Design and application of storage scheduling algorithm based on non-volatile memory

November 4, 2022

Abstract

With the accumulation of unbalanced development between traditional memory and processors, the performance of traditional memory has seriously lagged behind the performance of processors. Non-volatile memory has attracted much attention because of its potential to solve the problem that outdated traditional memory makes it difficult for high-performance processors to function. Since the number of writes to non-volatile memory is limited, once the number of writes to a memory unit reaches the upper limit, the performance of the entire non-volatile memory will degrade, and the reliability will be reduced. This paper innovatively proposes a memory scheduling algorithm based on file mapping structure, which can effectively make the pages in the file system achieve wear leveling without affecting the functions of the original file system.Three contents are included in this paper: (1) Improve the original file mapping structure and use the file mapping structure to realize the function of monitoring page writing with low overhead. (2) Improve memory scheduling algorithm, which can use the write monitoring provided by the file mapping structure to perform low-cost memory scheduling and effectively balance the page writing. (3) Implement a prototype (Wear-Leveling Hash File Mapping, WHFM) of the system and test on real non-volatile memory.This paper uses the unmodified file mapping structure as a baseline and test on a real non-volatile memory device. Experiments show that the system proposed in this paper can reduce the maximum number of page writes by 94%, and its time overhead is less than 3%.

1 Introduction

The emerging nonvolatile memory promises to open up entirely new memory systems and narrow the gap between memory and processor. Traditional memory technologies, such as SRAM and DRAM, are unable to meet the memory needs of technology scale due to low density, high standby power consumption, poor reliability and other problems. In order to solve these problems, a variety of emerging Non-Volatile Memory (NVM) technologies are proposed to replace SRAM/DRAM for the future memory hierarchy design because of its high density, zero standby power consumption, fast access speed, non-volatile and other advantages. These memories include spin moment transfer random access memory (STT-RAM), phase change random access memory (PCRAM), resistive random access memory (ReRAM), ferroelectric random access memory (FeRAM), magnetoresistive random access memory (MRAM), etc. These NVM storage media, It has the advantages of ultra-low power consumption, high density, low cost and non-volatile. NVM makes it possible for memory systems to pursue high performance, scalability, and new features.

The emerging nonvolatile memory technology blurs the line between main memory and storage. They have proven to be fast, persistent, and byte-addressable, and are widely considered a candidate replacement or complement to DRAM in main memory. Although NVM presents a great opportunity to reduce the persistence overhead of data center applications, the innate nature of write durability limits its potential. When the number of writes to the physical NVM address exceeds the maximum threshold, a single cell error may occur due to wear and tear, resulting in a crash of the entire memory. Therefore, wear equalization algorithms need to be carefully designed to extend the NVM life.

Foreign research on non-volatile memory wear equalization algorithm started early, among which Moinuddin et al proposed a phase-change memory (PCM) based Start-Gap wear equalization algorithm. It uses only two registers to reduce the maximum number of writes to the physical page by combining with simple address space randomization techniques. Start-Gap uses an algebraic mapping of logical addresses to physical addresses and avoids tracking write-per-line counts. Start-Gap performs wear equalization by periodically moving each line to its adjacent location, regardless of the line's write traffic. It consists of two registers: Start and Gap, and an additional memory line, GapLine, for easy data movement. Gap tracks the number of relocated rows in memory, and Start tracks the number of relocated rows in memory. It also includes two simple address space randomization schemes: Random Reversible Binary (RIB) matrix and Feistel network, which, combined with physical relocation, constitute an efficient wear equalization algorithm.

Although the domestic research started later than the foreign research, many effective wear equalization algorithms have been proposed. Chen et al. proposed a wear equalization algorithm based on memory allocation. In order to support Wear leveling, a wear-aware Fine-grained Allocator (WAFA) was proposed for the space management of NVM. First, WAFA divides a memory page into multiple memory units. Once an NVM page is taken apart, the unit information of the page is persisted in the last unit for fault recovery. Second, a Clockwise Best-Fit (CBF) policy was devised to allocate the units of each page in a rotating manner, by which the storage units in a page are worn out as evenly as possible. Huang et al. designed the Quail write monitor and SET wear equalization algorithm through pure software method, and used the monitoring write tool in Quail to use the SET wear equalization algorithm to balance the page write, which only needed to link the dynamic library on the system. Specifically, SET wear balancing algorithm uses a page mapping manager to provide the number of page writes, adjusts the mapping relationship of pages by logical linked list sorting of page writes, and exchanges the mapping relationship between pages with more write times and pages with fewer write times, so as to achieve the effect of write balance.

The WAFA algorithm proposed by Chen et al. has a strong wear balancing effect on the fine-grained distributor. However, its wear balancing effect is more reflected in the internal wear balancing of the page, and it cannot be fully effectively used in other systems. The Quail write monitor and SET wear balancing algorithm proposed by Huang et al can automatically monitor write and schedule memory to balance page write by linking dynamic library on the basis of no modification of software code. It can be easily embedded in the system by linking dynamic libraries, but there is still room for improvement in the cost of the whole algorithm and the effect of wear balancing. Moinuddin et al. proposed the Start-Gap wear balancing algorithm, which can realize memory wear balancing through a small amount of hardware and randomization algorithm. The Start-Gap algorithm does not keep track of the number of writes per page, so its registrie-based page swapping is not as efficient, it has more page swapping, which incurs more time overhead.

As mentioned above, although there are a number of storage scheduling algorithms based on non-volatile memory, there are few effective algorithms designed specifically for the characteristics of file systems. This paper innovatively proposes a memory scheduling algorithm based on file mapping structure, which can effectively make the pages in the file system wear balance without affecting the function of the original file system. The wear-balanced File Mapping structure (WHFM) proposed in this paper improves the original file mapping structure and comes up with a new memory scheduling algorithm to Leveling the page write leveling effectively. It also can be tested with real loads. We analyze its wear equalization effect and time cost to evaluate the effectiveness of the system.