of balance" in the revised Flash when the data can be taken to modify the way the replacement will be need to modify the data written to the Flash area on the other piece, and then the original data block can be set to null and void, so the need for frequent changes of data can reduce the duplication of the same region caused local flash shortened life expectancy [4].

3.3 File System Design

File system is most vulnerable to embedded systems damaged part of the operating system uses a different file system types are different, the power-down protection has not the same, but basically provide the appropriate mechanism for fault detection and maintenance. Embedded systems are generally not directly transplanted common file system, which requires its own file system, for two main reasons, one is the common file system design considerations for lack of reliability of data storage, easy to pass off by the sudden and illegal plug and other factors; the second is the file system records general information (such as FAT table) needs to be frequently modified, which for embedded systems will lead to frequent flash memory read and write fixed block, thus shortening the life of flash.

Embedded Linux EXT2 file system used for the power-down events are intolerable, when the Linux system encounters a non-normal shut down, the damage can easily result in the file system, because this problem belongs to the operating system level, as an ordinary hard disk and Flash memory will exist. Red Hat Linux 7.2 Linux version is basically following the EXT3 file system support, EXT3 file systems can effectively solve this problem, EXT3 is EXT2 with a log on the basis of function, so it has better security, EXT3 file system power-down events in case of only the most recent changes will be lost, the entire file system will not damage. So for the Linux operating system, using EXT3 file system is to ensure the reliability of data storage is an effective means.

JFFS (Journaling Flash File System) is designed specifically for embedded systems NOR Flash memory device based journaling file system, which provides better than EXT2 file system crash / power-down security, when you need to change a small amount of data, EXT2 file system will copy the entire sector into memory and merged into the new data memory write back of the sector. The JFFS needs can change at any time (not rewrite) the whole sector, but also has a crash / power-down security protection. JFFS file system is a "supplementary" type of file system, new data is always appended to the last write data back. This "additional" type of structure to the natural realization of the above mentioned "loss of balance."

YAFFS (Yet Another Flash File System) is designed specifically for embedded NAND Flash file system for large capacity storage devices, is the log structured file system that provides a loss of balance
and power-down protection mechanism that can effectively reduce. For these reasons the file system consistency and integrity.

4 Conclusion

To ensure the reliability of embedded systems data storage method there are many, such as the use of back-up battery, RAM, and a low-power application-level software protection and so on. This article lists several embedded system design is frequently used methods. Embedded system design process can also be combined with various means of approach to the reliability design, embedded system to better improve data storage reliability. Embedded system data storage and management has become an important research topic, as the microelectronics and semiconductor memory technology, increasing development, embedded technology, constantly updated, the reliability of embedded memory technology on research in the field will be the emergence of new opportunities and challenges.

References

[1] The SNIA IOTTA Repository. http://iotta.snia.org/.
[2] L.N. Bairavasundaram, G.R. Goodson, S. Pasupathy, and J. Schindler. An analysis of latent sector errors in disk drives. In ACM SIGMETRICS, pages 289—300, 2007.
[3] B. Schroeder, S. Damouras, and P. Gill. Under-standing latent sector errors and how to protect against them. In 8th USENIX FAST, 2010.
[4] B. Schroeder and G. Gibson. Disk failures in the real world: What does an MTTF of 1,000,000 hours mean too you? In 5th USENIX FAST, 2007.
[5] N. Mi, A. Riska, E. Smirni, and E. Riedel. Enhancing data availability through background activities. In 38th Annual IEEE/IFIP DSN, 2008.
[6] T. Schwarz, Q. Xin, E. L. Miller, D. D. E. Long, A. Hospodor, and S. Ng. Disk scrubbing in large archival storage systems. In IEEE 12th MASCOTS, 2004.