SMART® Modular Technologies

White Paper

Flash Management – Why and How?

A detailed overview of flash management techniques

Esther Spanjer
Director, SSD Technical Marketing
SMART Modular Technologies

November 2009

TABLE OF CONTENTS

INTRODUCTION

The inherent nature of NAND flash technology requires sophisticated flash management techniques to make it a practical storage medium for computing systems. Challenges intrinsic to using NAND flash in a solid state drive (SSD) include:

- Need to erase before writing
- Wear out mechanism that limits service life
- Data errors caused by write and read disturb
- Data retention errors
- Management of initial and runtime bad blocks

With proper flash management techniques, these characteristics of NAND flash can be managed to provide a highly reliable data storage device.

Five significant factors influencing reliability, performance, and write endurance of a solid state drive are:

- Use of Single Level Cell (SLC) vs Multi Level Cell (MLC) NAND flash technology
- Wear-leveling algorithms
- Ensuring data integrity through Bad Block management techniques
- Use of error detection and correction techniques
- Write amplification

Implementation of sophisticated flash management techniques that, properly balanced for the characteristics of each generation of NAND flash, deliver flash SSDs with high reliability, long service life, high performance, and excellent data integrity characteristics.

This white paper provides an overview of NAND flash technology, its intrinsic characteristics, and explains how proper flash management techniques address specific NAND issues to create reliable flash SSDs with a long service life.

NAND FLASH TECHNOLOGY

NAND flash is a nonvolatile solid state memory with the capability to retain stored data when unpowered. NAND and NOR are the two fundamental flash architectures used in electronic systems. Both NOR and NAND Flash memory were invented by Dr. Fujio Masuoka in 1984 [1]. Toshiba and Samsung introduced the first commercial NAND flash chip in 1989.

NAND flash offers faster erase and write times and up to ten times the write endurance when compared with NOR flash [2]. It requires a smaller chip area per cell (compared to NOR), thus allowing greater storage densities and lower cost per bit. NAND flash achieves these advantages by sharing some of the common areas of the storage transistor through strings of serially connected transistors. NOR devices require additional control circuits to independently access each storage transistor for random, independent addressability.

NAND flash is accessed on a block basis, making NAND flash unsuitable to replace random access NOR flash which can be accessed on a byte basis. For example, the Execute In Place (XIP) ability of NOR flash allows microprocessors to execute programs directly through its random access capability.

NAND flash access is similar to other block-oriented storage devices such as hard disks and optical media, and therefore is very suitable for use in mass-storage devices such as memory cards, solid state drives, and USB flash drives.

Today, two NAND flash technologies, SLC (Single-Level Cell) and MLC (Multi-Level Cell), service different applications. Section 0 explains in detail the differences between these two technologies.

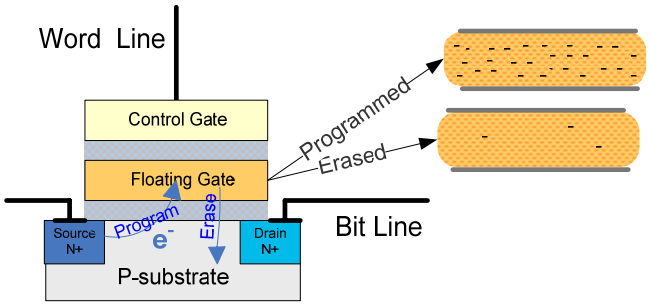
NAND Flash Cell

The basic NAND flash cell is a floating gate transistor with the bit value determined by the amount of charge trapped in the floating gate. NAND flash uses *tunnel injection* for writing/programming and *tunnel release* for erasing the cell [3]:

- Writing (i.e. programming) to a cell causes the accumulation of negative charge in the floating gate, resulting in a "0" bit value for that cell.
- Erasing a cell removes the negative charge in the floating gate, resulting in a "1" bit value for that cell. To change the bit content of a cell from "0" to "1", the cell must be erased. Due to the NAND architecture of sharing bit control lines across multiple storage transistors, erasing a cell requires erasing the entire Erase Block which contains that cell.

Figure 1 below shows the architecture of a NAND flash cell.

Figure 1: NAND Flash Cell Architecture [3]

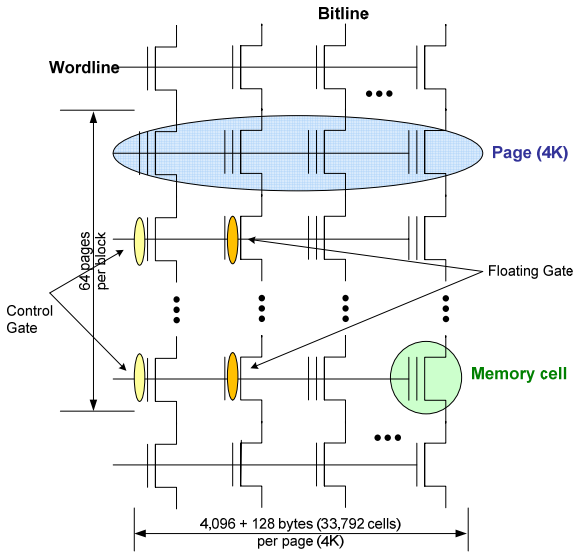


NAND Flash Architecture

NAND flash memory stores the information in an array of floating-gate transistors, i.e. memory cells, combined into bit and word lines. The serial cell architecture of NAND explains the device name. NAND (Not AND) is the Boolean logic reference to how information is read out of these cells. Each single level cell (SLC) transistor stores one bit of data, and each multi-level cell (MLC) NAND stores multiple bits of data in each cell.

Figure 2 below shows the NAND flash architecture of an 8Gb SLC 50nm flash [4], where 16,896 cells are located on the same bitline to create a 4KB page. An Erase Block consists of 64 pages, each occupying its own wordline.

Figure 2: NAND Flash Architecture (8Gb SLC 50nm)

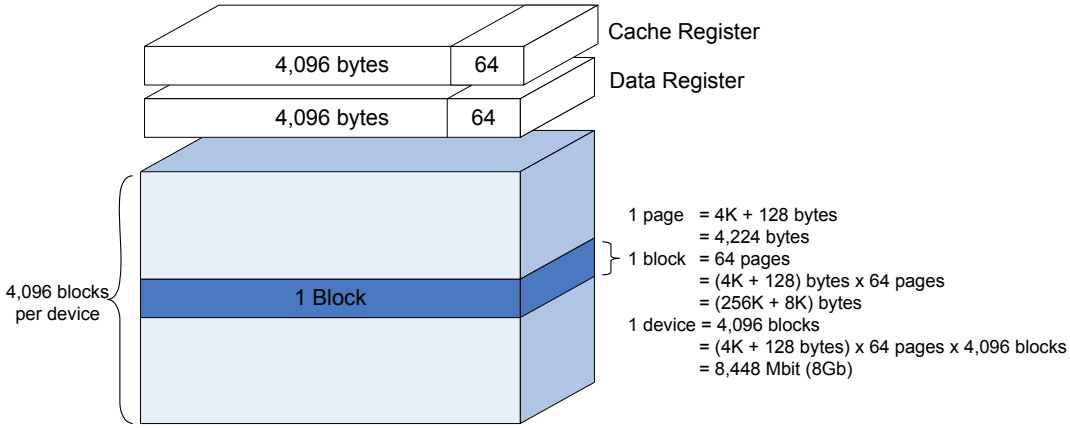


Erase Blocks and Pages

A page is the smallest area of the flash memory that supports a write operation and consists of all the memory cells on the same wordline. An Erase Block is the smallest area of the flash memory that can be erased in a single operation. Page and block sizes differ per manufacturer and flash generation.

- 50nm 8Gb SLC flashes contain 4KB page size and 256KB block size, as shown in Figure 3 below [5].
- 50nm 16Gb MLC flash contains 4KB page size and 512KB block size.

Figure 3: NAND Flash Block/Page structure (8Gb SLC 50nm)



A 4KB page size corresponds to 4,096 bytes that are dedicated for data and 128 bytes that are available for control and ECC information.

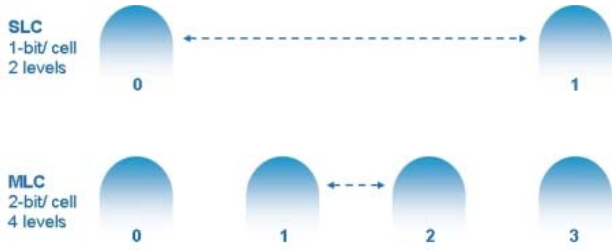
SLC vs. MLC NAND Technology

Multi-Level Cell (MLC) NAND and Single-Level Cell (SLC) NAND offer capabilities that serve two very different classes of applications – those requiring the lowest cost-per-bit, and those demanding higher performance and reliability.

MLC NAND flash allows each memory cell to store multiple bits of information, compared to the one bit per cell for SLC NAND flash. As a result, MLC NAND offers a larger capacity, twice the density of SLC, and at a cost and reliability point targeted for consumer products such as cell phones, digital cameras, USB drives and memory cards.

SLC NAND provides faster write speed, lower error rate and longer write endurance, making it the better fit for applications that require high reliability, performance, and viability in multi-year service life.

Figure 4: SLC vs. MLC NAND Technology [6]



As shown in Figure 4 above, both SLC- and MLC-based devices use the same size voltage window. The separation between adjacent voltage levels in MLC is much smaller than in SLC flash. This reduced separation affects data reliability and performance, due to the following causes:

- Electrons stored in adjacent levels can be disturbed with electrical noise and can shift from one level to another, causing a higher bit error rate

- Write and read performance is reduced by as much as 40-50%, since charging the cells to the correct voltage levels requires a more accurate process with many iterations
- Power consumption increases due to the extended read and write efforts
- Flash write endurance and data retention suffer due to the additional stresses from the extended read and write efforts. Specifications for 50nm MLC NAND show 10,000 write/erase cycles, and this will go down to 5,000 or even 1,500 write/erase cycles for 3xnm generations. SLC NAND specifications show 100,000 write/erase cycles for existing and future generations. Some vendors have recently introduced Enterprise Grade SLC and MLC NAND Flash. The Enterprise Grade Flash increases the P/E cycles up to 300,000 for SLC and up to 30,000 for MLC NAND Flash. The higher P/E cycles are achieved by relaxed programming/erase times and lower data retention rates.

INHERENT NAND FLASH CHALLENGES

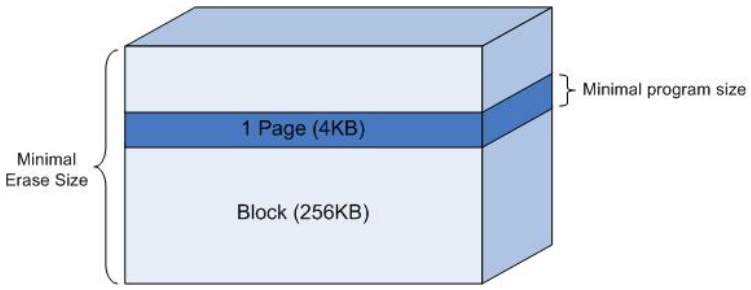
NAND flash suffers from reliability, write endurance, and data retention problems that require complex management solutions. Due to the increased density with each new flash generation, these problems are increasingly becoming more apparent, requiring more and more sophisticated NAND flash controllers and system solutions. NAND flash has the following intrinsic limitations:

- Need to erase before writing
- Wear out mechanism that limits service life
- Data errors caused by write and read disturb
- Data retention errors
- Management of initial and runtime bad blocks

Erase Before Write

The nature of NAND flash requires it to be read and written one page at a time, and erased a "block" at a time, as shown in Figure *5* below.

Figure 5: Page Write vs. Block Erase



The Erase operation sets all the bits in the block to a “1”. Starting with a freshly erased block, any page within that block can be written. Once written to a “0”, the only way to reset a bit to a “1” is by erasing the entire block. To minimize the number of block erases and maximize flash service life, sophisticated block management techniques eliminate unnecessary erase operations,

Read/Write Disturb

NAND flash is prone to *bit flips;* cells that are not meant to be accessed during a specific read or write operation can change contents due to read and write activities in adjacent cells or pages. MLC NAND is much more susceptible to bit flips than is SLC NAND.

Figure 6: Read/Write Disturb NAND Flash [4]

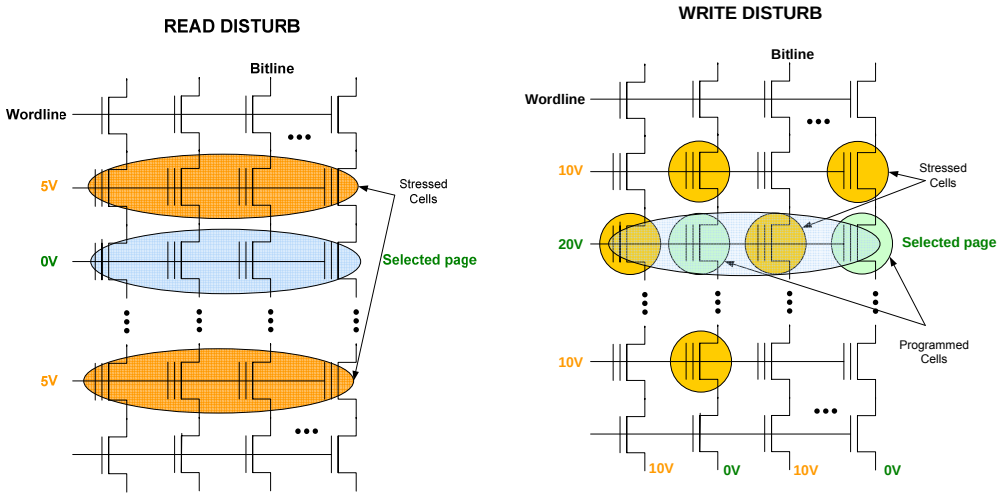


Figure 6 above shows the two causes for bit flips:

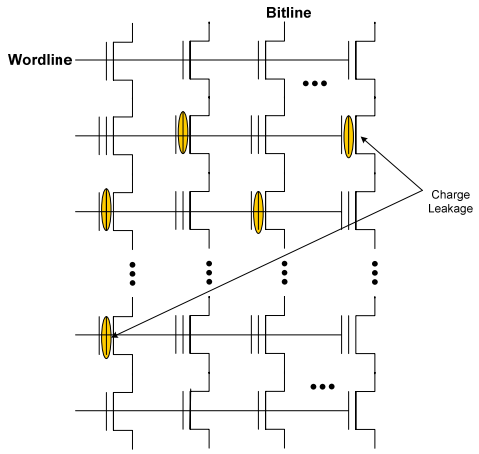
- **Read Disturb:** A read disturb occurs when a cell that is not being read receives elevated voltage stress. Stressed cells are always in the block that is being read and are always on a page that is not being read. The probability of read disturb is much lower than is a write disturb;
- **Write disturb:** A write disturb occurs when a cell that is *not* being programmed receives elevated voltage stress. Stressed cells are always in the block that is being programmed and can be either on the page that is programmed (but cell was not selected), or on any page within the same block.

Erasing the cell resets the cell to its original state, eliminating the data and, consequently the data errors which resulted from the read or write disturbs. An ECC mechanism in the data flow path detects bit flips and corrects them before providing the data to the host. As flash cell geometries decrease and more cells are placed onto wafers, the probability of errors and bit flips increases and NAND flash controllers require more powerful error detection/correction (EDC/ECC) algorithms.

Data Retention Errors

Data retention defines how long the written data remains valid within a memory device. Programmed NAND flash cells must retain a stable voltage level to ensure data retention for an acceptable period, typically defined as 10 years. Charge leakage from the floating gate, called charge drift, tends to slowly change the cell's voltage level from its initial level to a different level, as shown in Figure 7 below. This new level may incorrectly be interpreted as a different logical value.

Figure 7: Data retention errors through charge leakage [4]



The data retention time is inversely related to the number of Write/Erase cycles, which means that blocks that have been erased many times have a shorter data retention life than blocks with lower Write/Erase cycles. MLC flash data retention is orders of magnitude lower than SLC flash.

To achieve an acceptable bit error rate, an appropriate ECC mechanism needs to be implemented to detect data retention errors and correct them before providing the data to the host.

Bad Blocks

There are two types of bad blocks in a NAND flash device:

- **Initial Bad Blocks**: Due to production yield constraints and the pressure to keep costs low, NAND flash devices ship from the factory with a number of bad blocks. NAND flash manufacturers specify that up to 2% of the SLC flash can contain bad blocks; the number for MLC flash is about 5%;
- **Accumulated Bad Blocks:** Due to multiple write/erase cycles, trapped electrons in the dielectric cause a permanent shift in the voltage levels of the cells. When the voltage level shifts enough, this will be observed as a read, write, or erase failure.

Bad Block management is required to map out both the initial Bad Blocks, as well Bad Blocks that were accumulated during device operation.

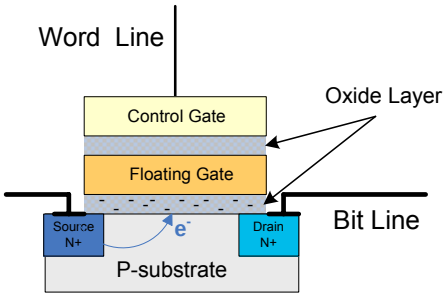
Limited Numbers of Writes

NAND flash memory has a finite number of Write/Erase cycles, caused by two reasons:

- Electrons that are trapped (i.e. trap-up) in the thin oxide layer that insulates the floating gate;
- Break down of the oxide structure, due to hot carrier injection [3].

Once the damage to the oxide layer is large enough, it becomes increasingly difficult for electrons to travel between the P-substrate and the floating gate, as shown in Figure 8 below. The Erase Block encompassing the oxide layer cannot be erased properly with the standard threshold voltages and needs to be retired and added to the pool of Bad Blocks.

Figure 8: Flash wear out - electrons cannot pass the oxide layer



The number of Write/Erase (W/E) cycles that NAND flash manufacturers specify for their flash devices is an indication of the expected wear out of the oxide layer. Today's SLC NAND Flash devices are guaranteed for 100,000 Write/Erase cycles per block. MLC flash provides 1,500-5,000 Write Erase cycles for 3xnm generation.

This number does not mean that the erase block will not be functional after this threshold is reached, but is merely an indication that the flash cells have "aged" and may start showing signs of wear-out at a more rapid rate. Many flash blocks can live much longer than their specified Write/Erase cycle limit.

With each process shrink, flash manufacturers are facing challenges to maintain the same number of write/erase cycles. For example, Samsung requires a 1-bit ECC implementation on its 50nm NAND SLC flash devices [5] to be able to maintain the 100,000 W/E cycles. MLC flash is even worse off, requiring 4-bit ECC to sustain 10,000 P/E cycles for 50nm MLC and up to 24-bit ECC for 3xm MLC flash.

Flash management techniques, such as Wear Leveling, Error Correction, and Bad Block management are required to overcome and manage the flash wear out limitation.

OVERCOMING FLASH LIMITATIONS

Flash Management Techniques

Proper flash management techniques must incorporate mechanisms to overcome the limitations inherent to NAND Flash. Typically, a combination of hardware and software solutions is used in a solid state drive to manage and overcome the NAND flash limitations. The five most important factors/techniques that play a role in achieving the most reliable, highest performance, and longest life span are listed in Table 1 below.

Table 1: Effect of Flash Management Techniques

Flash Management Technique	Reliability	Write endurance
Flash Media used (SLC or MLC)	√	√
Wear Leveling (Static/Dynamic)		√
Write Amplification		√
Error Detection and Correction	√	√

Flash Management Technique	Reliability	Write endurance
Bad Block Management	√	

When comparing different solid state drive solutions, it should be noted that the flash controller design, the algorithms used and flash media are the three most influential factors.

Wear Leveling

Wear Leveling ensures even distribution of erase operations on all blocks within the NAND flash. That is, each block within the NAND flash is erased and written approximately the same number of times as every other block within the drive.

To understand Wear Leveling, one needs to understand the different addressing schemes in a system. The operating system (OS) uses Logical Block Addressing (LBA) to read and write a block of data from the drive; the flash controller uses physical addresses on the flash to read and write data.

Wear Leveling is based upon two mechanisms:

- The controller has the ability to map an LBA address to different physical locations on the flash. The controller uses a mapping table to keep track of the relationship between the logical block and the physical address
- The presence of spare blocks on the flash for replacement of blocks that contain invalid data

Updated or new data is written to an available free block. The block that contains old data is erased in the background and then marked as a free block. This block rotating technique ensures even wear of memory blocks across the flash device. The Wear Leveling process is transparent to the operating system.

Write Amplification

Write amplification factor is the amount of data a flash controller has to write in relation to the amount of data that the host controller wants to write. The higher the write amplification, the quicker the solid state drive will run out of available blocks for program/erase operations. A write amplification factor of 1 means that the host wanted to write 1MB and the flash controller also wrote 1MB to the flash. Some flash controllers are able to achieve write amplification of less than 1, due to mirroring and data dedupliciation techniques.

Error Detection and Correction

One of the key factors to increase flash reliability and write endurance is the implementation of an Error Detection and Correction mechanism. The three most popular Error Correction algorithms that are used with NAND flash technology today are:

- **Reed-Solomon:** Invented in 1960; used in HDDs, CDs, NAND, and telecommunication and digital broadcast protocols. The algorithm is based on symbols; up to 8 errors can be detected per byte, as long as the bits fall within the symbol boundary. This algorithm is used mainly for MLC NAND to handle the bit-flipping phenomenon.
- **Hamming:** Invented in 1950, named after Richard Hamming [7]. This algorithm can detect and correct single-bit errors, and can detect (but not correct) double-bit errors. This code is mainly used for SLC NAND Flash.
- **BCH (Bose, Ray-Chaudhuri, Hocquenghem)**: BCH codes were invented in 1959 by Hocquenghem, and independently in 1960 by Bose and Ray Chaudhuri [8]. It can handle both random and burst errors and is commonly used for SLC NAND Flash.

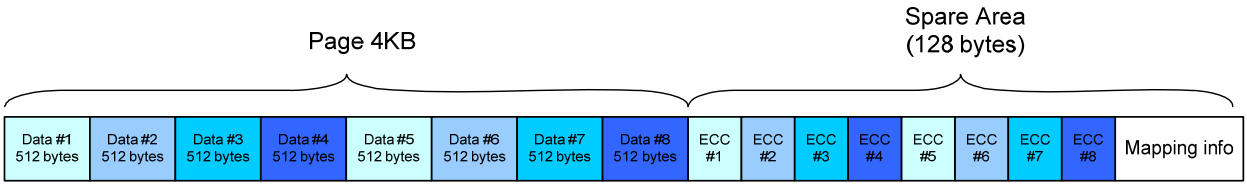
The maximum number of errors that can be detected and corrected is determined by the algorithm. Future NAND flash generations will require powerful error correction mechanisms due to their reduced reliability.

No matter what algorithm is used within the ECC engine of a NAND flash controller, the method of error detection and correction remains the same:

1. Every time a page of data is written to the flash, data is passed through the controller's ECC engine to create a unique ECC signature
2. The data and ECC signature are stored together on the flash; the data in the page area, the ECC signature in the spare area
3. When reading the data back, both data and stored ECC signature are read into the controller. A new ECC signature is generated, based upon the read-back data
4. The newly created ECC signature is then compared to the original stored ECC signature. If both signatures are the same, no errors have occurred, and the data will be provided to the host. If the two signatures differ, the data is corrected by the controller before being provided to the host

Some flash controllers will write the corrected data back to the flash media to optimize reliability, while others will not, since there is no guarantee that the data will not show errors again in the future. For 4KB page flash, typically 8 ECC signatures are created when writing data to the flash; one for each 512 bytes of data, as shown in Figure 9 below.

Figure 9: Four ECC signatures for 4KB page



Implementing an ECC mechanism improves the overall reliability of the flash device, as read, write and data retention errors are caught and corrected. Less known is the fact that a strong ECC engine is one of the most important factors to increase the life span of a flash device. When blocks start to age, more and more errors will occur on that block. When the ECC engine is not able to detect these errors, a hard "ECC" error occurs and the block is retired. The more powerful the ECC engine, the more "life" can be squeezed out of a block (even though it shows increasing failures) and the longer the overall lifespan of the flash drive.

SMART Modular's XceedIOPS SATA drives include an ECC engine, based on the Reed Solomon algorithm, which is able to detect and correct up to 12 9-bit symbols. Reed Solomon is a symbol-based algorithm, whereby symbols correspond to byte boundaries. As long as the errors occur within the symbol boundaries, up to 108 contiguous bits of data can be corrected per 4KB page.

Bad Block Management

To improve yield and lower cost, all NAND devices are shipped from the factory with some bad blocks which are identified and marked accordingly by the manufacturer. The first physical block (block 0) is always guaranteed to be readable and free from errors.

Throughout the lifespan of the flash, additional blocks are marked as bad once they show un-recoverable errors during write/program operations. The write will be done to another block and these blocks are retired and added to the pool of Bad Blocks.

The controller's firmware uses a Bad Block Table to map both initial and accumulated bad blocks and to make sure they are not used in any reading or writing operation. This not only ensures data integrity, but also enhances performance by eliminating the need for repeated write operations resulting from data being repeatedly mapped to the same Bad Block.

Flash manufacturers guarantee that no more than 2% of SLC flash will become bad throughout the 100,000 write/erase cycle lifespan of the flash device.

FLASH WRITE ENDURANCE

By implementing flash controllers that include a strong Wear Leveling and error correction mechanism, it is possible to extend the lifespan of the flash drive far beyond the specified 100,000 Write/Erase SLC

cycles specified by the flash device manufacturers. In general, SSD Life Expectancy depends on the following factors:

- P/E (Program & Erase) Cycles
- Drive Capacity
- Write Speed
- Drive Duty Cycle
- Read/Write Ratio
- Write Amplification

The following formula can be used to create comparable approximations of SSD Life Expectancy:

$$\textit{SSD Life (in sec)} = \frac{(\boldsymbol{P/E\ Cycle}) \boldsymbol{x}\ (\boldsymbol{Capacity})}{(\boldsymbol{Write\ Speed}) \boldsymbol{x}\ (\boldsymbol{Duty\ Cycle}) \boldsymbol{x}\ (\boldsymbol{Read/Write\ Ratio}) \boldsymbol{x}\ (\boldsymbol{Write\ Amp})}$$

Error! Reference source not found. below shows write endurance calculation results for the XceedIOPS SATA drives, based on the following assumptions.

1. Program/Erase Cycle for Enterprise grade MLC: 30,000
2. Sequential Write Speed: 250MB/s
3. Duty Cycle: 40%
4. Read/Write Ratio: 2:1
5. Write Amplification: 0.5

Table 2: XceedIOPS SATA Life Expectancy Calculation

Capacity	SSD Total Life
100GB	4.2 years
200GB	8.3 years
400GB	16.8 years

Please note that other factors influence the service life of a solid state drive, such as Mean Time Between Failure (MTBF). These calculations are intended to illustrate that SSD Life Expectancy is not the significant life-limiting factor of the flash SSD in an application.

CONCLUSION

NAND flash is a complex technology and is becoming more and more complex with each geometry shrink. Its intrinsic challenges require sophisticated flash management techniques, both in hardware and software.

The four main factors that influence the reliability, performance and write endurance of a solid state drive are:

- Flash media used (SLC vs. MLC)
- Implemented Wear Leveling algorithm
- Bad Block management
- Strength of ECC engine

Calculating SSD Life Expectancy is not trivial and depends on many assumptions, such as P/E cycles, file size, and file system overhead. While generic write endurance calculations can be made based upon these assumptions, many applications require a more in-depth discussion with the flash vendor.

The XceedIOPS SATA drives are designed for maximum reliability and write endurance. With a combination of Enterprise Grade MLC flash, static and dynamic Wear Leveling, and a powerful ECC engine, the XceedIOPS SATA drives provide among the highest reliability and write endurance available in the industry.

REFERENCES

[1] Dr. Fujio Masuoka, *A new flash E2PROM cell using triple polysilicon technology*. Proceedings of Electron Devices Meeting, 1984 International, Volume: 30, pages: 464- 467

[2] Toshiba, *NAND vs. NOR Flash memory*, http://www.toshiba.com/taec/components/Generic/Memory_Resources/NANDvsNOR.pdf

[3] Jitu J. Makwana and Dr. Dieter K. Schroder, *A Non-Volatile Memory Overview*, http://aplawrence.com/Makwana/nonvolmem.html

[4] Jim Cooke, *Inconvenient Truths of NAND Flash*, Flash Memory Summit, August 2007

[5] Samsung, *K9F8G08U0M NAND Flash Data Sheet*, page 8.

[6] www.Linuxdevices.com, *Opening the door for the latest NAND flash in open source mobile platforms,* Francois Kaplan, Sep 2006

[7] Richard W. Hamming, *Error Detecting and Error Correcting Codes*, Bell System Technical Journal 26(2):147-160, 1950.

[8] R. Bose and D. Ray-Chaudhuri, *On a class of error correcting binary group codes, Information and Control*, vol 3 p68-79, 1960

About SMART Modular

SMART Modular Technologies is a leading provider of memory products, offering more than 500 standard and custom products to top-tier OEMs in the computer, industrial, networking, and telecommunications sectors. Taking innovations from the design stage through manufacturing and delivery, SMART has developed a comprehensive memory product line that includes DRAM, SRAM, and Flash in various form factors. SMART also offers high performance, high capacity solid-state drives for enterprise, defense/aerospace, industrial automation, medical, and transportation markets. SMART's Display and Embedded Products Group designs, manufactures and sells thin film transistors (TFT) liquid crystal display (LCD) solutions to customers developing casino gaming systems as well as embedded applications such as kiosk, ATM, point-of-service, and industrial control systems. SMART's presence in the US, Europe, Asia, and Latin America enables it to provide its customers with proven expertise in international logistics, asset management, and supply-chain management worldwide. More information on SMART may be obtained at www.smartm.com.