# Quantifying Reliability of Solid-State Storage from Multiple Aspects

Hairong Sun, Pete Grayson, and Bob Wood
*Fusion-IO*
*2855 E. Cottonwood Parkway, Suite 100*
*Salt Lake City, UT 84121*
*{hsun,pgrayson,bwood}@fusionio.com*

## Abstract

*Reliability and data retention of Solid-State Storage have multiple facets which should be traded off carefully. This paper presents a framework of reliability quantification deployed in Fusion-io®, allowing us to design Fusion-io® products more competitively regarding reliability by trading off various reliability aspects. It discloses a comprehensive set of test results with NAND flash parts from a variety of vendors and different NAND technologies (3x nm, 4x nm, and 5x nm). The sample size in our tests is the largest published so far and the set of reliability quantification is one of the most comprehensive in the industry. Based on the test data, exponential growth models are proposed to characterize the impact of P/E cycles and read disturb to Raw Bit Error Rate.*

## 1. Introduction

Although NAND flash device has been used extensively in consumer electronic devices, such as USB drives, mp3 players, digital cameras and cellular phones, its entry into enterprise data centers faces FUD (Fear, Uncertainty and Doubt) due to its reputation as being unreliable and the limited P/E (Program/Erase) cycles it can endure in its life time.

As the impressive performance of NAND flash device in tier-0 storage finds its adoption in enterprise data centers, its reliability becomes a focal point and bottleneck that may determine its wide success and deployment for enterprise applications.

This paper outlines a framework we use to quantify the reliability of the Fusion-io® ioDrives [8] (NAND flash based solid state storage device). This framework allows us to design our products more competitively regarding reliability by trading off various reliability aspects.

The rest of the paper is organized as follows: Section 2 briefly discusses the related works. Section 3 discusses factory bad blocks and infant mortality of NAND flash. Section 4 presents the endurance test results. Based on detailed error measurement, an exponential growth model is proposed in Section 5 to characterize the impact of wear-out on bit error rate. The same type of model is fitted for the impact of read disturb in Section 6. Data retention is the last piece of our framework and detailed test data and tradeoff is dissected in Section 7. We conclude our paper in Section 8.

## 2. Related Works

The reliability of NAND flash devices have been studied at component level extensively [1, 2, 3, 4] and quantified from specific aspects. For instance, previous work has studied endurance regarding how many P/E cycles NAND flash can tolerate before wear-out [1], while other work has measured RBER (Raw Bit Error Rate) [2], quantified program disturb and read disturb impact [2], or characterized retention time [2][3][4]. However, there is a need for more information from testing Solid-State Storage at the system level and from studying the reliability from multiple angles at the same time; Factory bad blocks, infant mortality, endurance, bit error rate, read disturb and retention time are important aspects of reliability and should be quantified together in a single framework, considering a variety of tradeoffs. Mielke et al. [1,2] provide the most comprehensive test results in the literature, in terms of RBER versus P/E counts, retention and read disturb [2] and NAND flash reliability at component level [1].

This paper discloses a comprehensive set of test data with NAND flash parts from a variety of vendors and different NAND technologies (3x nm, 4x nm, and 5x nm). The sample size in our tests is the largest among published so far compared with existing tests from previous researcher and the set of reliability quantification is one of the most comprehensive in the industry. The total capacity tested in our framework exceeds 10 TB, and more than 1000 ioDrives are tracked for factory bad block and infant mortality analysis. About 60,000 erase blocks for each type of flash devices were exercised and worn out during the tests. Based on the test data, exponential growth models are proposed to characterize the impact of P/E cycles and read disturb to Raw Bit Error Rate. These models are used to fine-tune our design of products.

## 3. Reliability In Terms of Factory Bad Blocks and Infant Mortality

We track all the FBB (Factory Bad Blocks) identified by vendors and run MAT (Manufacturing Acceptance Test) for 24~48 hours. The MAT goes through a variety of test cases and runs about 10~100 P/E cycles with all the failures and root causes being logged.

Figure 1 and 2 are a subset of samples from our MAT. There are about 1000 ioDrives tested in the subset, each ioDrive has 200 NAND flash dies on board. So there are about 200,000 NAND flash dies for statistical analysis. There are 8096 erase blocks per die. We count the number of bad blocks per die before and after MAT test.

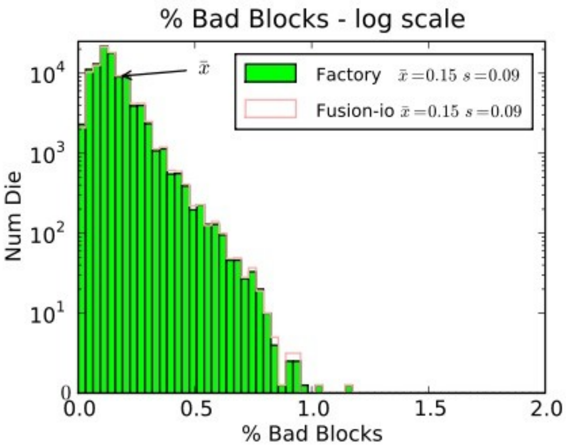

**Figure 1. Bad blocks from vendors and MAT**

In Figure 1, we note:

- A small increase in the number of bad blocks during MAT, which is infant mortality escaped from vendors' MAT but captured by our MAT;
- The number of bad blocks is bell-shape distributed at first glance, however, a close look shows linearity at logarithmic scale when the number of bad blocks is greater than 10;
- There is a heavy tail in the distribution of number of bad blocks with some dies having more than 100 bad blocks; and
- We use x% as our MAT pass/fail criteria: ioDrive with more than x% of PEB (Physical Erase Block) failure will not be shipped to customer. ioDrives failed in MAT will be reworked to replace the problematic dies and the manufacturing lot will be further investigated for quality control. The PEB failure criterion varies by product, in part because different NAND Flash technologies have different specifications for FBB and grown bad blocks over product life. Historically 1% was used for 5x nm for example.

Figure 2 presents a logistic view of the mean and standard deviation of number of factory bad blocks ($\bar{x}$ *and* $s$) from MAT which is a very powerful chart for our quality control. Any spikes on the charts will be investigated and root-cause analyzed.

With MAT, factory bad block and infant mortality analysis, we weed out the weak NAND flash parts received from vendors to guarantee our product's quality and improve customer's satisfaction.

Besides MAT, we have also implemented a process to track time to failure data from field.

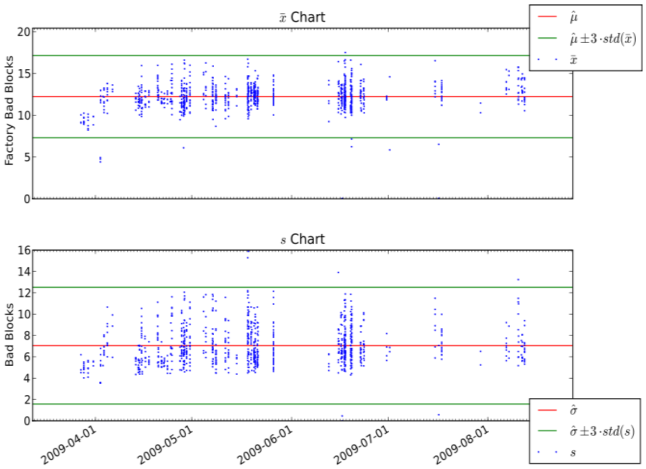

**Figure 2.** $\bar{x}$ *and* $s$ **chart for factory bad blocks**

## 4. Reliability In Terms of Endurance

The repetitive program and erase cycles will induce electrical stress to flash devices and trap the injected charges in the tunnel oxide, which will degrade the gate oxide insulating properties at low field over time. The buildup of the trapped oxide charge leads to threshold voltage variations that can considerably change the programmed and erased voltage level of flash memory cells, and generate Stress Induced Leakage Current (SILC), by means of a Trap Assisted Tunneling (TAT) mechanism [1].

- At certain point, flash memory cell will be worn out and we cannot use it to store data anymore. There are primarily three types of NAND flash failures, i.e., erase failure, program failure and ECC failure.
- As the trapped charges being build up in the tunnel oxide, eventually they cannot be pulled out any more. Flash memory cannot reset to erase state “1”, and will return a “fail to erase” error. We will dissect this type of failure in more detail in this section.
- Flash memory cannot be programmed to charge state “0” and will return a “fail to program” error. We find during our tests that program error is very rare in 5x nm flash device and an early indication of a bad flash die in 3x nm and 4x nm flash devices. We will do further investigation in future work.
- The charge loss or disturb from other cells causes voltage level to shift so severely that there are so many bit errors in an ECC codeword and ECC scheme cannot correct the bit errors. This will be discussed in details in next section.

Different vendors may have different proprietary designs about how to deal with oxide trap, determine and handle erase errors, and extend the life of flash devices in terms of time-to-failure, however, there is a hard wall determined by physical limit.

Figure 3 is the Weibull plot of ioDrive with 5x nm MLC (Multi-Level Cell) flash from vendor A. Erase operations generate “fail to erase” errors at 23k ~ 32k P/E cycles.

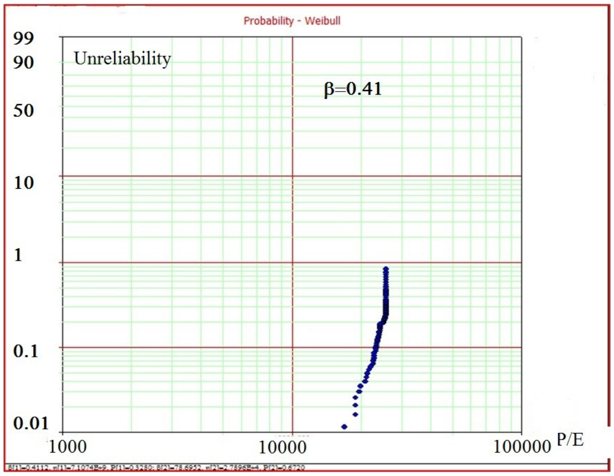

**Figure 3. Time to erase failure for 5x nm MLC**

When there is a failed erase block, we retire the block and remap the data to a healthy block. If the number of failed blocks exceeds a threshold, we trigger an algorithm called Flashback$^{TM}$ to retire the whole die and use a redundant die to replace it. Flashback$^{TM}$ Technology eliminates Single-Points-of-Failure (SPoF) in the Flash array internal to Fusion-io$^{®}$ products. The remapping/retirement algorithm and Flashback$^{TM}$ in our product can significantly improve reliability.

## 5. Reliability In Terms of Raw Bit Error Rate

Compared to SLC (Single-Level Cell), MLC technology scales up the capacity of flash devices, however, induces more bit errors at the same time. At microscopic level, adjacent voltage levels differ only by hundreds of electrons. As we advance from 5x to 2x nm technology, it becomes more and more challenging to maintain the guard band between the adjacent voltage levels, especially when a flash device ages.

Figure 4 illustrates the RBER versus P/E counts for 5x/4x/3x nm products. NAND flash parts of 3x and 4x nm are from Vendor B and C respectively. The RBER reading was a function of test conditions we imposed that may not be reflective of end-user experience.

High bit error rates can be handled by stronger ECC. Fusion-io’s products deploy very strong ECC scheme to achieve $10^{-20}$ UBER (Uncorrectable Bit Error Rate) target. Besides this, we add another level of parity protection, i.e., block retirement/remapping and Flashback$^{TM}$, as discussed in Section 4.

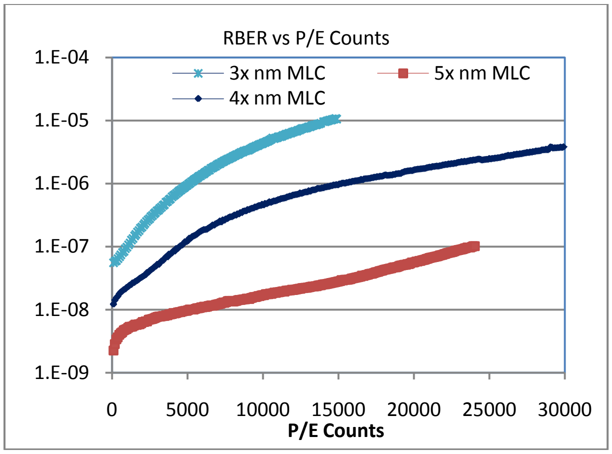

**Figure 4. Raw bit error rate versus P/E counts for different type of flash devices**

We curve-fit the RBER versus P/E counts and find it can be approximated by an exponential growth model very precisely, especially when the P/E count is high (Figures 5, 6, and 7):

$$RBER = Ae^{Bx} + C, \qquad (x\ is\ P/E\ Count)$$

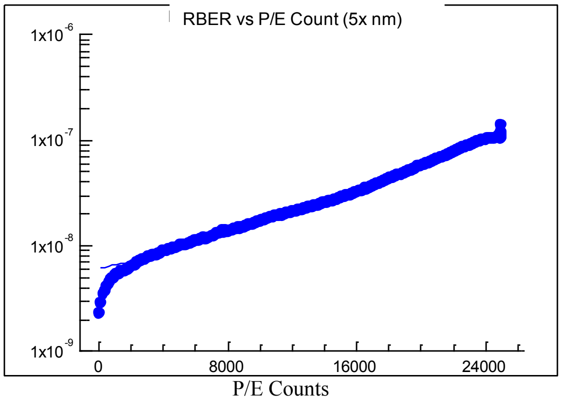

**Figure 5. Exponential growth model for 5x nm MLC (raw bit error rate versus P/E counts)**

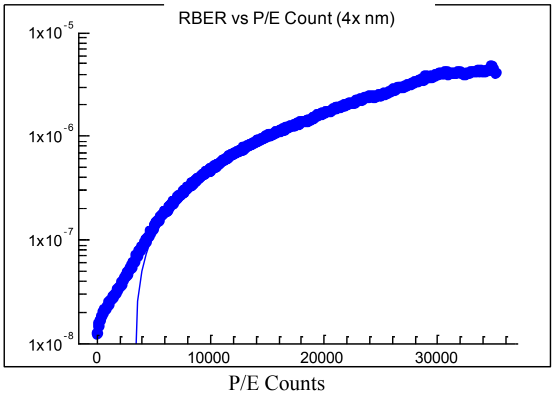

**Figure 6. Exponential growth model for 4x nm MLC (raw bit error rate versus P/E counts)**

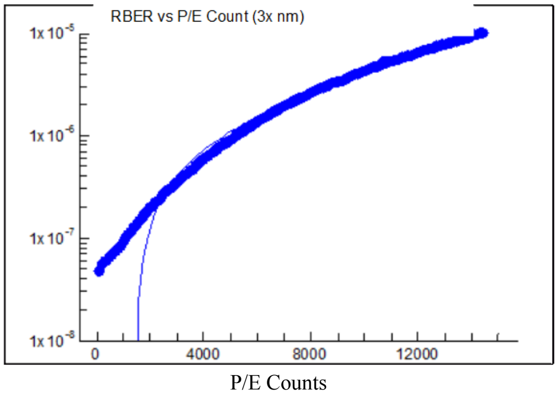

**Figure 7. Exponential growth model for 3x nm MLC (raw bit error rate versus P/E counts)**

The exponential growth model can be explained by the existing discoveries in [5, 6, 7] where the increase of threshold voltage of a memory cell due to charge trapping versus the number of stress events (program or erase cycles) was found to follow power-law. Table 1 lists the parameters for the fitting.

During the endurance test, we not only collect the RBER statistics, but also log how many bit errors occur in one ECC codeword. Figure 8 illustrates the multi-bit error rate (>1 bit error in one ECC codeword) for 5 ioDrives (3x nm) we tested. Although on the RBER chart, these five boards show very smooth and close curves, their multi-bit error characteristics are dramatically different. Some boards show a rate two orders of magnitude higher and drop back to normal after we retire the problematic blocks.

**Table 1. Parameters for the exponential growth models (raw bit error rate versus P/E counts)**

| 3x nm MLC | Value | Std Err | 95% Confidence |
|---|---|---|---|
| A | 1.1831E-06 | 5.37E-08 | 1.08E-06 |
| B | 0.0001543 | 2.85E-06 | 0.000149 |
| C | -1.4696E-06 | 8.01E-08 | -1.6E-06 |

| 4x nm MLC | Value | Std Err | 95% Confidence |
|---|---|---|---|
| A | 1.3631E-06 | 8.6E-08 | 1.19E-06 |
| B | 4.6896E-05 | 1.58E-06 | 4.38E-05 |
| C | -1.4805E-06 | 1.01E-07 | -1.7E-06 |

| 5x nm MLC | Value | Std Err | 95% Confidence |
|---|---|---|---|
| A | 2.6953E-09 | 1.97E-10 | 2.31E-09 |
| B | 0.0001608 | 4.12E-06 | 0.000153 |
| C | 5.4685E-09 | 7.77E-10 | 3.94E-09 |

In our block retirement/remapping algorithm, any block with “fail to erase” or “fail to program” errors will be retired and remapped. Additionally, when we see that the number of bit errors in an ECC codeword exceeds certain thresholds, we selectively retire the blocks contributing to this type of events.

Erase failure and bit error are two primary competing failure mechanisms. We see ECC failures more often than erase failures in 3x nm and 4x nm while in 5x nm the dominant failure is erase failure.

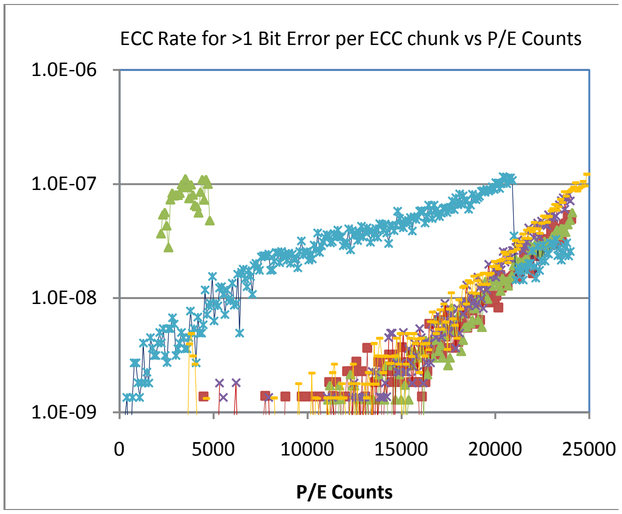

**Figure 8. Multi-bit error rate versus P/E counts**

## 6. Reliability In Terms of Read Disturb

Due to the layout of flash cells, some cells not being read will receive elevated voltage stress. Disturb occurs when charge collects on the floating gate, causing the cell to appear weakly programmed.

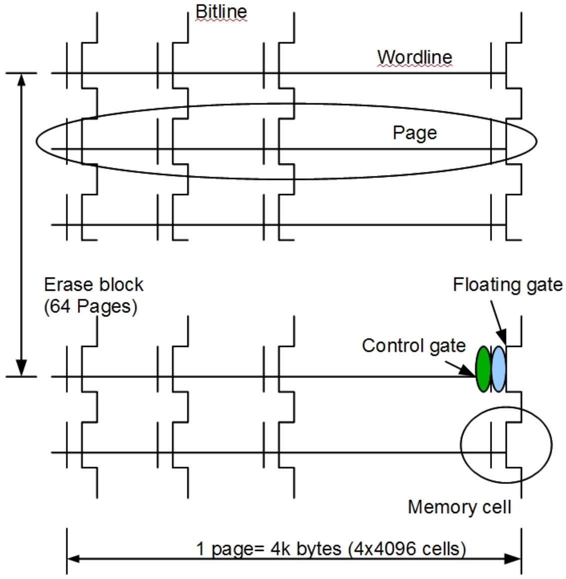

**Figure 9. NAND flash layout and read disturb**

Figure 9 illustrates cells in a NAND flash device. Each bit cell consists of a MOSFET with a floating gate. To find out if there are any electrons trapped on a particular floating gate, the memory device must read out as an entire word. It does this by putting a read bias on the gates of all the bit cells in a word via the word line. However, because of the organization of the NAND flash pages, the memory must also put some pass voltage on the gates of all the other cells in the block in each bit line. The pass voltage on the gates attracts electrons to the floating gates, which is similar to soft programming. Eventually, some of those electrons will move into the bit cell's floating gate over time. When enough electrons make the jump, the cumulative charge will be sensed as higher level than what was originally written. This is data corruption and the essence of the read-disturb problem.

Figure 10, 11, and 12 present the read disturb results in terms of RBER versus P/E counts and read counts, for 5x, 4x, and 3x nm flash respectively. The impact of read disturb varies with not only different technologies but also the P/E count. Read disturb gets more severe with P/E counts, and more significant as we move from 5x to 3x nm.

Again we find that read disturb impact can be curve-fitted with an exponential growth model as well, although different parameters need to be chosen for different technologies and at different ages in terms of P/E counts. Figures 13 and 14 show the exponential growth model curves for 5x nm at 5k and 15k P/E cycles respectively. We do not present curve-fitting results for 4x and 3x nm due to space limit of the paper.

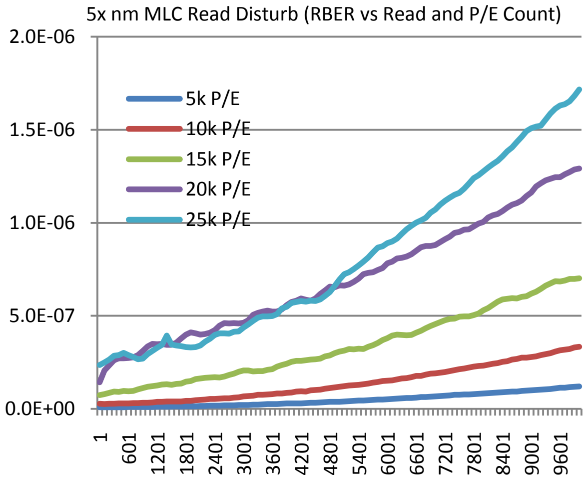

**Figure10. RBER versus P/E counts and read counts for 5x nm MLC flash**

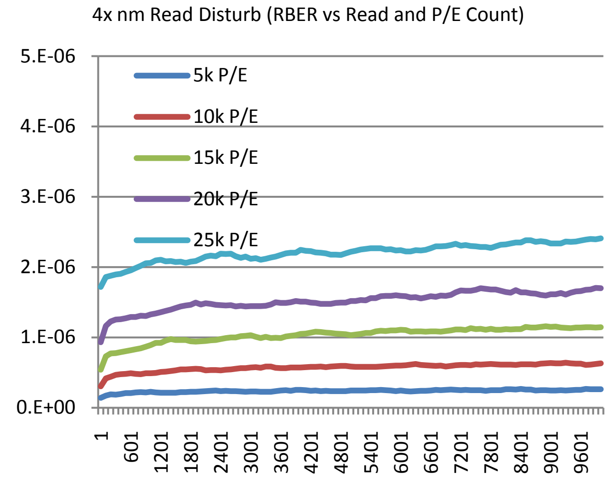

**Figure 11. RBER versus P/E Counts and read counts for 4x nm MLC flash**

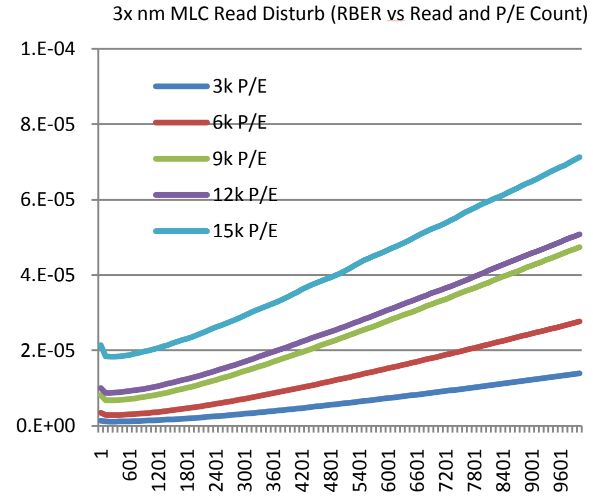

**Figure 12. RBER versus P/E Counts and read counts for 3x nm MLC flash**

In order to mitigate the impact of read disturb, we periodically scrub each page in the NAND Flash memory. When the read count exceeds a threshold, we refresh the block. The threshold is chosen based on the fitting models developed in this paper.

| | Value | Std Err | 95% Confidence |
|---|---|---|---|
| A | 4.56E-07 | 3.25E-08 | 3.91E-07 |
| B | 0.000148 | 5.58E-06 | 0.000137 |
| C | -2.4E-07 | 3.89E-08 | -3.2E-07 |

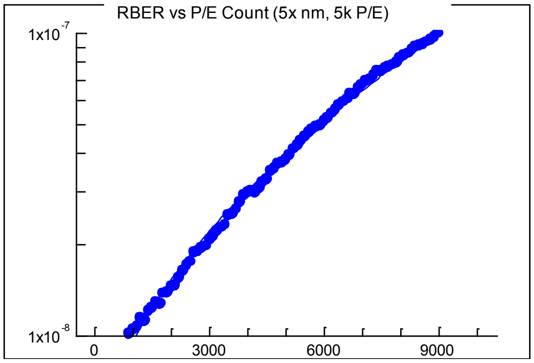

**Figure 13. Read disturb exponential growth model for 5x nm MLC flash at 5k P/E**

| | Value | Std Err | 95% Confidence |
|---|---|---|---|
| A | 2.62E-08 | 1.23E-09 | 2.38E-08 |
| B | 0.00017 | 3.9E-06 | 0.000162 |
| C | -2.1E-08 | 1.56E-09 | -2.5E-08 |

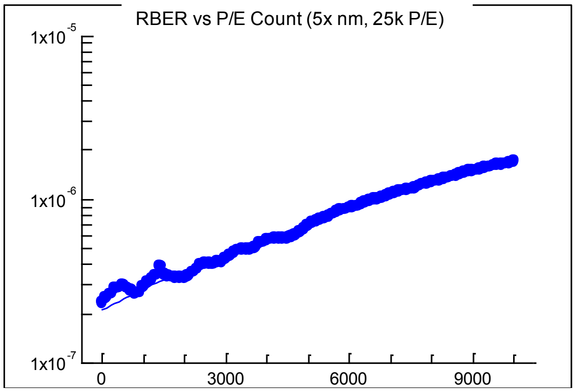

**Figure 14. Read disturb exponential growth model for 5x nm MLC flash at 25k P/E**

## 7. Reliability In Terms of Retention

Charge loss/gain occurring on the floating gate over time will lead to bit flipping. As a flash memory cell being repeatedly programmed and erased, the tunnel oxide layer becomes weak which leads to an increase in the SILC of the memory cell, thus affecting data retention.

The typical data retention time for flash memory is specified as 10-20 years when a device is new. Data Retention in Solid-State Storage has two aspects.

1. Power-on data retention. This is concerned with behavior similar to that known in DRAM for many years, albeit on a different time scale. In other words, an active solid-state storage system must periodically refresh its data. This is done at cost, both in terms of performance load due to related background tasks, and life, as each refresh consumes a quantum of life in the media.
2. Power-off data retention. This pertains to the same data "loss" phenomena as power-on data retention, without the benefit of refreshing. There are a few scenarios to consider:
   a. The "Hurricane Katrina" scenario, in which power is lost in a data center for an extended period of time due to natural disaster.
   b. Scheduled downtime: Servers may be re-located, or remotely provisioned. The expectation is that data or applications residing on Solid-State Storage will persist in the face of related downtime. What duration of downtime is to be expected is a very valid question.
   c. "72 Hours": Traditional HDD-based storage systems employed lead-acid batteries with enough charge to hold up RAID Cache systems for at least 72 hours. More modern storage systems have evolved to super-cap technology instead of batteries, and have employed cache de-staging to alternate NVRAM, such as Flash drives, possibly with enough reserve to hold up cache to "ride-out" short duration power interruptions.

Data retention requirements for Solid State Storage are therefore different from tape drives. For instance, tape drives are used primarily for archives and a 10-20 year retention time is required, while in Solid State Storage, we typically target 3 months of data retention time based on the use and outage scenarios for solid-state in Enterprise applications.

Data retention is heavily dependent on the applied ECC methods, RBER, flash age in terms of P/E cycles and read counts. So when we specify data retention, we also need to specify the context:

- What UBER we are targeting;
- What the ECC scheme capability is;
- How many P/E cycles we want to specify for the devices; and
- How many read counts we will tolerate before we refresh data.

Tables 2, 3, and 4 list the RBER results for 5x, 4x, and 3x nm MLC ioDrives at different P/E cycles. After being worn out to different P/E levels, the boards were put into an oven for accelerated temperature testing. It is generally assumed [1] that the relationship between time and temperature for the failure mechanisms associated with retention (and indeed most MOS failure mechanisms) may be modeled by the Arrhenius Equation. An activation energy of 1.1 eV was used to calculate the Arrhenius acceleration factor.

Fusion-io's target UBER is $10^{-20}$. For example, an RBER of $6x10^{-5}$ translates to a UBER of $10^{-20}$ when 11-bit/240-byte ECC is used, assuming bit-error independence, i.e., binomial relationship between RBER and UBER. Using this scheme, we can achieve 3 months of retention time at 25000 P/E cycles without compromising the $10^{-20}$ UBER target. Moving to 3x nm, we have to strengthen the ECC scheme by increasing the number of corrected bits in a larger codeword, if we are to safely achieve 3 months of retention time at 15000 P/E cycles.

Tables 2, 3, and 4 show the tradeoff between P/E cycling and retention time: 12000 P/E cycles (2 month retention) and 15000 P/E cycles (1 month retention) have similar RBER. If we can only correct RBER at about 5.8E-5, we have to choose between higher P/E cycles (15000) but shorter retention time (1 month) or lower P/E cycles (12000) but longer retention time (2 months).

**Table 2. Data retention for 5x nm MLC (RBER)**

| P/E | 0 month | 1 month | 2 month | 3 month |
|---|---|---|---|---|
| 5000 | 8.035E-09 | 4.512E-08 | 6.415E-08 | 7.67E-08 |
| 10000 | 1.621E-08 | 7.366E-08 | 1.280E-07 | 1.779E-07 |
| 15000 | 4.789E-08 | 1.479E-07 | 2.989E-07 | 4.891E-07 |
| 20000 | 1.122E-07 | 2.402E-07 | 5.320E-07 | 9.76E-07 |
| 25000 | 2.759E-07 | 5.008E-07 | 1.484E-06 | 3.139E-06 |

**Table 3. Data retention for 4x nm MLC (RBER)**

| P/E | 0 month | 1 month | 2 month | 3 month |
|---|---|---|---|---|
| 5000 | 1.513E-07 | 4.664E-07 | 1.155E-06 | 2.281E-06 |
| 10000 | 5.407E-07 | 1.426E-06 | 3.639E-06 | 7.62E-06 |
| 15000 | 1.167E-06 | 3.66E-06 | 1.021E-05 | 2.128E-05 |
| 20000 | 1.786E-06 | 9.565E-06 | 2.984E-05 | 6.028E-05 |

**Table 4. Data retention for 3x nm MLC (RBER)**

| P/E | 0 month | 1 month | 2 month | 3 month |
|---|---|---|---|---|
| 3000 | 3.217E-07 | 4.077E-06 | 5.744E-06 | 7.268E-06 |
| 6000 | 9.555E-07 | 1.009E-05 | 1.571E-05 | 2.106E-05 |
| 9000 | 2.427E-06 | 2.169E-05 | 3.483E-05 | 4.78E-05 |
| 12000 | 4.386E-06 | 3.369E-05 | 5.676E-05 | 7.935E-05 |
| 15000 | 7.958E-06 | 5.88E-05 | 1.044E-04 | 1.506E-04 |

## 8. Conclusions

A framework was developed to quantify reliability of ioDrives, PCI-e based Solid-State Storage products from Fusion-io®. A comprehensive set of test data for a variety of vendors, different NAND technologies (3x, 4x, and 5x nm) were analyzed in-depth. The total capacity tested in our framework exceeds 10 TB and about 60,000 erase blocks for each type of flash devices were exercised and worn out during the tests. The key point from our analysis is that reliability and data retention of Solid-State Storage have multiple facets which should be traded off carefully. Some key innovations from Fusion-io®, e.g., Flashback™, online block retirement/replacement, strong ECC, scrubbing, etc., improve reliability significantly and were briefly discussed. Exponential growth models were proposed to characterize the impact of P/E cycles and read disturb to Raw Bit Error Rate, used to guide our engineering designs. Future work should consider the implications of parameters in the exponential growth models for different NAND flash technologies, statistical analysis to program failures, and error bias between 0 to 1 and 1 to 0 bit-flips.

## References

[1] Jian Justin Chen et al.., *Flash Memory Reliability, Nonvolatile Memory Technologies with Emphasis on Flash: A Comprehensive Guide to Understanding and Using Flash Memory Devices*, Wiley Interscience, 2010.

[2] Neal Mielke, et al., “Bit Error Rate in NAND Flash Memories”, *IEEE CFP08RPS-CDR 46th Annual International Reliability Physics Symposium*, Phoenix, 2008, pp.9-19.

[3] Nelson Duann, “Error Correcting Techniques for Future NAND Flash Memory in SSD Applications”, *Flash Memory Summit 2009 Proceedings*, http://www.flashmemorysummit.com/English/Conference/Proceedings_Chrono.html

[4] Eitan Yaakobi, “Error Correction Coding for Flash Memories”, *Flash Memory Summit 2009 Proceedings*, http://www.flashmemorysummit.com/English/Conference/Proceedings_Chrono.html

[5] H. Yang and et al., “Reliability Issues and Models of Sub-90nm NAND Flash Memory Cells”, *International Conference on Solid-State and Integrated Circuit Technology*, 2006, pp.760-762.

[6] Vidyabhushan Mohan, Modeling The Physical Characteristics Of NAND Flash Memory, University of Virginia, Master Thesis, May 2010.

[7] R. Yamada and et al., “A Novel Analysis Method of Threshold Voltage Shift Due to Detrap in a Multi-level Flash Memory”, In *Symposium on VLSI Technology*, Digest of Technical Papers, 2001.

[8] http://www.fusionio.com/products/iodrive/