SCHWEITZER ENGINEERING LABORATORIES, INC.
WHITE PAPER

# NAND Flash Memory Reliability in Embedded Computer Systems

Ian Olson

## INTRODUCTION

NAND flash memory—named after the NAND logic gates it is constructed from—is used for nonvolatile data storage in digital devices. The different types of NAND flash memory target different applications and have significantly different costs and longevity. This paper addresses the two main reliability concerns in NAND flash memory (data retention and endurance), what type of NAND flash memory is best suited for embedded computer systems, and how to optimize computer systems for maximum reliability of NAND flash memory.

### History

In the late 1990s, NAND flash memory first began to be used in consumer products such as USB flash drives (also known as thumb drives) and digital cameras. Even though the cost was very high compared to other storage media, such as floppy disks, rotating hard drives, and CD-RWs, the size, durability, and power efficiency of flash memory opened a new world of portable data storage.

Due to its rapidly increasing capacity and decreasing price, flash memory became a viable primary storage medium by 2005 for embedded computer products like first-generation SEL computers (e.g., the SEL-3351 System Computing Platform). At that time, most flash memory was available in the form of memory cards such as CompactFlash®, SD, and xD cards, which were typically used in digital cameras and portable MP3 music players. The 2.5-inch solid-state drives were also available and offered the greatest capacity, but cost prevented them from seeing widespread use until years later.

The SEL-3351 used the CompactFlash form factor as a primary storage device due to its large capacity and widespread availability. High-capacity for its time, the 8 GB industrial-grade CompactFlash card used in SEL computers in 2005 cost over $1,000 compared to $100 to $150 for the same size consumer-grade CompactFlash. Industrial-grade cards were pricier for many reasons, including higher quality manufacturing processes, more rigorous testing and screening, temperature trials, and low market demand compared to consumer-grade devices.

Today, flash memory is the most common data storage medium and is used in nearly every computing and intelligent electronic device. While advancements in capacity and cost are slowing down due to technical limitations, there is currently no competing technology positioned to overtake flash memory anytime soon.

Though improvements in flash manufacturing technology continue to decrease cost and increase storage capacity, the industrial-grade flash memory storage devices used in SEL computers still cost significantly more than similar consumer-grade devices. One of the biggest reasons is storage density, or how much data the different types of NAND flash can store on a single flash chip.

## Types of NAND Flash Memory

Most NAND flash memory is categorized by how many bits of information are stored in a single memory cell. Flash memory stores data by charging flash cells to a specific level, and it reads data by measuring the level of charge on the cells. Single-level cell (SLC) memory stores one bit per cell; a high charge indicates a binary 0 while a low charge indicates a binary 1. Multi-level cell (MLC) stores two bits per cell, dividing the same charge range into four regions, each corresponding to the binary pairs 11, 10, 01, and 00.

Additional multi-bit types of flash memory further subdivide the charge range to store additional bits into each cell. For example, a triple-level cell (TLC) stores three bits by dividing the voltage range into eight regions and a quad-level cell (QLC) stores four bits using 16 regions. For each additional bit, the number of required charge subdivisions doubles, yielding exponentially diminishing returns.

In addition to the number of bits per cell, NAND flash has evolved through different chip topologies to improve storage capacity and cost. Older technology uses a planar or 2D topology, meaning all memory cells are arranged in a single layer on each flash chip. Newer technology uses a 3D topology, which stack layers of memory cells on top of each other within a single chip. Current 3D NAND technology contains dozens or even hundreds of layers in each chip as manufacturers continue to advance their production systems.

Almost all consumer-grade products use one of the multi-bit types of flash memory (at the time of writing, 3D TLC and 3D QLC are most common) because their higher data density yields lower cost-per-gigabyte than SLC. This creates a snowball effect where SLC flash technology is less popular, which reduces its production volume, which further increases its cost. When considering the cost differences of consumer-grade versus industrial-grade and multi-bit versus SLC memory, the combined effect leads to a large cost disparity between industrial-grade SLC flash devices and consumer-grade flash devices.

# Failure Mechanisms in NAND Flash

The most commonly known failure mechanism in flash memory is related to the overall operational lifespan of the device. Writing and overwriting data to the flash memory result in program/erase (P/E) cycles. P/E cycles create a trapped charge in the NAND flash cells, which reduces the margin for bit errors to occur. As this trapped charge accumulates over time, the bit error rate eventually becomes so high that the error-correcting code (ECC) cannot compensate, making the flash memory no longer useable.

The number of P/E cycles the flash memory can endure over its lifespan while still providing reliable data storage is called endurance. Endurance is tested using a process defined by Joint Electron Devices Engineering Council (JEDEC) standards, and it is typically specified as a number of terabytes written (TBW) the flash memory device can endure and still provide adequate endurance for the specified workload. Client workloads require one year of powered-off data retention at 30 degrees Celsius while Enterprise workloads require three months of powered-off data retention at 40 degrees Celsius [1]. While this is a very useful measurement, it only represents a single data point on what is actually a three-dimensional plane. Often overlooked, the two other dimensions (temperature and data retention) are not fixed points, they vary for each system in operation, and are critical in embedded and industrial applications.

## Data Retention

In flash memory, data retention is the measure of how long the integrity of data can be guaranteed after being written to the flash memory without suffering from data corruption. Fundamentally, each NAND flash memory cell can be in one of two states: programmed or erased. To program a cell, its electrical charge is precisely increased to a threshold level (Vt). Erasing a cell removes all charge, which enables the cell to be programmed again. Once a flash cell is charged, the electrons stored in the cell slowly leak across the NAND gate, causing the charge on the cell to decrease over time. With enough leakage, the charge level on the cell will drift into the neighboring region, causing the incorrect binary value to be read (a bit error).

In SLC flash, the cells are programmed to a level near the maximum Vt, leaving a very large margin for the charge to degrade before a bit error occurs. MLC flash utilizes that margin area to create four data value regions, doubling the data density to two bits per cell but greatly reducing the margin for bit errors. TLC flash further subdivides the programmed state to create eight regions to gain another 50 percent increase in data density (three bits per cell) but reducing the margin even further.

Figure 1, Figure 2, and Figure 3 illustrate how SLC, MLC, and TLC flash store data by charging the cell to the level that correlates with the desired data/bit value(s). The shaded regions represent the probability curve of a proper cell charge level, and the red arrows show the relative margin for bit errors.

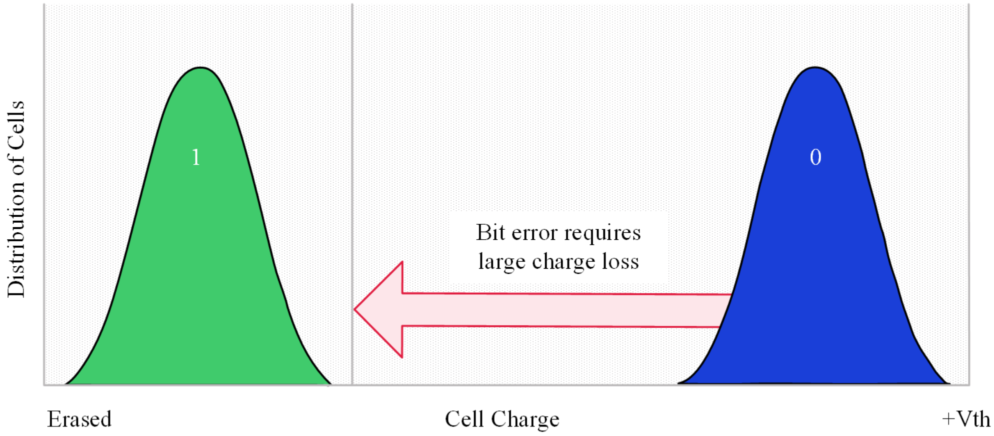

Figure 1 SLC Flash Data Storage

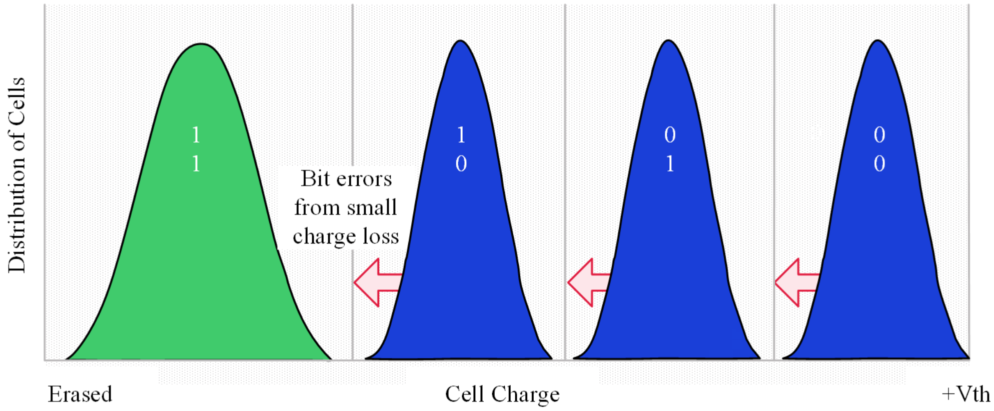

Figure 2 MLC Flash Data Storage

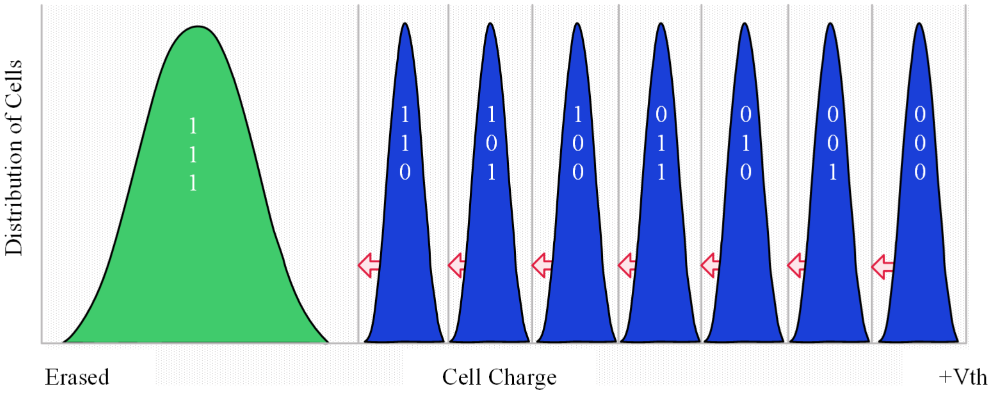

**Figure 3 TLC Flash Data Storage**

Data retention is a function of the rate of charge loss and the amount of margin between charge regions. The charge loss is strongly driven by the temperature of the flash memory, and the margin is affected by the type of flash (SLC, MLC, TLC, etc.) and the number of P/E cycles the flash memory has endured.

## A Complete View of Data Integrity

Assuming SLC NAND flash specifications of 1 year of data retention at 55 degrees Celsius after enduring 100,000 P/E cycles, and using the Arrhenius equation [2] to calculate the acceleration factor relative to temperature, we can estimate the effect of temperature on data retention. Additionally, using formulas from JEDEC and some empirical data, we can also estimate the increase in data retention when the flash memory is below the maximum number of P/E cycles. Finally, we can combine these two curves into a set of curves to estimate the data retention of SLC flash memory given the average number of P/E cycles it has endured and the average operating temperature.

The graphs in Figure 4 are hypothetical. In practice, data retention at extremely low P/E cycles may not be as high as indicated because the mathematical function approaches infinity at zero P/E cycles.

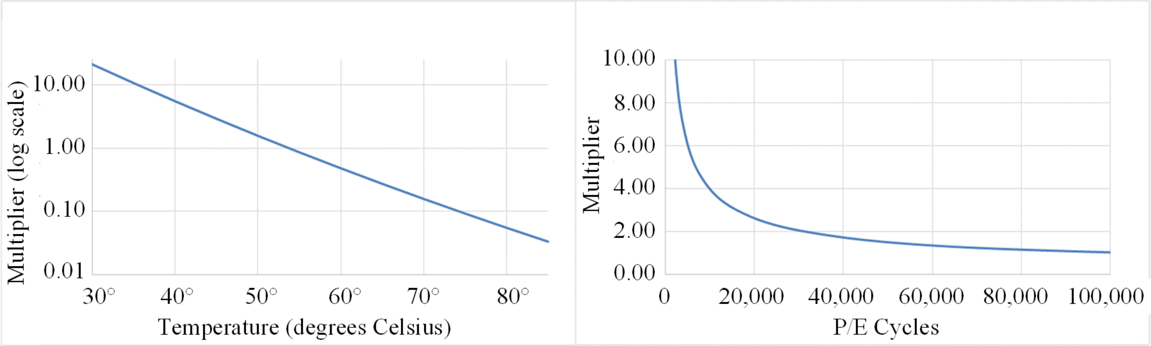

**Figure 4 SLC Flash Data Retention Relative to Temperature (left) and P/E Cycles (right)**

Figure 5 shows that a lightly used (1,000 P/E cycles) SLC flash memory operating at 55 degrees Celsius may have around 16 years of data retention. At higher temperatures, however, the data retention is substantially lower: 6 months or less at 85 degrees Celsius.

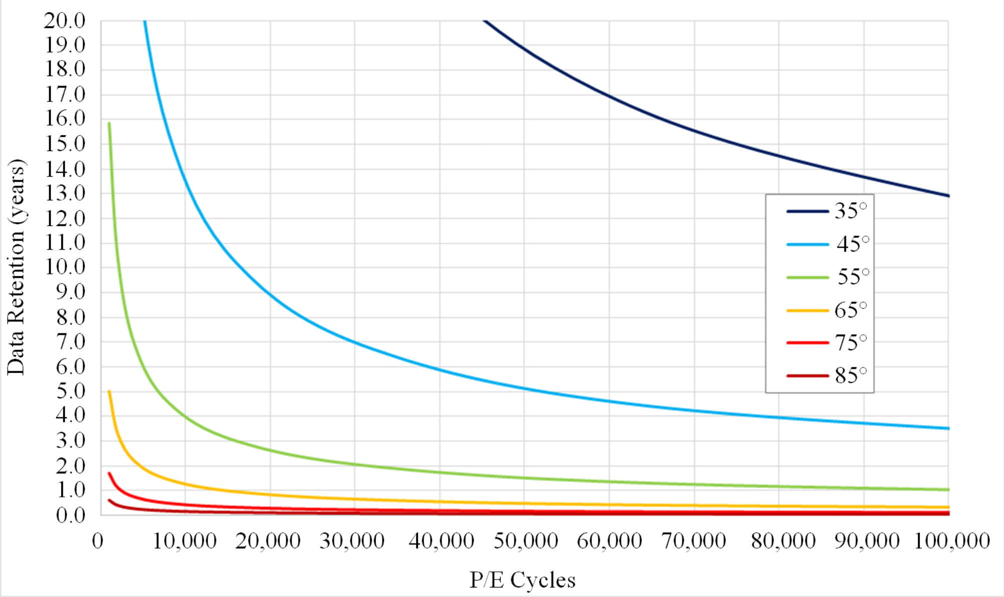

**Figure 5 SLC Flash Data Retention Relative to P/E Cycles at Different Temperatures**

Figure 6 compares SLC to MLC at 55 degrees Celsius.

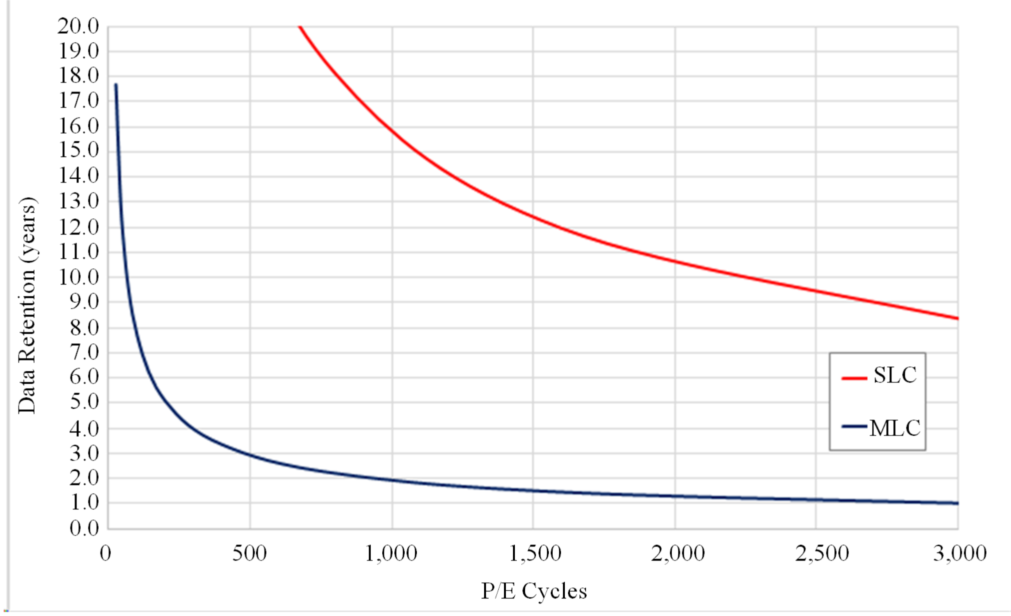

**Figure 6 Comparison of MLC and SLC Flash Data Retention at 55 Degrees Celsius**

Although MLC flash memory has much lower endurance than SLC, it may still be considered acceptable for applications that are write-protected or that write very little data to the flash memory. However, when operating in a warm environment, the MLC flash memory system could fail due to data corruption within the expected operational life of the system. In less severe conditions, the MLC flash memory may provide acceptable reliability, but the point remains that for any given temperature and P/E cycle count, SLC flash memory provides approximately ten times the data retention and over 30 times the endurance of MLC.

### Cold Operation

Flash memory is negatively affected by cold temperatures. While data retention is incredibly good at low temperatures, the ability to accurately charge a flash cell decreases rapidly as temperatures drop below 10 degrees Celsius. This means that data written while the flash memory is warm fit the data retention model discussed previously, but retention is reduced for data written while the memory is cold. This problem affects both types of flash, but it is more severe in MLC, making it less suitable than SLC in cold applications as well.

### pSLC Flash

Pseudo-SLC (pSLC) storage devices use lower-cost MLC flash memory, but effectively store only one bit per cell like SLC flash. The flash controller treats the erased state (normally 11) as 1 and any programmed/charged state (00, 01, 10) as 0. Using the MLC flash in this way improves the margin against bit errors and also increases endurance by using a less stringent program/erase process. Available at a slightly higher cost, pSLC offers significantly better endurance (typically 20,000–30,000 P/E cycles) than standard MLC. This makes pSLC a good middle-ground alternative to the more expensive SLC flash memory.

## Improving Reliability With System Design

While the problem of data retention can be a concern even for SLC flash memory, the demonstrated reliability of embedded and industrial computer systems over the last decade indicates that SLC is indeed an excellent storage medium for these applications. In addition, there are ways to design the embedded system to minimize P/E cycles, maintain data retention, and monitor flash memory health, further improving the reliability of flash memory.

### Minimize P/E Cycles

The number of P/E cycles can be minimized by avoiding disk-write activity unless necessary for the application. For example, virtual memory in the operating system (OS) can be disabled. Virtual memory creates a page or swap file on the system drive and accesses it like random-access memory (RAM), with substantial amounts of read and write operations. Embedded systems typically have a static application load. If the system is designed with the appropriate amount of system RAM, virtual memory is completely unnecessary, and it can actually decrease performance and determinism if left enabled.

Some operating systems include a RAM drive overlay or write-filter, which caches all disk-write operations in RAM until a commit operation is executed. The commit operation writes the cached data all at once to the flash memory, minimizing the effect of periodic small drive-write operations that are unavoidable in many applications.

### Maintain Data Retention

Although minimizing P/E cycles is the best first step, it is beneficial to not completely eliminate all P/E cycles. The controllers used in flash memory devices have wear leveling algorithms, which spread the write operations across all flash cells evenly, to prevent any one flash cell from wearing out before all of the rest. Writing small amounts of data to the flash memory over time will cause the wear leveling feature to eventually rewrite all data stored in the flash memory, resetting the data retention clock, and ultimately improving overall data retention (even though it is moderately increasing the P/E cycles).

A computer running a typical Windows or Linux® OS will have enough disk activity from background tasks and logging to not require this kind of upkeep. However, an OS with a RAM drive overlay or write-filter protection could benefit from a background task writing small amounts of data to a partition that is not protected.

Many modern flash memory controllers already have background integrity scans built into the controller, making this type of upkeep automatic and completely transparent to the OS and application software. An example of this already in use is read-disturb monitoring. Whenever a flash cell is read, the charge of nearby flash cells in the same block or page can be disturbed, eventually causing data corruption. When the flash memory controller detects this corruption during a read command, its read-disturb feature will automatically rewrite that block of data. More advanced controllers include a patrol-read background process, which reads the entire logical contents of the drive on a slow periodic interval and re-writes data that has correctable bit errors before further degradation makes them uncorrectable.

### Monitor Status

Most flash memory storage devices that operate on mass storage interfaces (such as CompactFlash, CFast, and 2.5-inch solid-state drives) provide status information through the Self-Monitoring, Analysis and Reporting Technology (S.M.A.R.T.) interface. Although much of the information available through S.M.A.R.T. is vendor-specific, the interface usually provides the operating temperature of the storage device and often includes data for the average number of P/E cycles. Some flash memory storage devices even provide a percent estimate of the remaining flash memory life. Also useful in predicting imminent storage device failure is the spare block count, which typically starts to decrease rapidly as the storage device nears the end of its useful life.

Status monitoring could be incorporated into the background task mentioned previously. The task could monitor the operating temperature and P/E cycle count on the storage device to dynamically adjust the rate that the flash memory is written to, optimizing P/E cycles versus data retention at any temperature.

### Overprovision

Probably the easiest way to maximize data integrity on any system using flash memory is to purchase a larger capacity storage device. Due to the wear leveling feature built into the flash memory controller, a larger storage device takes longer to wear out because the P/E cycles are spread across more flash cells. This results in both longer flash memory life and longer data retention. While using the largest storage device available is often not economically viable, using one that is at least one size larger than necessary for the application is money well spent.

## Conclusion

While flash memory has known weaknesses, it has proven to be far superior in reliability and performance to its predecessor: rotating hard drives. Even though the initial cost of SLC flash memory is considerably higher than MLC variants, the long-term cost is significantly less in high-endurance applications. MLC flash can corrupt and lose data within the operational lifetime of an embedded computer system with moderate or high data workloads, even if the computer is not run in extreme temperatures. In the event of a drive failure, the cost to replace the drive, recover lost data, and reconfigure the computer typically exceeds the cost of a properly selected high-endurance flash storage device. This is why SEL recommends carefully selecting a flash storage device that meets the long-term needs of the application. SEL strives to offer the most reliable storage technologies available in order to minimize maintenance and total cost of ownership and to maximize reliability and availability.

For guidance in selecting the appropriate flash storage device for your application and calculating storage device lifetime estimates, refer to SEL Application Note AN2016-03, “Determining Solid-State Drive (SSD) Lifetimes for SEL Computing Platforms.”

## References

[1] JEDEC standard JESD218, Solid State Drive (SSD) Requirements and Endurance Test Method; JEDEC standard JESD219, Solid-State Drive (SSD) Endurance Workloads.

[2] JEDEC standard JEP122G, Failure Mechanisms and Models for Semiconductor Devices.

## Biography

**Ian Olson** works as a lead application engineer for Schweitzer Engineering Laboratories, Inc. (SEL) in the automation platforms group. Ian joined SEL in 2005 to support customers with communications, protocols, and system integration. Now, as a lead application engineer, Ian helps manage the growth, direction, and technology of computer and automation products. Ian graduated from the University of Idaho in 2004 with a bachelor’s degree in computer engineering.

SCHWEITZER ENGINEERING LABORATORIES, INC.
2350 NE Hopkins Court • Pullman, WA 99163-5603 USA
Tel: +1.509.332.1890 • Fax: +1.509.332.7990
www.selinc.com • info@selinc.com

*LWP0015-01*