



NEPP Non-Volatile Memory FY09 Summary Report

Douglas Sheldon
Jason Heidecker

Jet Propulsion Laboratory
Pasadena, California

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

JPL Publication 10-2 01/10



NEPP Non-Volatile Memory FY09 Summary Report

NASA Electronic Parts and Packaging (NEPP) Program
Office of Safety and Mission Assurance

Douglas Sheldon
Jason Heidecker

Jet Propulsion Laboratory
Pasadena, California

NASA WBS: 724297.40.43
JPL Project Number: 103982
Task Number: 03.02.03

Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

<http://nepp.nasa.gov>

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the National Aeronautics and Space Administration Electronic Parts and Packaging (NEPP) Program.

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

Copyright 2010. California Institute of Technology. Government sponsorship acknowledged.

TABLE OF CONTENTS

1	Overview.....	1
2	NAND Flash Test Setup and Software Development.....	2
3	Reliability Characterization of 8 Gb NAND Flash Devices.....	9
3.1	Bad Block Analysis.....	10
3.2	Endurance Testing	14
4	Conclusions.....	15
	References.....	16
	Appendix A.....	17

1 OVERVIEW

This report documents the activities and results of the fiscal year 2009 (FY09) funding for the NASA Electronic Parts and Packaging (NEPP) program for non-volatile memory (NVM) devices.

The FY09 task was divided into two main efforts:

1. Development of in-house measurement capability for flash devices
2. Reliability characterization of multi-level vs. single-level flash devices

The *FY09 NEPP Non Volatile Memory* study was organized into these two sections in recognition of the continued and increasing importance of NVM to NASA. All NASA missions use NVM for boot code storage and some limited data archiving. Space-grade NVM has lagged commercial NVM development by many orders of magnitude, however. This gap in density only continues to increase as space-grade memories remain constant in the 1–16 Mbit levels, while commercial flash-based devices shrink in size and double in capacity every 18 months. Commercial flash devices are now at the 32 Gbit level, 10,000 times denser than current NASA NVM.

This FY marks the beginning of a significant technological transition in NVM for NASA. The aggressive scaling of commercial NVMs has, as a by-product, made some of the flash devices radiation-tolerant enough to be at least considered for low-level radiation environment missions [Irom 2008]. This NEPP task is focused on understanding the reliability and potential future applications of such highly scaled flash memories.

2 NAND FLASH TEST SETUP AND SOFTWARE DEVELOPMENT

In recent years [Sheldon 2007 and 2008, Chen 2009], NEPP-supported NVM testing was performed at outside test facilities. While use of outside test capabilities provides NEPP with a useable end product, the detailed level of understanding required for NASA mission assurance and risk mitigation activities fundamentally requires in-house device testing and characterization capability. Most commercial test houses are designed for pass/fail testing, not the in-depth levels of characterization needed to ensure NVM success in NASA missions.

The focus for this FY's test development efforts was NAND flash devices. The other common type of flash device is the NOR flash. The difference between the NOR and NAND flash can be seen in Figure 1. The NAND flash has a much smaller cell size compared to the NOR flash.

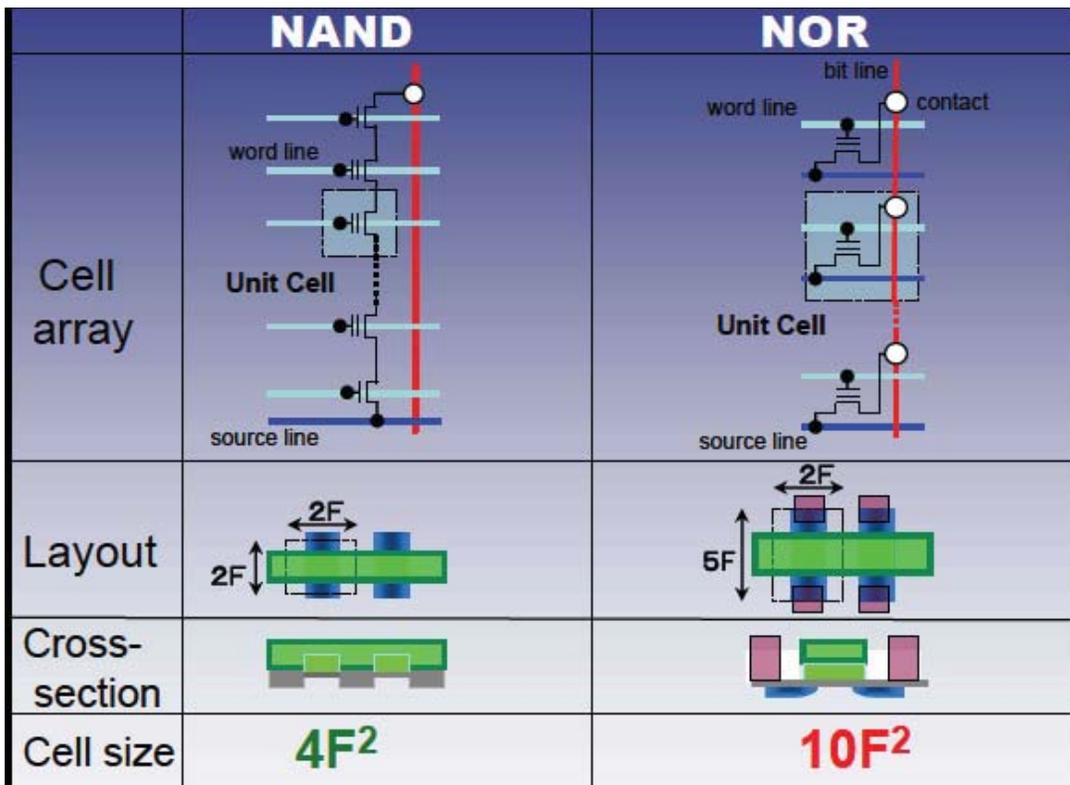


Figure 1. Differences between NAND and NOR flash [Cooke 2008]

Because of its much smaller cell size, NAND devices will always be able to provide more bits per device for a given technology level than NOR devices. Figure 2 shows an example of this (NAND and NOR devices of the same 90 nm technology). It takes two NOR devices to provide the same level of storage as one NAND device. Figure 2 also shows the relative increase in size of the NOR device over the NAND device.

Table 1 shows a comparison of NAND versus NOR flash in terms of advantages and disadvantages.

NAND is very similar to a disk drive in that it is sector-based. In NAND flash terms, the sectors are called pages. This is why NAND flash is well suited for storage of sequential data such as pictures, audio, and data files. Like a disk drive, NAND is not well suited for random access, such as executing code, although random access can be accomplished at the system level by shadowing the data to RAM (similar to what a PC does with BIOS). Like a disk drive, NAND devices have bad sectors (pages) or blocks, which require software management. Also like a disk drive, NAND requires error correction code (ECC) to maintain a specified error rate for given system performance. Unlike a disk drive, it is possible to wear out the NAND cell. Therefore, NAND devices require the use of wear-leveling techniques.

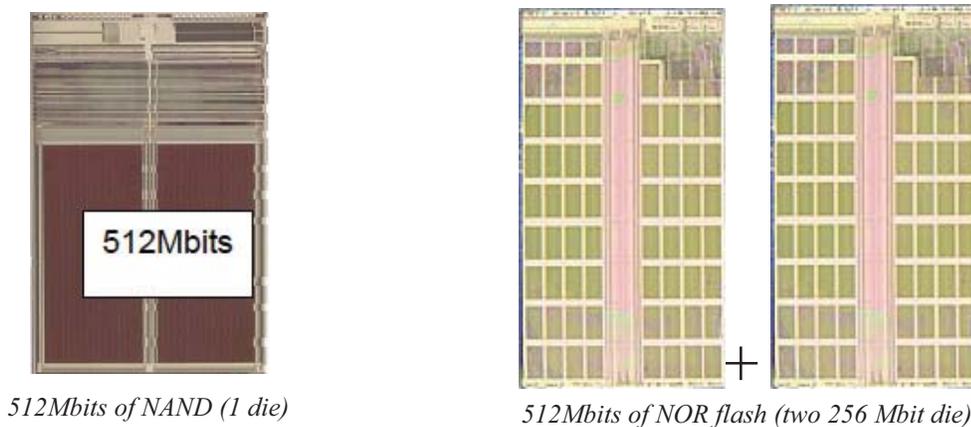


Figure 2. NAND and NOR flash devices in 90nm CMOS technology

Table 1. Comparison of NAND and NOR flash

Device	Advantages	Disadvantages	Applications
NAND	Fast write and erases	Slow random access and difficult byte writes	File applications and large sequential data like video, voice, etc.
NOR	Random access with byte writes	Slow writes and erases	Execute directly from memory like EPROM

NAND flash data are grouped into blocks, which are the smallest erasable entity. Erasing a block sets all bits to “1” or bytes to FFh. The programming operation changes erased bits from “1” to “0.” This can be seen graphically in Figure 3.

Figure 3 shows that Program and Read operations are page-based while Erase operations are block-based. Flash devices get their name from the ability to erase large amounts of data at once (by block). This amount of data is said to be erased in a “flash.”

The page and block architecture of a NAND flash is shown in Figure 4. Figure 4 also provides the representative times it takes to accomplish these steps. Figure 4 shows that a NAND page is made up of 2,112 bytes while a NAND block has 64 pages. The time to erase one block (64 pages) is approximately 2 msec. The time to program a single page is 300 msec. Programming one page takes over 100 times the amount of time it takes to erase 64 pages. Once a page has been loaded into the register, the read times are quite fast (at approximately 30 nsec).

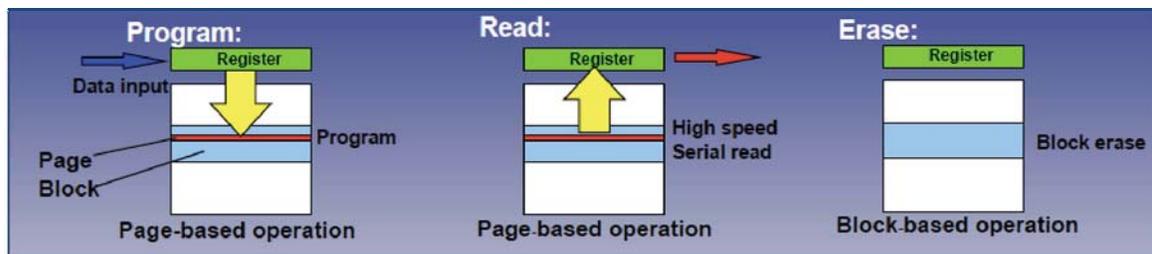


Figure 3. Program, Read, and Erase operations on a NAND flash [Cooke 2008]

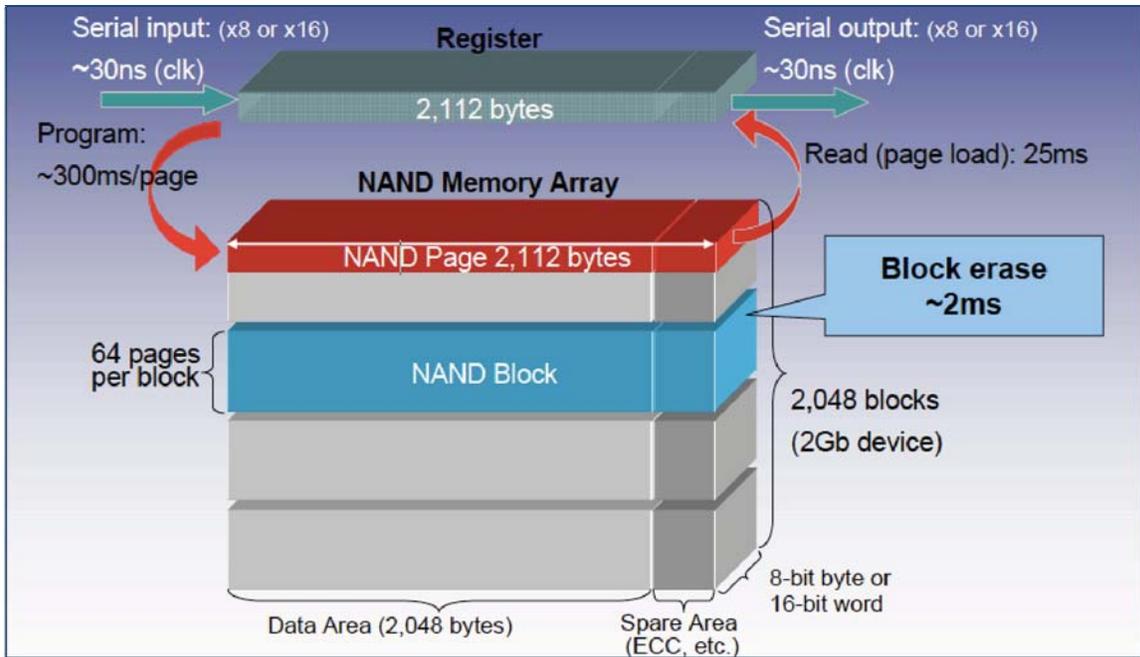


Figure 4. NAND flash memory diagram

The NAND device has the following pins:

- CE#—Chip Enable
- WE#—Write Enable
- RE#—Read Enable
- CLE—Command Latch Enable
- ALE—Address Latch Enable
- WP#—Write Protect
- R/B#—Ready/Busy
- I/O 0-7—Data Bus (for a 8 bit wide device)
- Vss and Vcc—Ground and Power

This type of indirect addressing enables no changing of testing pinouts as densities change, for example. Indirect addressing also means that programming, erasing, and reading commands are actually sent into the device in a series of command cycles. The input / output (I/O) pins are used to send in specific command information. This means operations of a NAND flash device can be driven by combinations of four fundamental commands: CMD (command), ADDR (address), READ, and WRITE. Figure 5 shows a timing diagram for a generic single CMD sequence.

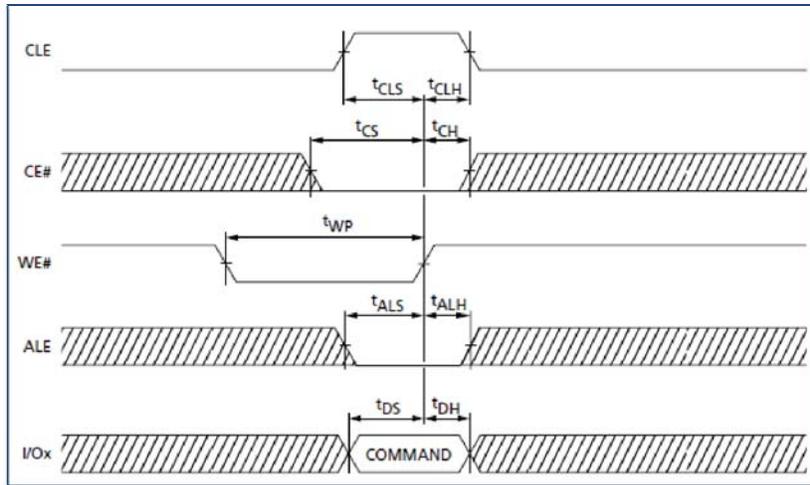


Figure 5. Individual CMD sequence with indirect NAND addressing scheme

This study used the JDI Instruments ATV system as the testing platform. The JDI ATV is a custom hardware and software tester designed for portability and ease of use. The JDI ATV features:

- 32 digital I/Os
- 50 MHz
- 24 digital out-only
- 0 to 5.5 Volt outputs
- 7 ns minimum waveform
- Graphical Waveform Editor

Figure 6 shows the JDI ATV tester.

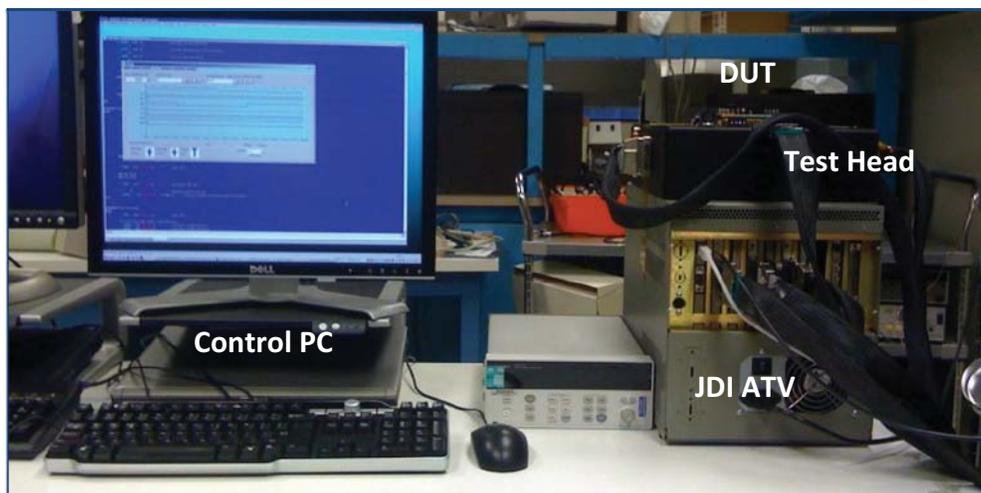


Figure 6. JDI ATV tester

The JDI ATV provides a dynamic test environment (DTE) to both develop test software as well as integrate operational commands. This environment is shown in Figure 7.

With the JDI ATV test environment, the timing diagram waveform shown in Figure 5 can then be easily implemented into the JDI ATV hardware. The JDI ATV equivalent of CMD command sequence in Figure 5 is shown in Figure 8.

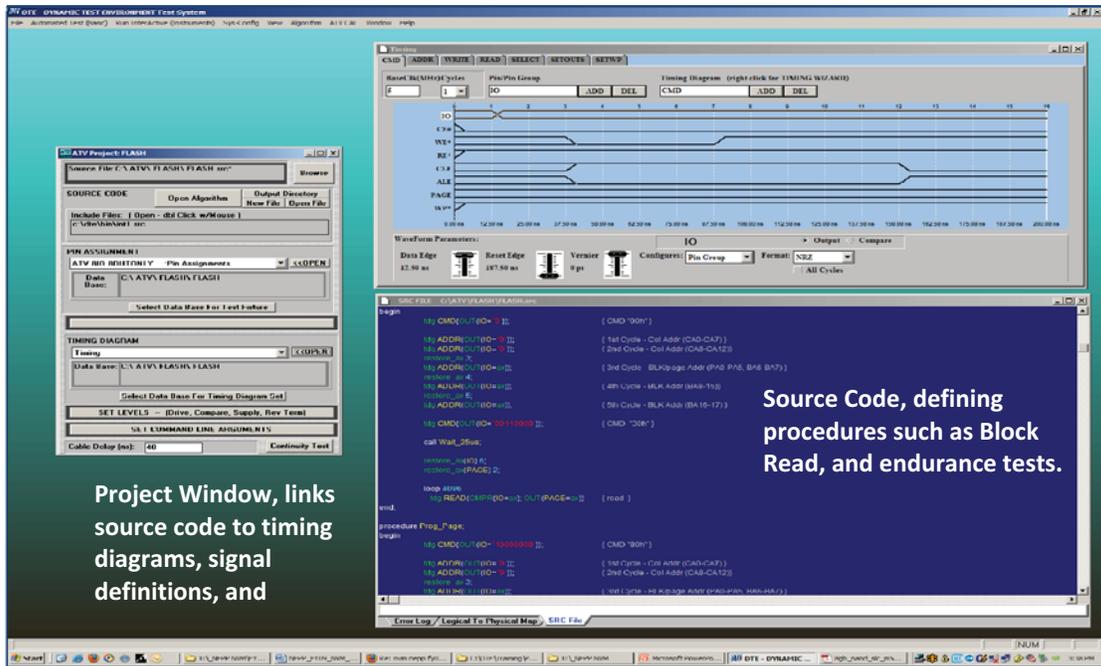


Figure 7. JDI ATV DTE screen capture

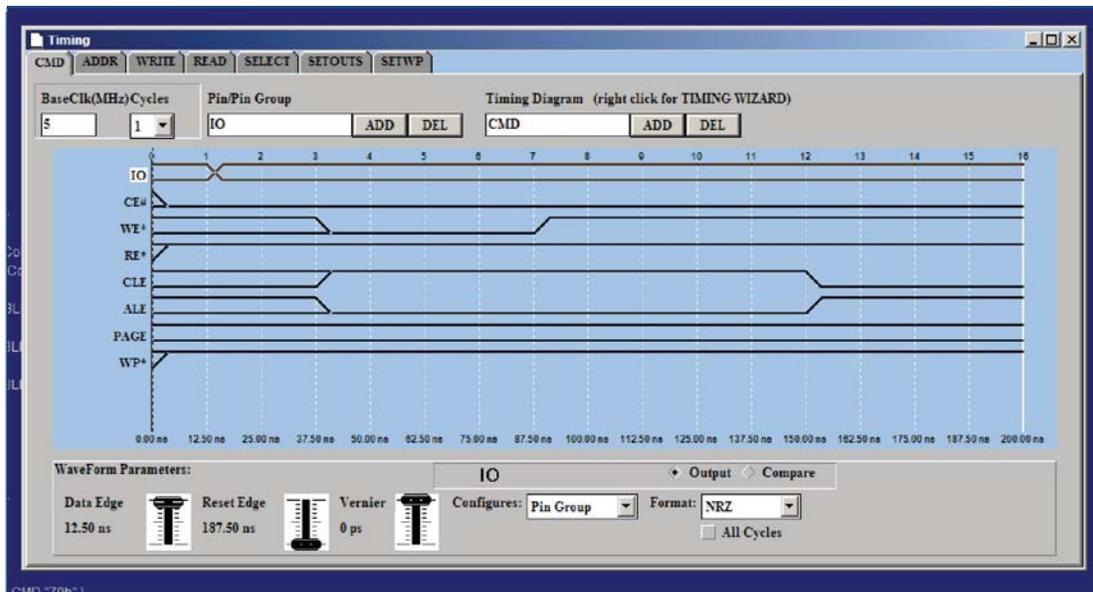


Figure 8. CMD latch cycle (JDI ATV timing diagram)

CMD, ADDR, READ, and WRITE are then called upon programmatically in various ways to produce the standard NAND flash communication procedures such as PAGE READ, PAGE PROGRAM, BLOCK READ, BLOCK PROGRAM, and BLOCK ERASE. Figure 9 shows the timing diagram for the Page Read operation.

Having developed this testing capability with the JDI ATV system, reliability issues were then explored on multi-level and single level NAND flash devices.

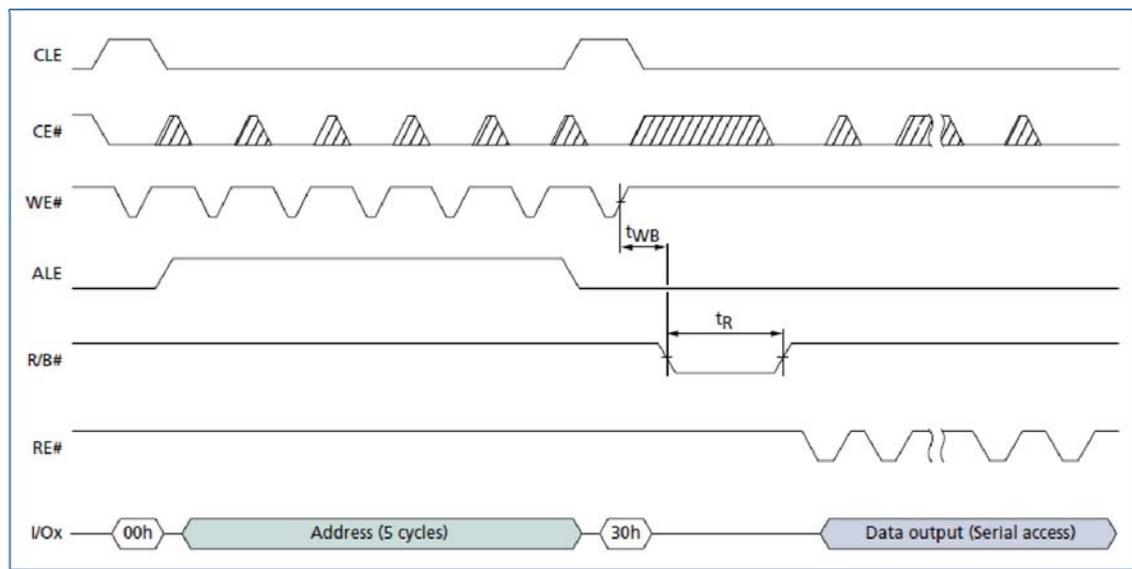


Figure 9. Page Read Operation includes CMD 00h, five ADDR commands, followed by 30h command, followed by repeated READ commands.

3 RELIABILITY CHARACTERIZATION OF 8 GB NAND FLASH DEVICES

NAND flash reliability concerns are data retention, cell endurance, and disturb-related failures [Sheldon 2006]. For this FY, cell endurance was examined for single-level and multi-level 8 Gb NAND devices.

A single-level cell (SLC) stores two states per memory cell and allows one bit programmed / read per memory cell. This is represented in Figure 10.

Figure 10 shows that two normal distributions of threshold voltages (V_t) exist in a SLC flash device. The sense amp uses the V_t information to determine the state of the cell. Programming and erasing the cell moves the reference point between the two different distributions. A multi-level cell (MLC) NAND stores four states per memory cell and allows two bits programmed/read per memory cell. This is shown in Figure 11.

Figure 11 shows how the V_t distributions are smaller but more numerous. The four different distributions enable two separate bits of information to be stored in the same cell. Table 2 is a comparison of the features of SLC and MLC NAND flash.

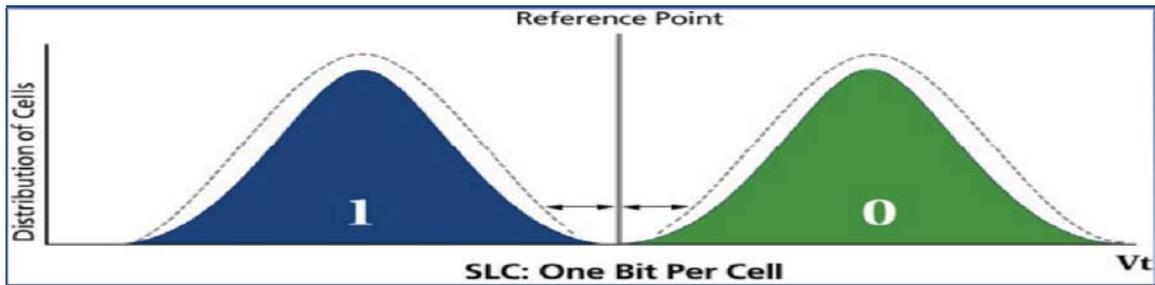


Figure 10. Threshold voltage (V_t) distribution for SLC flash memory

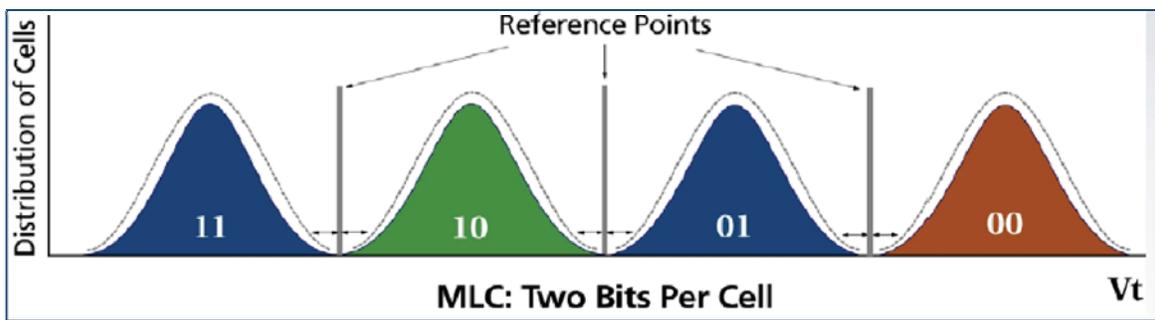


Figure 11. Threshold voltage (V_t) distributions for MLC flash memory

Table 2. Comparison of SLC and MLC NAND flash devices

SLC	Features	MLC
3.3 V, 1.8 V	Voltage	3.3 V
×8, ×16	Data width (bits)	×8
1 or 2	Number of planes	2
2 KB or 4 KB	Page size	2 KB or 4 KB
64	Pages per block	128
4	NOP partial page programming	1
1	ECC per 528 bytes	4+
100 K	Endurance cycles	10 K
25 μsec	t Read	50 μsec
200–300 μsec	t Program	600–900 μsec
1.5–2 msec	t Block erase time	3 msec

The MLC devices are much slower to read and program when compared to the SLC devices. Block erase time for the MLC devices is also as much as two times that of SLC devices. MLC devices also require substantially more ECC.

3.1 Bad Block Analysis

All NAND flash devices are manufactured with a number of bad blocks. The NAND flash cell is designed to have a reduction in ground wires and bit lines to allow a denser layout and greater storage capacity per chip. Manufacturers try to maximize the amount of non-faulty storage by shrinking the size of the transistor below the size where they can be made reliably, to the size where further reductions would increase the number of faults faster than it would increase the total storage available. These bad blocks are identified by the manufacturers before shipment to the customers. Management of bad blocks is a critical requirement for any implementation of NAND flash memory.

How the bad blocks are marked varies across parts and manufacturers. For example, in very large page (4224-byte page) SLC devices, bad blocks are marked by not having FFh in the first and sixth spare bytes of the first page. For MLC devices, any block, where the first byte in the spare area of the last page does not contain FFh, is a bad block. For the Samsung K9F8G08UOM SLC device used in this effort, the first ‘spare’ byte of the first page of each bad block is filled with zeros to mark it as a bad block. By reading all the first spare locations of the first page of each block, a bad block table is recorded.

Bad block information was measured on 10 Samsung 8Gb NAND SLC devices. The results are shown in Figure 12. The x-axis in Figure 12 is block address. The y-axis in Figure 12 is the number of times a bad block was found in that address. For example, two different devices had a bad block at logic block address 500h

The information in Figure 12 shows that more bad blocks are found at higher logical address values than lower logical address values. This means that bad blocks are not randomly distributed. At this time, we do not have the ability to relate logic addresses to physical addresses on the die, so it is uncertain if there is a special layout dependence for these bad blocks. A long-term concern for NASA missions is that this dependency on location may translate into a preferred likelihood for additional bit failures to occur. If such a pre-disposed failure location exists, then mission critical information can be mapped around such sites, for example.

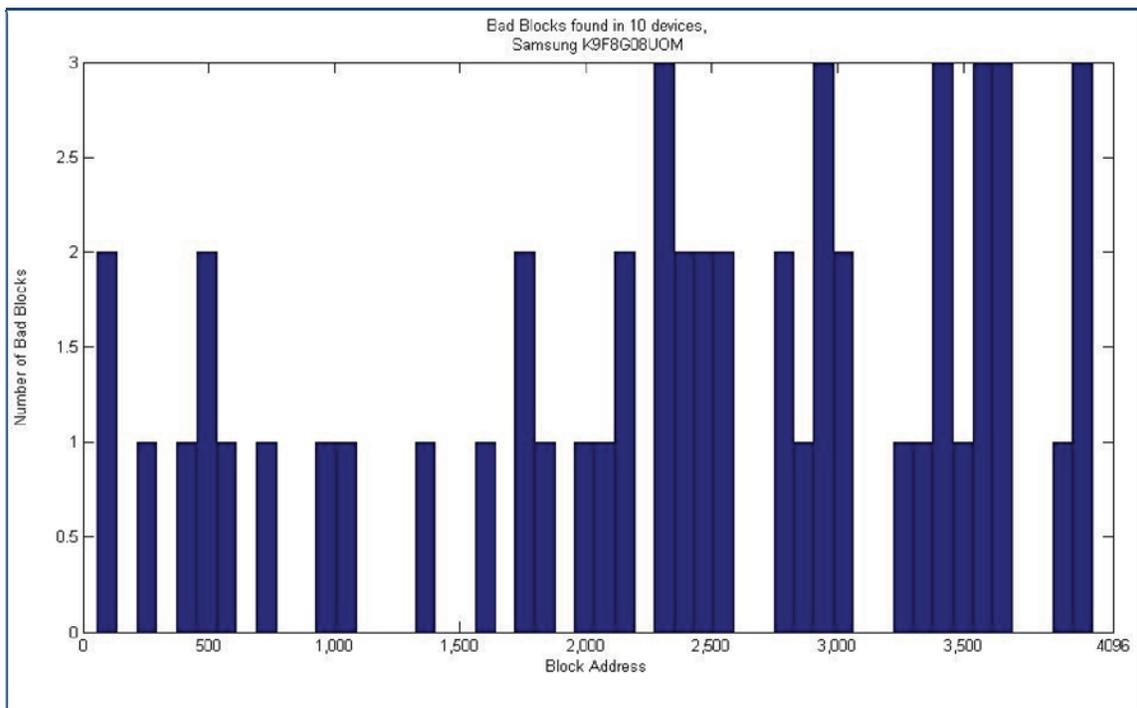


Figure 12. Number of bad blocks versus block address for 10 Samsung K9F8G08UOM devices

Due to the potential significance of this result, additional statistical analysis was performed on the underlying data shown in Figure 12. Both quantitative and graphical tests for statistical normality were performed on the data. A Kolmogorov-Smirnov test of the data showed that the null hypothesis could in fact be rejected at a 95% confidence level. This is a strong indicator that the data are in fact not described by a normal distribution, adding validity to the claim of a non-random distribution of bad blocks. A graphical analysis using a normal probability plot showed that 25% of the values were off the expected normal line, a further re-enforcement of this position.

However, when the Lilliefors test was applied to the distribution, the null hypothesis could not be rejected at a 95% confidence level. It could be rejected at a 70% confidence level however. The differences between the Lilliefors and Kolmogorov-Smirnov tests have to do with how the mean and standard deviation of the reference distribution are calculated. Because two out of the three normality tests verified the initial conclusion, the result appears significant and needs to be further researched and explored. The statistical analysis details are provided in Appendix A.

Usually flash manufacturers define bad blocks as those having reliability that is not guaranteed. However, these bad blocks will often have the same AC and DC performance characteristics as other cells. Manufacturers also isolate bad blocks from other valid blocks by a select transistor. This isolates the bit line and common source of the bad blocks from the valid blocks.

The bad block information must be read before any erase is attempted, because the bad block information is erasable and cannot be recovered once erased. It is highly recommended not to erase the original bad block information. Reading all the spare areas in the NAND flash memory creates a bad block table. The bad block recognition methods that build the bad block table without using the original bad block information provided in the spare areas of the memory are not equally effective.

The invalid blocks are detected at the factory during the testing process, which involves severe environmental conditions and program/erase cycles as well as proprietary test

modes. This makes tracking bad block failure mechanisms at the NASA-user level more complicated. Ideally, the time=0 bad block failures would be defined as quality defects and then, when experiencing failures during operation, as reliability degradation. Usually, these two failure types are viewed as being driven by separate mechanisms. However, in the case of NAND flash, the mechanisms may in fact be both correlated and interrelated.

The data in Figure 12 can also be re-plotted to investigate the number of bad blocks for the various devices. This is shown in Figure 13.

Figure 13 shows that one device had an excessive number of bad blocks (11) when compared to the rest of the devices where the average number of bad blocks is 4.5. The device with 11 bad blocks is two sigma away from the overall average and represents only a ~2.2% chance of occurring. If this device is considered in a separate population from the others, then it is three sigma away from that average and represents only a 0.1% chance of occurring.

This part may in fact be a statistical outlier and therefore a reliability risk. This result leads to the strong recommendation that all commercial NAND flash devices must be upscreened for use on NASA missions. Bad block analysis as demonstrated above should be included as part of this upscreening process.

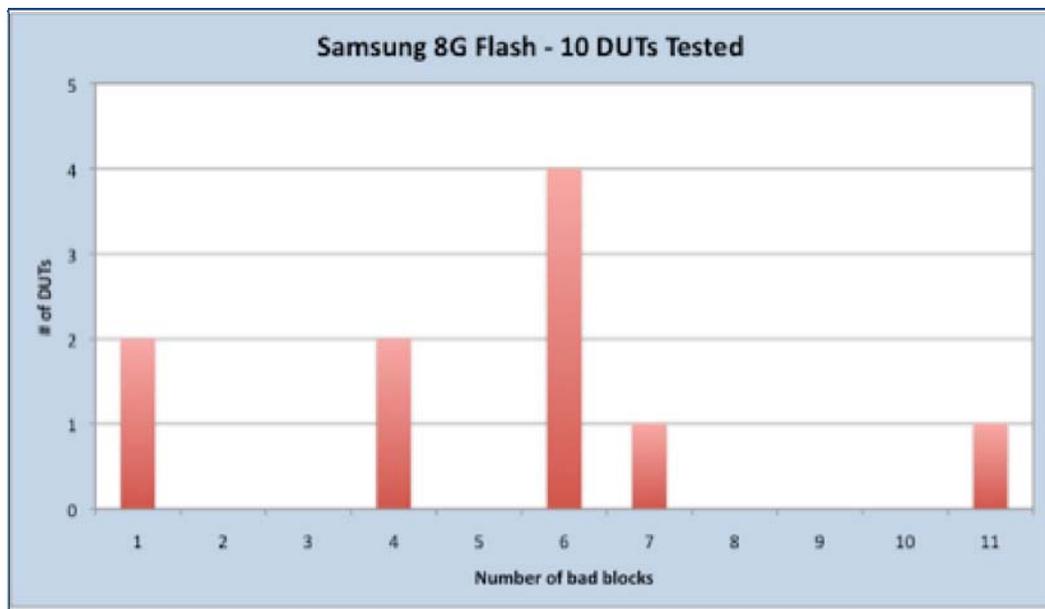


Figure 13. Number of bad blocks per number of 8 Gb NAND flash devices

3.2 Endurance Testing

Endurance testing was performed on both SLC and MLC versions of the 8 Gb Samsung NAND flash device. Table 3 provides the results.

Table 3 confirms the expected result that SLC NAND flash devices have 3–4 orders of magnitude better endurance than MLC flash cells. Endurance performance is a fundamental aspect of overall device reliability. Devices that have reduced endurance performance run the risk of having increased sensitivity to disturb errors, for example.

MLC device development has been driven by strong commercial demand for high-density storage. These MLC devices trade off this storage capability for reduced reliability. Commercial products assume much reduced lifetimes compared to NASA missions. Also commercial NAND users have sophisticated levels of software management to compensate for MLC device shortcomings [Numonyx 2008]. Such layered operations systems are not available for spacecraft with limited addressing capabilities and overall system frequency and performance.

MLC devices are not recommended for NASA flights. Only SLC devices should be considered.

Table 3. Endurance results for MLC and SLC NAND flash

Device	# Program/Erase Cycles to First Failure
K9F8G08UOM (SLC)	2.0×10^7
K9G8G08UOM (MLC)	1.0×10^3

4 CONCLUSIONS

This report documents the work done in FY09 under the NASA NEPP program for flash NVM. Efforts focused on developing an experimental test capability and then applying it to practical characterization concerns of NAND flash devices.

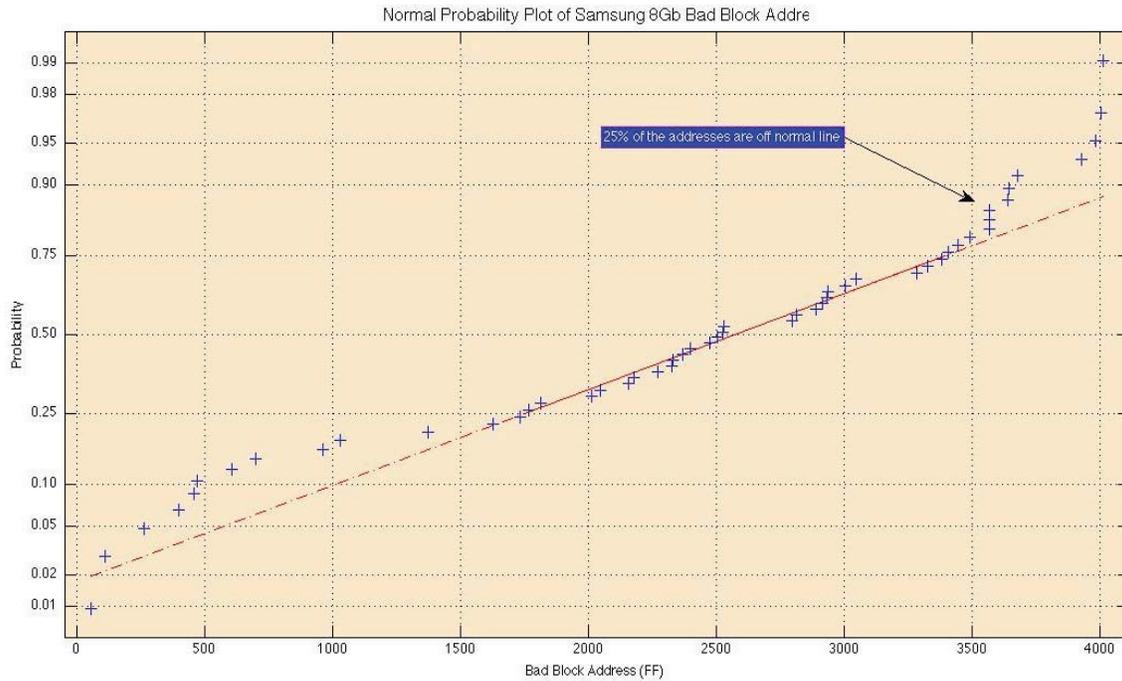
The study showed that upsampling of commercial NAND devices is a critical quality control step for NASA missions. Bad block analysis should be done as part of this upsampling as a way to identify possible reliability outlier devices that could then be excluded from further consideration. Potential preferential dependence on location for failures was also identified. This has important system implications for NASA missions. A new test capability was also used to verify the much-reduced performance of MLC NAND flash. These MLC NAND devices should not be considered for use in NASA missions.

It is recommended that further work be done in the areas of NAND flash upsampling and potential derating for mission use. The reliability of NAND flash devices has been proven to be very application-specific. Additional NEPP support will help define this application space for successful infusion of these devices into NASA missions.

REFERENCES

- Chen, Yuan, "Flash Memory Reliability NEPP 2008 Task Final Report," JPL Publication 09-9 3/09, 2009.
- Cooke, Jim, "Introduction to Flash Memory (T1A)," Flash Memory Summit, 2008.
- Irom, Farokh, and Duc Nguyen, "Radiation Tests of Highly Scaled High Density Commercial Nonvolatile Flash Memories," JPL Publication 08-27 12/08, 2008.
- Numonyx application note AN 1819, "Bad block management in NAND flash memories," <http://www.numonyx.com/Documents/Application%20Notes/AN1819.pdf>.
- Sheldon, Douglas, Pierre Maurice, Nicolas Fiant, and Pierre-Eric Berthet, "Gate Disturbance Effects in Flash Memories," NASA NEPP Program, 2007.
- Sheldon, Douglas, and Michael Freie, "Disturb Testing in Flash Memories," JPL Publication 08-7 3/08, 2008.

APPENDIX A



- **Kolmogorov-Smirnov**

$h = kstest(x)$ performs a Kolmogorov-Smirnov test to compare the values in the data vector x to a standard normal distribution. The null hypothesis is that x has a standard normal distribution. The alternative hypothesis is that x does not have that distribution. The result h is 1 if the test rejects the null hypothesis at the 5% significance level, 0 otherwise.

RESULT for bad block data $h=1$, null hypothesis is rejected.

- **Lilliefors test**

$h = lillietest(x)$ performs a Lilliefors test of the default null hypothesis that the sample in vector x comes from a distribution in the normal family, against the alternative that it does not come from a normal distribution. The test returns the logical value $h = 1$ if it rejects the null hypothesis at the 5% significance level, and $h = 0$ if it cannot.

RESULT for bad block data $h=0$, null hypothesis is cannot be rejected.

- **Interesting result.** This indicates sensitivity to sample size perhaps. We should do more analysis and more part testing to figure this out.