ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology
Vol. 2, Issue 6, June 2015

Addressing the Wear-Leveling in Storage Class Memory

Shashank Dhananjaya[1] , Syeda Misba[2]

Assistant Professor, Dept. Of Information Science & Engineering, NIE, Mysuru, Affiliated to VTU Belagavi, India[1]

Assistant Professor, Dept. Of Information Science & Engineering, VVCE, Mysuru, Affiliated to VTU Belagavi, India[2]

Abstract: Storage Class Memory is a new class of data storage device. The memory cells of the SCMs are electrically erasable and reprogrammable. To rewrite the data to a particular memory cell, it has to be erased first and then rewritten. The SCMs are given with a particular write/erase cycle value. If the write/erase operation is performed more than this specified number of times for a particular cell, then that cell will wear out and will be of no use. If this happens for the set of cells in the SCMs, then the storage capacity of the device will be reduced. The solution to this write endurance problem of SCM is wear leveling. Wear-leveling is a technique for prolonging the service life of some kinds of erasable storage media. This technique can be used to increase the life of the device. The project adopts dynamic wear-leveling technique which maps the logical block address to the physical address. When the data has to be written it writes to a free block and the map is updated. This technique has a shorter life expectancy but it is faster and less complex compared to the static wear-leveling technique.

Keywords: Hot data, Cold data, pooling, Flash translation layer.

I. INTRODUCTION

Flash memories supports system level Read/Write with low power consumption and fast speed and have greater density rate than non volatile RAM. Adoption of flash memories as a storage device in a system requires several consideration to take into the system design. The writing data into flash memories would be preceded by the deletion operations that take longer time and take place in bulks.

The methods to overcome the problems in designing flash memories is the NAND flash memory. NAND type is used to store data, which is fast in writing and cheaper per unit data. The writing operation on flash memories should be preceded by deletion operations the unit of which is greater than that of writing operations. In future by 2020, sever room power and space demands will be too high and evolution of hard-disk drive storage and flash cannot help.

There is a need of new efficient device that is Storage Class Memory (SCM)._Storage class memory (SCM) is the type of NAND flash memory and is a new class of data storage device. Storage class memories (SCMs) are playing an increasingly important role in the storage hierarchy. SCMs combines the benefits of a solid state memory which has a high performance and robustness and the archival capabilities and low cost of conventional hard disk drive (HDD). The combination of low cost and high performance makes the SCM much more impact than just the server room. Here are some of the silent Features of SCM, Nonvolatile: The information is not lost if the server crashes or loses power, Fast access time: Storage class memory has low read access time.

The introduction of the storage class memory (SCM) has led to the problem of memory cell wear out.
The existing algorithm which does not take care about the problem of cell wear out. Cell wear out can be reduced by wear leveling technique. Wear leveling is a technique for prolonging the service life of some kind of erasable storage media.

These wear leveling algorithms [1, 2, and 3] store the data in an efficient distribution method in memory. This technique can be used in SCMs to reduce the write endurance and increase the life of the device.

This project uses Dynamic wear leveling, which maps the logical block address to the physical address. It maintains the data about the free blocks. When the data has to written it writes to a free block and the map is updated. This technique has a shorter life expectancy but it is faster and less complex compared to the static wear leveling technique.

The proposed project is to design and develop an algorithm that uses the dynamic wear leveling technique to improve the wear leveling in SCMs. The efficiency of the algorithm is measured. The developed algorithm will be verified on the industry standard benchmark. Main SCM characteristics are: Wear out of blocks: The life time of the SCM is reduced, when frequent block erase operations is performed. This will lead to wear out problem. A block is said to be worn out, when it has been erased the maximum possible number of times.

The number of times that a block can be reliably erased is limited. Garbage collection: Every page in the SCM blocks as three states valid, invalid and free. Valid pages contain data that is still valid. Invalid pages contain data that is used and is no more valid. Free pages are those that are already in erased state and can accommodate new data in them.

When the number of free pages in the SCM device is low, the process of garbage collection is triggered. Since erase operations can only be done at the block level, valid pages are copied elsewhere and then the block is erased.

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 2, Issue 6, June 2015

Garbage collection needs to be done efficiently because frequent erase operations during garbage collection can reduce the lifetime of blocks.

Block pooling: There are two type pools of blocks - hot and cold. The data that is updated more frequently is defined as hot data, while the data that is relatively unchanged is defined as cold data. The blocks are initially assigned to the hot and cold pools randomly. If hot data is being written repeatedly to certain blocks, then those blocks may wear out much faster than the blocks that store cold data.

Flash translation layer: The FTL is a middleware between the host file system and SCM. FTL hides the internal organization of NAND flash memory and presents a block device to the file system layer. FTL maps the logical address space to the physical locations in the flash memory. [1] FTL is also responsible for wear leveling and garbage collection operations.

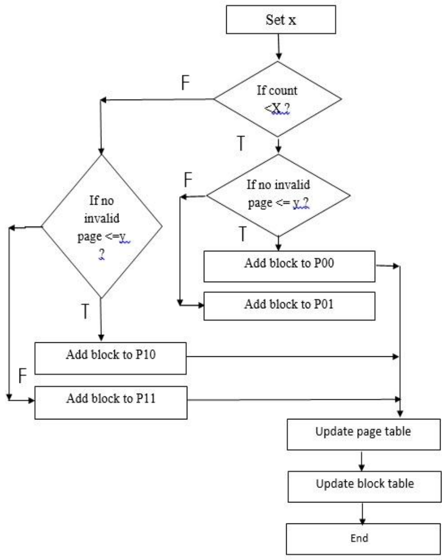

Fig 2.1 block pooling.

II. ALGORITHM

The design objective of this algorithm is to achieve wear leveling with the garbage collection.

A. Overview

As with any wear leveling algorithm the objective of proposing algorithm is to balance the wear count of the memory blocks so that it reduces the write endurance problem in the storage class memory. Storage Class Memory is a type of NAND flash memory.

As mentioned before the proposing algorithm is to balance the wear count of the blocks by mapping the hot data to the blocks with lesser wear count and cold data to more worn blocks in the memory.

This algorithm explicitly identifies the hot data and allocates it in its appropriate blocks. The hot data and cold data are defined based on the time it referred in particular logical address. These logical address will be mapped to the physical address.

We maintain the blocks in four different pools based on the wear count of blocks. The blocks present in each pool can migrate to another pool after the wear count of some blocks has been increased or when the garbage collection is done. The hot data and cold data concept in inspired from [2].

The identification of hot data and cold is done based on the access time of that particular data with reference to the logical block.

When a logical address is to be mapped to physical address, first the type of data in the logical address is identified by comparing the last access time with current access time, based on this difference the logical address will be mapped.

B. Basic Algorithm

The aim of this proposing algorithm is to map a logical address to a physical address which is performed by LBA to PBA mapping function. The flow of LBA to PBA mapping is explained in Fig 2.2 flow chart. The algorithm will map logical address to physical address based on the type of data that has to be written. The data is classified as Hot or Cold based on the *time* variable associated with the logical block. Whenever a logical address is referred the *time* count associated with it is updated. Once the logical address has been obtained, the *time* count is compared with the standard time count *a*, which is used to classify data as Hot or Cold. If the *time* is less than the value *a* then it is again compared with a value *b* which is another constant to classify the data as most frequently used or less frequently used. Based on the *time* of data compared with *a* and *b* the algorithm selects a block from the pool having the lesser wear count.

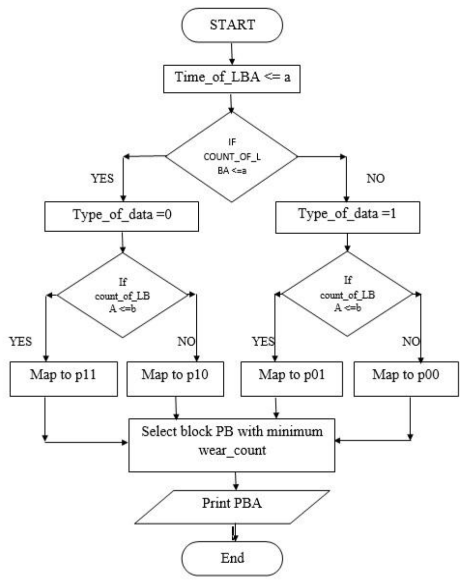

Fig 2.2 LBA to PBA mapping

Copyright to IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 2, Issue 6, June 2015

III.EVALUATION

This section discusses the overheads involved with the implementation of algorithm and evaluates the performance of algorithm via detailed experiments. Analysis of overheads. The most significant overhead of our algorithm is the extraction of block number and its count. When we write data to a file, that data will be written to the respective blocks in the Disk. When there is such a write operation taking place, we have to find out, to which block (physical blocks in the disk) the data is written to. Similar with read operation. Benchmarks are the simulation tools. It will simulate number of reads and writes for a period of time. On running this benchmark, many read and write operations will be performed.

This sequence of operations simulated by benchmark will be based on real time system behavior. The benchmark which simulates more read and write operations has to be chosen. We use industry standard file system benchmark tool called Debench and Blktrace for generating huge amount of read/ write operations and to extract block number and its count. Dbench as one of our benchmark is an emulation of the file system load function of the Netbench benchmark. It does all the same I/O calls that the smbd server daemon in Samba would produce when it is driven by a Netbench workload. Blktrace is a block layer IO tracing mechanism which provides detailed information about request queue operations up to user space. Both these tools are executed simultaneously and scripts are run to extract the block number and its respective counts.

The scripts generate file with two columns having block numbers and wear count. This file is used to generate graph using Gnuplot tool. Gnuplot is a command-line program that can generate two- and three-dimensional plots of functions, data, and data fits. Gnuplot can produce output directly on screen, or in many formats of graphic files including (PNG), Encapsulated PostScript (EPS), Scalable Vector Graphics (SVG), JPEG and many others.

IV. EXPERIMENTS

This section explains in detail our experimental setup and the results of our simulation.

The proposing algorithms is compared with the existing system. Blktrace and Debench tools were used to analyze the block allocation in the existing method. With the help of scripts the input file generated for both existing algorithm and proposed algorithm. Gnuplot is used to plot graph for both algorithms and show the efficiency and performance of our algorithm.

Benchmarks are the simulation tools. It will simulate number of reads and writes for a period of time. On running this benchmark, many read and write operations will be performed. This sequence of operations simulated by benchmark will be based on real time system behavior. The benchmark which simulates more read and write operations has to be chosen.

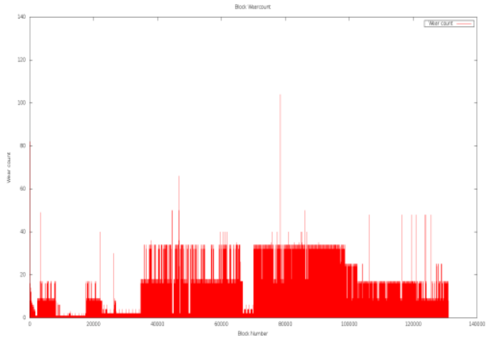

Fig 4.1 block wear count graph for existing system

When the same input file was given to proposed algorithm the graph obtained had max wear count 3. The analysis of the graph gave the following values

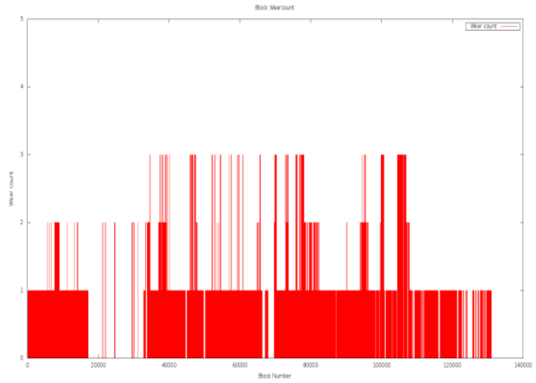

Fig 4.2 block wear count graph for proposed system

	Existing algorithm	Proposed algorithm
Mean	13.538956	0.762962
Variance	91.862404	0.474281
Standard deviation	9.584488	0.688681

In order to find the efficiency of our algorithm in wear leveling the device we scaled the input values 66667 times. It was observed that the max wear count of blocks were close to 7 lakh. It was also observed that 55.09% of blocks will wear out in 95 days

When the same scaled input was given to our algorithm, it was observed that the maximum wear count of blocks were close to 3 lakh. When compared to the existing algorithm it was observed that in 95 days the 55.09% of blocks would reach only 2 lakh. The graph had 84886 blocks whose wear count was above 1 lakh and it would take 33333 hrs (3.23 years) for all these blocks to reach 10 lakh. Assuming the wear out count to be 10 lakh, in our algorithm 64% of blocks will reach its wear count 10 lakh after 3.23 years.

V. CONCLUSION

Storage class memories (SCMs) constitute an emerging class of non-volatile storage devices that promise to be significantly faster and more reliable than magnetic disks. This project is support SCMs for the problem of memory wear out. This project aims only on the memory allocation, i.e. the mapping of logical address to physical

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 2, Issue 6, June 2015

address. As a result of classifying the data blocks in the physical memory into different pools, the algorithm proposed here is much efficient when compared with the existing methods. The algorithm designed is efficient and easy to understand that's how there is a transition from existing algorithm to new designed algorithm. There were problems in the existing algorithms which are designed to take care of the wear out problem of the storage class memory, which are taken care of the same to some extent, for the same wear out problem in the SCM a new approach is designed here as a solution and we can clearly say that the algorithm designed is more efficient and will be help full.

VI. FUTURE WORK

Every day a new type of situation will be encountered in the field of memory device which may have the new kind of problems. This project is designed take care of the existing problem of Storage Class Memory. The algorithm designed here can be enhanced and in the near future it can take a new shape.

As the future enhancement of this project, the algorithm designed can be implemented by considering the time and space complexity and it can be integrated with the kernel of the operating system. Thus many new ideas can be taken in to account and the code is capable of being altered so as to enhance the project.

REFERENCES

[1] D. M. Han: Fast Erase Algorithm using Flash Translation LayerNAND-Type Flash Memory.
[2] Rejuvenator: A Static Wear Leveling Algorithm for NAND Flash Memory with Minimized Overhead by Muthukumar Murugan University Of Minnesota Minneapolis, USA-55414 Email: murugan@cs.umn.edu and David.H.C.Du University Of Minnesota Minneapolis, USA-55414 Email: du@cs.umn.edu
[3] Using a Shared Storage Class Memory Device to Improve the Reliability of RAID Arrays Sara Chaarawi1, Jehan-François Pâris1, Ahmed Amer2, Thomas J. E. Schwarz, S. J.3, Darrell D. E. Long4
[4] Yangwook Kang, Thomas Marlette, Ethan L. Miller, Rekha Pitchumani, Muninn: a Versioning Key-Value Store using Object-based storage model, Proceeding of the 7th International Systems and Storage Conference (SYSTOR'14), June 2014.
[5] Yangwook Kang, Yang-suk Kee, Ethan L. Miller, Chanik Park, Enabling Cost-effective Data Processing with Smart SSD, *the 29th IEEE Symposium on Massive Storage Systems and Technologies (MSST 13)*, May 2013.
[6] Exploring Storage Class Memory with Key Value Stores Katelin A. Bailey Peter Hornyack Luis Ceze Steven D. Gribble Henry M. Levy University of Washington Dept. of Computer Science and Engineering Seattle, WA, USA
[7] PCM-Based Durable Write Cache for Fast Disk I/O Zhuo Liu ; Bin Wang ; Carpenter, P. ; Dong Li ; Vetter, J.S. ;Weikuan Yu Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2012 IEEE 20th International Symposium
[8] SCMFS:A file system for Storage Class Memory Xiaojian Wu ; Reddy, A.L.N. High Performance Computing, Networking, Storage and Analysis (SC), 2011 International Conference for Publication Year: 2011
[9] Energy efficient page initialization for storage class memory Fei Xia ; Dejun Jiang ; JinXiong NonVolatile Memory Systems and Applications Symposium (NVMSA), 2014 IEEE
[10] Hybrid solid-state storage system with storage classmemory and NAND flash memory for big-data application Takeuchi, K. Circuits and Systems (ISCAS), 2014 IEEE International Symposium