Rachana Shelat, Kavita Tandel ,Hiteshi Diwanji/ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 2, March -April 2013, pp.754-758

Analyze the Power Consumption of NAND Flash Memory

Rachana Shelat*, Kavita Tandel, Hiteshi M. Diwanji***

*(Department of Wireless Mobile Computing, Gujarat Technological University, India)
** (Department of Wireless Mobile Computing, Gujarat Technological University, India)
***(Department of Computer Engineering, L.D. College of Engineering, Gujarat Technological University, India)

ABSTRACT

Flash memory is the most popular solid-state memory used today. Flash is a type of EEPROM (Electrically Erasable Programmable Read-Only Memory) that supports read, program and erase as the basic operations. Although initially used only in consumer electronics, such as PDA, cellphones and portable music players, the drop in the price of NAND flash memory has paved the way for its use in mass storage devices as well, in the form of Solid State Disks (SSDs). SSDs are replacing HDDs as the storage of choice in laptops, desktops and even servers. There has been growing interest in the computer architecture community on flash memory. Computer architects have begun exploring a variety of topics related to flash, including the design of SSDs , disk-caches , new flash-based server architectures and even hybrid memories. Power is an important characteristic of NAND because the design of a NAND flash based memory array is closely related to the power consumption budget within which it is allowed to operate.

Keywords – **SSD,EEPROM,FGT,SLC,MLC**

I. INTRODUCTION

Power is an important consideration because the design of a NAND flash based memory array is closely related to the power budget within which it is allowed to operate. For example, NAND flash used in consumer electronic devices has a significantly lower power budget compared to that of a SSD used in data centers, while the power budget for NAND flash based disk caches is between the two. Therefore, we require tools that can accurately estimate the power consumption of various memory organizations and tailor the design of NAND flash to the power. We analyze FlashPower models the energy dissipated during the basic flash operations and when the chip is idle.

II. NAND Flash Memory

Depending on the type of operation to be performed, the control unit within the chip enables the decoders which use the address bits to select the appropriate physical block. The control unit is also responsible for activating the correct analog circuitry to generate high voltages needed for program and erase operations. The Flash Memory array is a two-dimensional array of semiconductor memory similar in structure to those used in other types of memories such as SRAMs and DRAMs. The array is a matrix like structure composed of rows (connected to word-lines) and columns (connected to bit-lines). At the intersection of a row and a column is a Floating Gate Transistor (FGT) which stores logical data. In this thesis, the term memory cell and the term FGT refer to the same physical entity and are used interchangeably. These memory cells that store one bit of data per cell are referred to as Single-Level Cells (SLCs), while memory cells that store multiple bits of data per cell are referred to as Multi-Level Cells (MLCs). The organization of FGTs inside the array determine whether the Flash Memory is a NAND or NOR memory.

For NOR flash, FGTs are connected in parallel to a bit-line with the control gates connected to a word-line while for NAND flash, the FGTs are connected in series. The two ends of this serial connection are connected to access transistors which in turn are connected to the bit-line. So NAND flash has fewer contacts than NOR flash and resulting in higher cell densities.

The NAND cells can be programmed faster than NOR but their read latency is higher compared to NOR. Another difference between them is that the NAND memory is not bit-programmable while the NOR memory is bit programmable.

Usually for NAND flash, read,program and erase operate at different bias conditions and the granularity of an erase differs from read/write program. Since the circuitry behaves differently for these different operations, we believe that it is more appropriate to model energy for the basic operations on the flash memory.

Finally,flash memory chip comprises of latches and sense amplifiers, which constitute the buffers. The latches store the data that is transferred to/from the memory array while the sense amplifiers sense the bit-lines during a read operation. The status register is used by the controller to monitor the status

of the command that was sent to the NAND Flash. Flash memory structure:

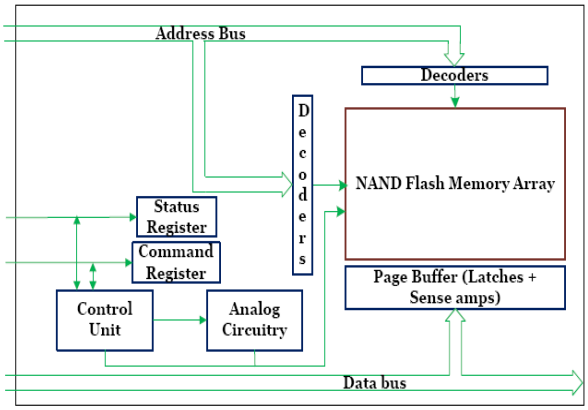

Figure 2.1 NAND flash memory Chip

Figure 2.2 shows the block diagram for the NAND memory array. The NAND flash memory array is partitioned into blocks that are, in turn sub-divided into pages. A page is the smallest granularity of data that can be addressed by the external controller. Some devices like [39] allowsub-page accesses by the controller. A set of FGTs connected in series is referred to as a string. The number of FGTs in a string is equal to the number of pages in a block. In figure 2.2, each column corresponds to a string while each row corresponds to a page. Word-lines select a page of memory to perform

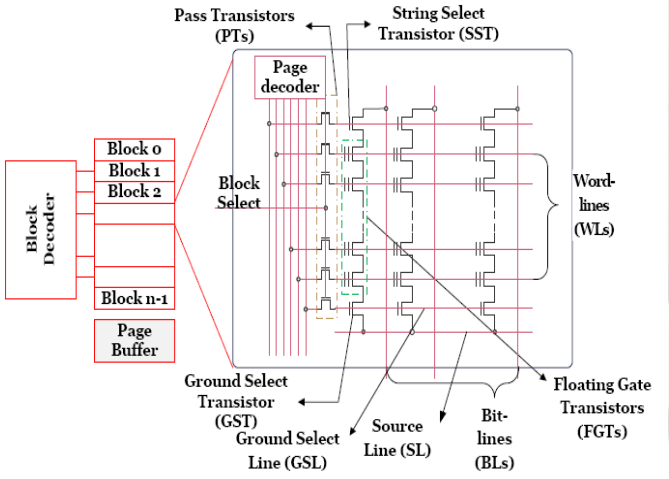

Figure 2.2 A NAND Flash Memory Array. Adapted from [7]

read or program operation. These operations first involve selecting a block using a block decoder following which one of the rows in a block is selected using a page decoder. Since NAND memory does not support in-place updates, a page needs to be erased before its contents can be programmed; but unlike a program or a read operation which work at a page granularity, the erase operation is performed at a block ranularity. The reason for this is explained in the next subsection. The drain of the String Select Transistor (SST) connected to each bit-line controls the bit-line biasing to each string while its gate is connected to the String Select Line (SSL) that switches the SST on and off. A pass transistor connected to each word-line controls the word-line biasing for each row in the array. A Ground Select Transistor (GST) connects the other end of the string to the Source Line (SL) which connects the source of one-end of the FGT to the power supply based on the type of operation to be performed.

During a page program operation, the controller transmits both the actual data and the ECC bits to the Flash memory. Upon system boot, the controller also scans the spare area of each page in the entire memory array to load the logical to physical address mapping into its own memory.

The controller uses a Flash Translation Layer (FTL) to determine the logical-to-physical address mapping. In addition to mapping, the FTL also performs garbage collection to erase stale copiesof pages that are left behind on flash due to the non in-place writes and performs wear-leveling operations to ensure that all the flash blocks wear evenly. More information about FTLs can be found in [12, 5, 15].

Single-Level Cells (SLCs) vs Multi-Level Cells (MLCs)

The operations that are explained above correspond to Single-Level Cells (SLCs) where the presence/absence of charge in the floating gate is used to represent a single bit. Since a logical bit corresponds to an analog voltage, multiple bits can be stored in a single cell by having multiple levels of voltage inside a floating gate. These are referred to as Multi-level cells (MLCs) which behave like the same way as Single-level Cells (SLCs) but require complex sense amplifiers to differentiate multiple voltage levels in the floating gate. Because varying voltage levels correspond to different logical bits, the program operation should be performed slowly to make sure that no excess or less charge is tunneled into the floating gate.

Both these factors result in the read and program time to be slower for MLCs compared to SLCs. Lifetime of MLCs is also considerably lesser than SLCs because MLCs quickly lose their ability to store varying levels of voltage.

III. Power State Machine

With respect to Figure 2.2, the components that dissipate energy are,

- The bit-line (BL) and word-line (WL) wires.
- The SSL, GSL and SL.

- The drain, source and the gate of the SST, GST and PTs.
- The drain, source and control gate of the FGTs.
- The floating gate of the FGTs - Energy dissipated during program and erase operation.

In addition to the above components, the energy dissipated by the block and page decoders, the sense amplifiers present in the page buffer (for the read operation), the charge pumps (that provide high voltages for program and erase operation) and the I/O pins are modeled. The energy perread, program and erase operation is determined by aggregating the energy dissipated by all the aforementioned components.

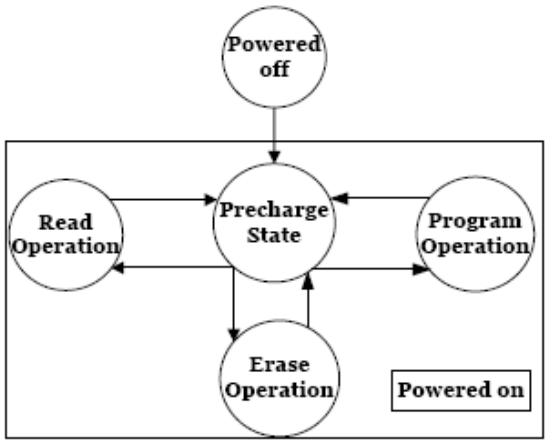

Figure 3.1 Power State Machine for a SLC NAND Chip

Before delving into the details of the model, we list the components that are modeled and the power state machine.

Figure 3.1 describes the power state machine for a SLC NAND flash chip. The circles represent the individual states, while the solid lines denote state transitions. Upon completion of the command, the Power state machine switches back to the precharge state, dissipating energy in the process.The array is isolated by the select lines but is ready to respond to commands from the controller. Upon receiving a read, program, or erase command from the controller, the state machine switches to the corresponding state.

IV. Power Modeling Methodology

Power has been extensively studied by computer architects. [8] provides a framework for analyzing and optimizing the power consumption of microprocessors. [40] provides a detailed modeling of the power consumption of hard disk drives, while has been used to study the power consumption of the memory system. [16] analyzes power from a full system perspective and quantifies the impact of application and operating system on the power behavior of the microprocessor, memory hierarchy and hard disks. Due to lack of a detailed power model for NAND flash memory, existing studies use data sheets like [39] to determine the power consumption.

However for NAND flash, read,program and erase operate at different bias conditions and the granularity of an erase differs from read/write program. Since the circuitry behaves differently for these different operations, we believe that it is more appropriate to model energy for the basic operations on the flash memory.

V. NAND Flash Power and Endurance Measurement

There have been recent efforts to understand the characteristics of flash memory by measuring and reverse engineering NAND flash chips [14, 6]. Grupp et al. [14] study the performance, power, reliability of several SLC and MLC NAND flash chips and show that the endurance of these chips tend to be much higher than their datasheet values.

The power measurements estimated by FlashPower are validated against the chip level measurements studied by [14]. Desnoyers [6] conducted a similar study of the performance and endurance characteristics of several NAND flash memory chips and found the endurance trends to be similar to those reported in [14]. These papers show that the number of P/E cycles that the pages and blocks can sustain is much higher than those given in datasheets. However, these papers do not explain the underlying cause for this trend.

VI. Conclusion

Flash memory is used in a wide range of systems varying from consumer electronics to data centers.To support such a diverse range of systems, tools that provide detailed insights into the characteristics of NAND flash memory are required. This paper describes important characteristics of NAND flash, namely power consumption. Power consumption makes NAND more reliable.

In the future, we plan to extend FlashPower to support MLC based NAND flash memory.Similar to such tool like FlashPower, we can validate the endurance model with real chip measurements and test board to perform the validation. We can also plan to model SILC to factor-in retention.

Overall, modeling all these phenomena can provide a comprehensive analysis framework for studying the power and reliability characteristics of NAND flash memory.

REFRENCES

[1] J. Abella, X. Vera, and A. Gonzalez. Penelope: The NBTI-Aware Processor. In Proceedings of the 40th IEEE/ACM International Symposium on Microarchitecture, 2007.

[2] N. Agrawal and et al. Design Tradeoffs for SSD Performance. In Proceedings the USENIX Technical Conference (USENIX), June 2008.

[3] A.M. Caulfield and L.M. Grupp and S. Swanson. Gordon: using flash memory to build fast, power-efficient clusters for data-intensive applications. In Proceedings of the International Conference on Architectural Support for ProgrammingLanguagesandOperating Systems (ASPLOS), pages 217–228, March 2009.

[4] S. Aritome and et al. Reliability issues of flash memory cells. Proceedings of the IEEE, 81(5) :776–788, May 1993.

[5] A. Birrell, M. Isard, C. Thacker, and T. Wobber. A design for high-performance flashdisks. SIGOPS Operating System Rev., 41(2):88–93, 2007.

[6] S. Boboila and P. Desnoyers. Write Endurance in Flash Drives: Measurements and Analysis. In Proceedings of the USENIX Conference on File andStorageTechnologies(FAST),February 2010.

[7] J.E. Brewer and M. Gill, editors. Nonvolatile Memory Technologies with Emphasis on Flash.IEEE Press, 2008.

[8] D. Brooks, V. Tiwari, and M.Martonosi. Wattch: A Framework for Architectural-Level Power Analysis and Optimizations. In Proceedings of the International Symposium on Computer Architecture (ISCA), pages 83–94, June 2000.

[9] C. Dirik and B. Jacob. The Performance of PC Solid-State Disks (SSDs) as a Function of Bandwidth, oncurrency, Device Architecture, and System Organization. In Proceedings of the International Symposium on Computer Architecture (ISCA), pages 279–289, June 2009.

[10] Taehee Cho and et al. A dual-mode NAND flash memory: 1-Gb multilevel and highperformance 512-Mb single-level modes. Solid-State Circuits, IEEE Journal of, 36(11):1700–1706, Nov 2001.

[11] P. Desnoyers. Empirical Evaluation of NAND Flash Memory Performance. In Proceedings of the Workshop on Hot Topics in Storage and File Systems (HotStorage), October 2009.

[12] E. Gal and S. Toledo. Algorithms and Data Structures for Flash Memories. ACM Computing Surveys, 37(2):138–163, June 2005.

[13] G.R. Ganger, B.L. Worthington, and Y.N. Patt. The DiskSim Simulation Environment Version http://www.pdl.cmu.edu/DiskSim/.

[14] L. Grupp and et al. Characterizing flash memory: Anomalies, observations, and applications. In Microarchitecture, 2009. MICRO-42. 2009 42nd IEEE/ACM International Symposium on, Nov. 2009.

[15] A. Gupta, Y. Kim, and B. Urgaonkar. DFTL: a flash translation layer employing demand-based selective caching of page-level address mappings. In ASPLOS '09: Proceeding of the 14th international conference on Architectural support for programming languages and operating systems, pages 229–240, 2009.

[16] S. Gurumurthi and et al. Using Complete Machine Simulation for Software Power Estimation: The SoftWatt Approach. In Proceedings of the 8th International Symposium on High-Performance Computer Architecture, page 141. IEEE Computer Society, 2002.

[17] S. Gurumurthi, A. Sivasubramaniam, and V. Natarajan. Disk Drive Roadmap from the Thermal Perspective: A Case for Dynamic Thermal Management. In Proceedings of the International Symposium on Computer Architecture (ISCA), pages 38–49, June 2005.

[18] Intel X25-E Extreme SATA Solid-State Drive. http://www.intel.com/design/flash/nand/extreme /index.htm.

[19] K. Ishida and et al. A 1.8V 30nJ adaptive program-voltage (20V) generator for 3D-integrated NAND flash SSD. In Solid-State Circuits Conference - Digest of Technical Papers, 2009. ISSCC 2009. IEEE International, pages 238–239,239a, Feb. 2009.

[20] Process Integration and Device Structures, ITRS 2007 Edition.http://www.itrs.net/Links/2007ITRS/2007 Chapters/2007 PIDS.pdf.

[21] T. Kgil, D. Roberts, and T. Mudge. Improving NAND Flash Based Disk Caches. In Proceedings of the International Symposium on Computer Architecture (ISCA), pages 327–338, June 2008.

[22] J.K. Kim and et al. A 120-mm2 64-Mb NAND flash memory achieving 180 ns/Byte effective program speed. Solid-State Circuits, IEEE Journal of, 32(5):670–680, May 1997.

[23] J.K. Kim and et al. A PRAM and NAND flash hybrid architecture for high-performance embedded storage subsystems. In EMSOFT '08: Proceedings of the 8th ACM international conference on Embedded software, pages 31–40, New York, NY, USA, 2008. ACM.

[24] S.V. Kumar, C.H. Kim, and S.S. Sapatnekar. Impact of NBTI on SRAM Read Stability and Design for Reliability. In Proceedings of the

International Symposium on Quality Electronic Design, 2006.

[25] J. Lee and et al. A 1.8V NAND Flash Memory for Mass Storage Applications. In Proceedings of the IEEE International Solid-State Circuits Conference (ISSCC), pages 290–494, February 2003.

[26] J. Lee and et al. A 90-nm CMOS 1.8-V 2-Gb NAND flash memory for mass storage applications. Solid-State Circuits, IEEE Journal of, 38(11):1934–1942, Nov 2003.

[27] M. Lenzlinger and E.H. Snow. Fowler-Nordheim tunneling into thermally grown SiO2. Electron Devices, IEEE Transactions on, 15(9):686–686, Sep 1968.

[28] Y. Manabe and et al. Detailed Observation of Small Leak Current in Flash Memories with Thin Tunnel Oxides. In Proceedings of the International Conference on Microelectronic Test Structures (ICMTS), pages 95–99, March 1998.

[29] N. Mielke and et al. Flash eeprom threshold instabilities due to charge trapping during program/ erase cycling. Device and Materials Reliability, IEEE Transactions on, 4(3):335–344, Sept. 2004.

[30] N. Mielke and et al. Recovery effects in the distributed cycling of flash memories. In Reliability Physics Symposium Proceedings, 2006. 44th Annual., IEEE International, pages 29–35, March 2006.

[31] Neal R. Mielke. Intel Corporation, October 2009. Private Correspondence.

[32] H. Nakamura and et al. A 125mm2 1Gb NAND Flash Memory with 10 MB/s Program Throughput. In Proceedings of the IEEE International Solid-State Circuits Conference (ISSCC), pages 106–107, February 2002.

[33] D. Narayanan, A. Donnelly, and A. Rowstron. Write Off-Loading: Practical Power Management for Enterprise Storage. In Proceedings of the USENIX Conference on File and Storage Technologies (FAST), February 2008.

[34] D. Narayanan and et al. Migrating Enterprise Storage to SSDs: Analysis of Tradeoffs. In Proceedings of EuroSys 2009, March 2009.

[35] Y. Park and D.K. Schroder. Degradation of thin tunnel gate oxide under constant fowlernordheim current stress for a flash eeprom. In IEEE Transactions on Electron Devices, 1998.

[36] S. Kavalanekar and et al. Characterization of storage workload traces from production Windows servers. In Proceedings of the IEEE International Symposium on Workload Characterization (IISWC), pages 119–128, October 2008.

[37] S. Lee and et al. FlexFS: A Flexible Flash File System for MLC NAND Flash Memory. In Proceedings of the USENIX Annual Technical Conference (USENIX), June 2009.

[38] Koji Sakui. Intel Corporation/Tohoku University, May 2009. Private Correspondence.

[39] Samsung corporation. K9XXG08XXM flash memory specification.

[40] S. Sankar, Y. Zhang, S. Gurumurthi, and M.R. Stan. Sensitivity-Based Optimization of Disk Architecture. IEEE Transactions on Computers, 58(1):69–81, January 2009.

[41] R.K. Sharma and et al. Balance of power: dynamic thermal management for internet data centers. Internet Computing, IEEE, 9(1):42–49, Jan.-Feb. 2005