Study of Flash Memory its implementation application and limitation

Suman Bala Nande[1] Rakesh Patel[2]
[1]Student [2]Lecturer
[1,2]Department of Information & Technology
[1,2]Kirodimal Institute of Technology, Raigarh (C.G.), India

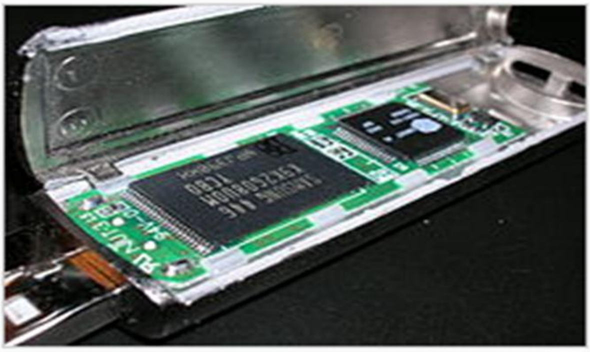
Abstract— Flash memory is used at high speed as storage of personal information utilities, ubiquitous computing environments, mobile phones, electronic goods, etc. This is because flash memory has the characteristics of low electronic power, non-volatile storage, high performance, physical stability, portability, & so on. To improve the overall performance & service life of the flash memory, a new type of file management algorithm for flash memory has been designed. A new linear ordering function is added to the traditional file management system. Its function is to balance the used frequency of every port in the flash memory. The new file should be stored in the space whose writing times are smaller, thus improving the overall efficiency, extending the service life of flash memory. The capability of the management system designed is better than the traditional one, especially in the case of being operated more times. Phase_ change nonvolatile semiconductor memory technology is based on an electrically initiated, reversible rapid amorphous-to-crystalline phase-change process in multicomponent chalcagenide alloy materials similar to those used in rewriteable optical disks. Flash memory controller is interfaced with flash memory. It handles all signals required by the flash memory. With the required signals different types of commands are also sent through flash memory controller to the flash memory. Flash memory- based solid- state disks are fast becoming the dominant, form of end- user storage devices, partty even replacing the traditional hard- disks.

Key words: Chalcogenides, nonvolatile memory, phase change, flash memory, cleaning policy, mobile device, data clustering, command decoder

I. INTRODUCTION

Flash memory is an electronic non-volatile computer storage medium that can be electrically erased & reprogrammed. Introduced by Toshiba in 1984, flash memory was developed from EEROM (electrically erasable programmable read-only memory). There are two main types of flash memory, which are named after the NAND & NOR logic gates. The internal characteristics of the individual flash memory cells exhibit characteristics similar to those of the corresponding gates. Whereas EPROMs had to be completely erased before being rewritten, NAND type flash memory may be written & read in blocks, which are generally much smaller than the entire device. NOR type flash allows a single machine word to be written _ to an erased location- or read independently.

The NAND type is primarily used in main memory, memory cards, USB flash drives, solid-state drives, & similar products, for general storage & transfer of data. The NOR type, which allows true random access & therefore direct code execution, is used as a replacement for the older EPROM & as an alternative to certain kinds of ROM applications.



A USB flash drive. The chip on the left is the flash memory. The controller is on the right.

Fig. 1:

Example applications of both types of flash memory include personal computers, PDAs, digital audio players, digital cameras, mobile phones, synthesizers, video games, scientific instrumentation, industrial robotics, medical electronics, & so on. In addition to being non-volatile, flash memory offers fast read access times, as fast as dynamic RAM, although not as fast as static RAM or ROM.

Flash memory has inherently strong points compared to traditional hard disk: non-volatility, fast access speed, shock resistance, & low power consumption. Therefore, it has been widely adopted in embedded applications such as USB flash memory, CF card memory, mobile drives, & so-on. However, due to its hardware characteristics, flash memory-based applications require special software operations while reading data from flash memory. There are two types of flash memories are available i.e. NOR based & NAND based. Different types of series & parallel NOR flash memory chips are available. Figure 1 shows the general blocks which are used in to in flash memory controller.

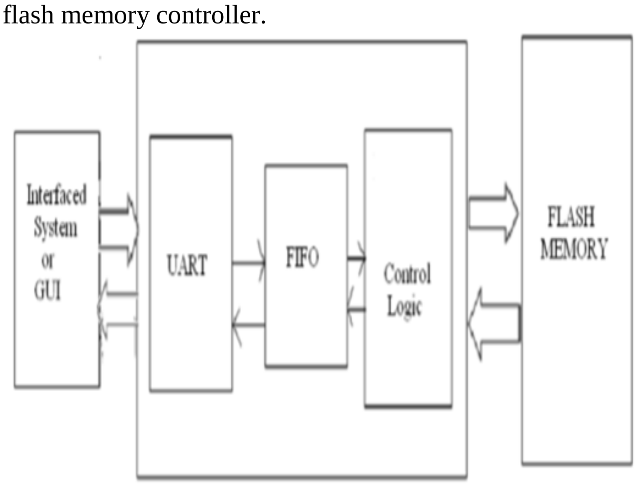


Fig. 2: General diagram of Flash memory of controller

In many practical applications, one needs to compute on data that exceeds the capacity of the main memory of the available computing-device. This happens in a variety of settings, ranging from small devices, such as PDAs, to high performance servers & large clusters. In such cases, the cost of data transfers between disk & the main memory often proves to be a critical bottleneck in practice, since a single disk transfer may be as time-costly as millions of CPU operations.

Flash memory currently has been widely used in mobile devices, consumer electronics, & embedded applications due to its attractive advantages in accessing-speed, shock-resistance, non-volatility, power-consumption etc.

II. History

Flash memory (both NOR & NAND types) was invented by Dr. Fujio Masuoka while working for Toshiba circa 1980. According to Toshiba, the name "flash" was suggested by Masuoka's colleague, Shoji Ariizumi, because the erasure process of the memory contents reminded him of the flash of a camera. Masuoka & colleagues presented the invention at the IEEE 1984 International Electron Devices Meeting (IEDM) held in San Francisco.

Intel Corporation saw the massive potential of the invention & introduced the first commercial NOR type flash chip in 1988. NOR-based flash has long erase & write times, but provides full address & data buses, allowing random access to any memory location. This makes it a suitable replacement for older read-only memory (ROM) chips, which are used to store program code that rarely needs to be updated, such as a computer's BIOS or the firmware of set-top boxes. NOR-based flash was the basis of early flash-based removable media: Compact Flash was originally based on it, through later cards moved to less expensive NAND flash.

NAND flash has reduced erase and write times, & requires less chip area per cell, thus allowing greater storage density & lower cost per bit than NOR flash, it also has up to ten times the endurance of NOR flash.

III. Principle

A. Floating-Gate Transistor:

In flash memory, each memory cell resembles a standard MOSFET, except the transistor has two gates instead of one. On top is the control gate (CG), as in other MOS transistors, but below this there is a floating gate (FG) insulated all around by an oxide layer. The FG is interposed between the CG & the MOSFET channel.

B. Internal Charge Pumps:

Despite the need for high programming & erasing voltages, virtually all flash chips today require only a single supply voltage, & produce the high voltages using on-chip charge pumps.

C. NOR Flash:

In NOR gate flash, each cell has one end connected directly to ground, & the other end connected directly to a bit line. This arrangement is called "NOR flash" because it acts like a NOR gate. When one of the word lines is brought high, the corresponding storage transistor acts to pull the output bit line low.

D. NAND Flash:

NAND flash also uses floating-gate transistor, but they are connected in a way that resembles a NAND gate: Several transistors are connected in series, & the bit line is pulled low only if all word lines are pulled high.

E. Vertical NAND:

Vertical NAND (V-NAND) memory stacks memory cells vertically & uses a charge trap flash architechture. The vertical layers allow longer areal bit-densities without requiring smaller individual cells.

IV. Applications

A. Serial Flash:

Serial flash is a small, low-power flash memory that uses a serial interface, typically Serial Peripheral Interface Bus (SPI), for sequential data access. When incorporated into an embedded system, serial flash requires fewer wires on the PCB than parallel flash memories, since it transmits & receives data one bit at a time. This may permit a reduction in board space, power consumption, & total system cost.

B. Firmware Storage:

With the increasing speed of modern CPUs, parallel flash devices are often much slower than the memory bus of the computer they are connected to conversely, modern SRAM offers access times below 10 ns, while DDR2 SDRAM offers access times below 20 ns.

C. Flash Memory As A Replacement For Hard Drives:

One more recent application for flash memory is a replacement for hard disks. Flash memory does not have the mechanical limitations & latencies of hard drives, so a solid-state drive (SSD) is attractive when considering speed, noise, power consumption, & reliability.

D. Flash Memory As RAM:

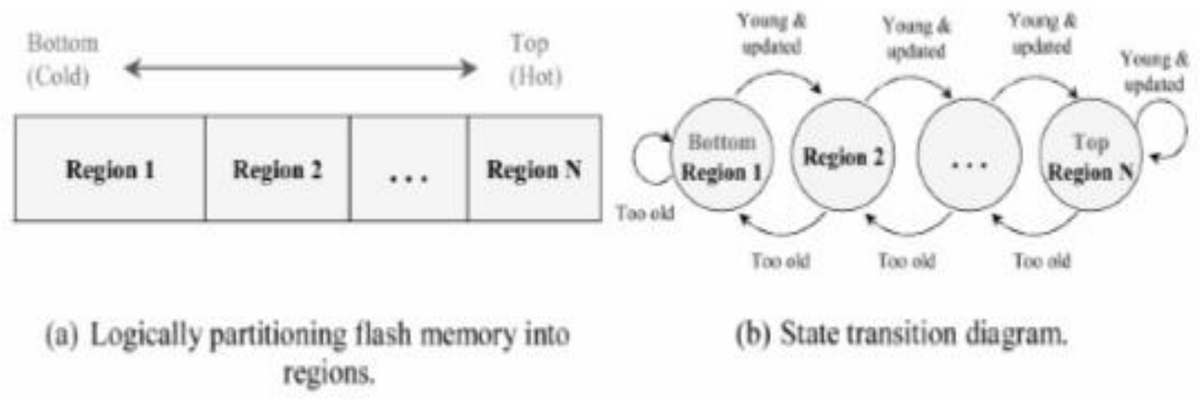
As of 2012, there are attempts to use flash memory as the main computer memory, DRAM. In this role, it is slower than conventional DRAM, but uses up to ten times less power & is also significantly cheaper.

E. Archival Or Long-Term Storage:

It is unclear how long flash memory will persist under archival conditions- i.e., benign temperature & humidity with infrequent access with or without prophylactic rewrite.

V. Flash Memory Management Using Data Clustering

Data reorganization by separating hot data from cold data can reduce cleaning overhead. Previous research reorganizes data only at cleaning time when migrating valid data in the segment that is being cleaned. We propose a new data reorganization method: DAC (Dynamic Data Clustering) approach. This approach dynamically clusters data not only during segment cleaning, but also during data update. This is motivated by the fact that when data blocks are updated, they are updated to another free flash space. Then hot data & cold data can be separately clustered at this time by updating them to separate flash memory spaces.



Data blocks are moved toword the Top region if their update frequencies increase, whereas they are moved toward the Bottom region if their update frequencies decrease. So regions can be dynamically shrunk or enlarged.

A state machine is used for the region switching. The state machine contains several states & the state transition diagram is shown in above figure (b). Each data block is associated with a state indicating the region it insides in. The starting state is 'Bottom region', where newly created data blocks reside.

VI. Design & Implementation of a Flash Memory Server

A flash memory server providing the DAC data ilustering was implemented. The server manages flash memory as fixed-size blocks. Every data block is associated with a unique logical block member.

A. *Data Layout On Flash Memory:*

Each segment has a segment header to record segment information such as the number of times the segment has been erased, per block information array, etc. The per-block information array describes every block in the segment, such as logical block number, region number, the number of times the block has been updated, flags indicating free, valid, or obsolete, etc.

B. *Block-Based Translation Table:*

Each table entry contains a region number indicating which region the block belongs to & a timestamp indicating when this block was allocated in the region.

VII. Industry

One source states that, in 2008, the flash memory industry includes about US$9.1 billion in production & sales. Other sources put the flash memory market at a size of more than US$20 billion in 2006, accounting for more than eight percent of the overall semiconductor market & more than 34 percent of the total semiconductor memory market. In 2012, the market was estimated at $26.8 billion.

VIII. Reliability issues

Reliability issues for flash electrically erasable programmable read-only memories are reviewed. The reliability of both the source-erase type flash memory & the NAND structure EEPROM are discussed. Disturbs during programming, write/erase endurance charge loss of both devices are reviewed, & the reliability of the tunnel oxide & the interpoly dielectric are described. It is shown that bipolarity F-N programming / erase, which is used in the NAND EEPROM, improves the charge to breakdown & decreases the stress –induced leakage current.

IX. Limitations

A. *Block Erasure:*

One limitation of flash memory is that although it can be read or programmed a byte or a word at a time in a random access fashion, it can only be erased a "block" at a time. This generally sets all bits in the block to 1. Starting with a freshly erased block, any location within that block can be programmed. However, once a bit has been set to 0, only by erasing the entire block can it be changed back to 1. In other words, flash memory offers random-access read & programming operations, but does not offer orbitary random – access rewrite or erase operations. For example, a nibble value may be erased to 1111, then written e.g. as 1110. Successive writes to that nibble can change it to 1010, then 0010, & finally 0000. Essentially, erasure sets all bits to 1, & programming can only clear bits to 0. File systems designed for flash devices can make use of this capability, for example to represent sector metadata.

B. *Memory Wear:*

Another limitation is that flash memory has a finite number of program – erase Cycles. Most commercial available flash products are guaranteed to wishstand around 100000 P/E cycles before the wear begins to deteriorate the integrity of the storage.

1) *Read Disturb:*

The method used to read NAND flash memory can cause nearby cells in the some memory block to change over time. This is known as read disturb.

C. *X-Ray Effects:*

Most flash ICs come in ball grid array (BGA) packages, & even the ones that do not are often mounted on a PCB next to other BGA packages. After PCB Assembly, boards with BGA packages are often X-rayed to see if the balls are making proper connections to the proper pad, or if the BGA needs network.

X. Conclusion

Flash memory is expected to be largely used as the increase of capacity & the decrease of price. Effective cleaning policies help to maximize the flash memory lifetime, improve system performance, & reduce power consumption. Flash memory lifetime is thus extended, system throughput is largely improved & flash memory is evenly worn. In examining the gegree of wear- leveling & exploring the impacts of flash memory utilization, flash memory size & degree of locality of reference, the proposed method all performed well. Several factors are important in determining how well the DAC data clustering will work in a given environment, such as the configuration for the number of states in the DAC state machine and the setting of the time threshold for state switching. In our experience, these factors are highly dependent on work load. We also noticed that different segment. Segment selection algorithms perform differently for the same setting of factors.The proposed method can not only be in flash memory, it can also be used in other applications that can benefit from data clustering or need segment cleaning. Flash memory offers attractive features for storage of data, such as non-volatility, shock resistance, fast access speed, & low power consumption. However, it requires erasing before it can be overwritten. The erase operations are slow & consume comparatively a great deal of power. Furthermore, flash memory can only be erased a limited number- of times. To overcome hardware limitations, we use the non-in-place update mechanism that requires a cleaner to reclaim space occupied by obsolete data.According to operation requested by interfaced system the respective command sequence & signals are generated at the output of flash memory controller. If invalid command is received then flash controller doesn't generate any signals. For valid command the data & command related information stored in FIFO.

REFERENCES

[1] B. Dipert & M. Levy, Designing with Flash Memory Amabooks, 1993.
[2] P. Torelli, 'The Microsoft flash file system', Dr. Dobb's Journal,62-72 (February 1995).
[3] R. Dan & J. Williams, 'A True FFS & Flite technical overview of M- systems flash file systems', 80-SR-002-00-6L Rev. L.30. http://www.m- sys.com / tech 1.htm, March 1997.
[4] M. L. Chiang & R. C. Chang, 'Cleaning policies in mobile computers using flash memory', Journal of Systems & software accepted.
[5] M. Rosenblum & J. K. Outerhout, 'The design & implementation of a log- structured life System', ACM Trans. Computer Systems, (10), 26-52 (February 1992).
[6] M. Resenblum, 'The design & implementation of a log- structured file sys.' , PhD Thesis, University of California, Berkeley, June 1992.
[7] Parallel flash memory information http://www.micron.com/products/nor-flash
[8] Vitter, J. S.: Algorithms & data structures for External Memory, now Publishers (2008).