JOURNAL OF INFORMATION, KNOWLEDGE AND RESEARCH IN ELECTRONICS AND COMMUNICATION ENGINEERING

ONFI 2.0 COMPLIANT NANAD FLASH MEMORY CONTROLLER

[1] MS. RIZU M. RAVAL, [2] MR. NIRANJAN RAYAPROLU

[1]M.E.[VLSI & Embedded System designs] Student, Department Of Electronics & Communication Engineering, Gujarat Technological University Ahmedabad, Gujarat
[2] Member of Technical staff, SEER Akademi, Hyderabad

rizuraval14@yahoo.com, niranjanr@seerakademi.com

ABSTRACT *: ONFI 2.0 Complaint NAND FLASH Controller IP core is full featured, easy to use ,synthesizable design that is easily integrated into any SoC FPGA development. Designed to support both SLC and MLC Flash memories, it can be verified and implemented. The controller works with any suitable memory device up to 512 Gb from leading memory provider such as Micron, Samsung, Toshiba and others. The controller supports a variety of host bus interface for easy adoption into any design architecture. The NAND Flash Memory Controller is best suitable for controlling embedded storage (e.g. in mobile device, network routers).Incorporating this design in embedded systems will help to get rid of CPU for NAND flash memory access.*

Keywords— Non-Volatile Memory, Flash Memory, NAND Flash Memory, Memory Controller, NAND Flash Memory Controller, ONFI 2.0, SLC Flash Memory, MLC Flash Memory.

I: INTRODUCTION

Flash memory has become a powerful and cost-effective solid-state storage technology widely used in mobile electronics devices and other consumer applications. Two major forms of Flash memory are: NAND Flash and NOR Flash, have emerged as the dominant varieties of non-volatile semiconductor memories utilized in portable electronics devices. Embedded system offers various processors for different types of operations, but there is a limited usage of memories in embedded systems. This controller can be used across all ONFI compliant NAND flash memories in an embedded system. So, the controller overcomes the issue of limited usage of memory. Basically, NAND technology provides a cost-effective solution for applications requiring high density solid-state storage.

In the internal circuit configuration of NOR Flash, the individual memory cells are connected in parallel, which enables the device to achieve random access. This configuration enables the short read times required for the random access of microprocessor instructions. NOR Flash is ideal for lower-density, high-speed read applications, which are mostly read only, often referred to as code-storage applications. NAND Flash was developed as an alternative optimized for high-density data storage, giving up random access capability in a tradeoff to achieve a smaller cell size, which translates to a smaller chip size and lower cost-per-bit. This was achieved by creating an array of eight memory transistors connected in a series. Utilizing the NAND Flash architecture's high storage density and smaller cell size, NAND Flash systems enable faster write and erase by programming blocks of data. It was also noted in the datasheet of Micron company for 2, 4, 8Gb: x8/x16 Multiplexed NAND Flash Memory. NAND Flash is ideal for low-cost, high-density, high-speed program/erase applications, often referred to as data-storage applications.

The Open NAND Flash Interface Working Group is a consortium of technology companies working to develop open standards for NAND flash memory chips and devices that communicate with them. ONFI 2.0 defines a high-speed NAND Flash interface that can deliver speeds greater than 133 MB/s, whereas the legacy NAND interface was limited to 50 MB/s. The full ONFI 2.0 specification was released in February of 2008.

A typical NAND Flash memory contains 2,048 and 4,096 erasable blocks respectively. Each block is subdivided into 64 programmable pages. Each page consists of 2,112 bytes (x8) or 1,056 words (x16). The pages are further divided into a 2,048-byte data storage region with a separate 64-byte area on the x8 device; and on the x16 device, separate1,024-word and 32-word areas. The 64-byte and 32-word areas are typically used for error management functions. The contents of each 2,112-byte page can be programmed in 300µs, and block can be erased in 2ms. ERASE/PROGRAM endurance is specified at 100,000 cycles when using appropriate error correcting code (ECC) and error management.

II:NAND FLASH MEMORY OVERVIEW:

In NAND Flash Memory Data, commands, and addresses are multiplexed onto the same pins. This provides a memory device with a low pin count. The architecture for NAND Flash Memory is shown below:

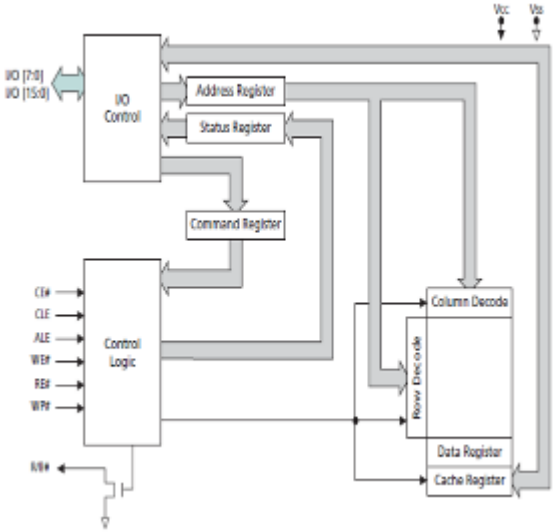

Fig 1 NAND Flash Memory Architecture

The internal memory array is accessed on a page basis. When doing reads, a page of data is copied from the memory array into the data register. Once copied to the data register, data is output sequentially, byte-by-byte on x8 devices, or word-by-word on x16 devices. The memory array is programmed on a page basis. After the starting address is loaded into the internal address register, data is sequentially written to the internal data register up to the In the cache programming mode, data is first copied into the cache register and then into the data register. Once the data is copied into the data register, programming begins. After the data register has been loaded and programming started, the cache register becomes available for loading additional data. Loading the next page of data into the cache register takes place while page programming is in process. The INTERNAL DATA MOVE command also uses the internal cache register. Normally, moving data from one area of external memory to another uses a large number of external memory cycles. By using the internal cache register and data register, array data can be copied from one page and then programmed into another without using external memory cycles.

III: NAND Flash Memory Signals:

Address latch enable (ALE): During the time ALE is HIGH, address information is transferred from I/O[7:0] into the on-chip address register upon a LOW-to-HIGH transition on WE#. When address information is not being loaded, the ALE pin should be driven LOW.

Command latch enable (CLE): When CLE is HIGH, information is transferred from I/O[7:0] to the on-chip command register on the rising edge of WE#. When command information is not being loaded, the CLE pin should be driven LOW.

Chip enable (CE): Gates transfers between the host system and the NAND device. Once the device starts a PROGRAM or ERASE operation, the chip enable pin can be de-asserted.

Data inputs/output (I/O[7:0]): The bidirectional I/O pins transfer address, data, and instruction information. Data is output only during READ operations; at other times the I/O pins are inputs.

Ready/busy(R/B) #: An open-drain, active-LOW output that uses an external pull-up resistor. The pin is used to indicate when the chip is processing a PROGRAM or ERASE operation. The pin is also used during a READ operation to indicate when data is being transferred from the array into the serial data register. Once these operations have completed the R/B #returns to the High-Z state.

Read enable (RE#): Gates transfers from the NAND device to the host system.

Write enable (WE#): Gates transfers from the host system to the NAND device.

Write protect (WP#): Pin protects against inadvertent PROGRAM and ERASE operations. All PROGRAM and ERASE operations are disabled when the WP# pin is LOW.

IV:NAND FLASH MEMORY COMMAND DEFINITION:

Commands are written to the command register on the rising edge of WE#, when CE# and ALE are LOW, CLE is HIGH, and device is not busy. The exceptions to this are the READ STATUS and RESET commands. Commands are transferred to the command register on the rising edge of WE#. After this commands are input on I/O[7:0] only. On initial power up, each device defaults to read mode.

Page Read: To enter the read mode while in operation, write the 00h-30h command sequence to the command register along with the five ADDRESS cycles. Writing 00h to the command register starts the ADDRESS LATCH cycle. Five ADDRESS cycles are input next. Finally the 30h command is loaded into the command register. After 30h is written, the page data is transferred to the data register, and R/B# goes LOW during the transfer. When the transfer to the data register is complete, R/B# returns HIGH.

Random Data Read: The RANDOM DATA READ command enables the user to specify a new column address so the data at single or multiple addresses can be read. The random read mode is enabled after a normal PAGE READ (00h-30h sequence).

Random data can be output after the initial page read by writing an 05h-E0h command sequence along with the new column address (two cycles).

Read ID: The READ ID command is used to read the 5 bytes of identifier codes programmed into the devices. The READ ID command reads a 5-byte table

that includes Manufacturer's ID, device configuration, and part-specific information. Writing 90h to the command register puts the device into the read ID mode. At the time of writing 90h to I/O[7:0] CLE stays high. After that 00h is written to command register the device gains address for performing operation. At that time 00h is written to I/O[7:0] and CLE remains high. Device will give appropriate data to controller on I/O[7:0] , after performing the operation.

Read Status: After the READ STATUS command has been issued to the NAND Flash device, all subsequent READ cycles will output data from the status register until another command is issued. In addition, after a READ STATUS command has been issued to the NAND Flash device, the status register provides continually updated output on I/O[7:0] as long as CE# and RE# are held LOW.

Program Page: Micron NAND Flash devices are inherently page-programmed devices. Within a block, the pages must be programmed consecutively from the least significant bit (LSB) page of the block to most significant bit (MSB) pages of the block.

Read for Internal Data Move: This READ command is used in conjunction with the INTERNAL DATA MOVE (85h-10h) command. First, (00h) is written to the command register, and then the internal source address is written (five cycles). After the address is input, the READ

For INTERNAL DATA MOVE (35h) command writes to the command register. This transfers a page from memory into the cache register.

Block Erase: The BLOCK ERASE command operates on one block at a time. Three cycles of addresses A[28:18] are required for the x8 device. The ERASE SETUP (60h) command is first written to the command register. Then three cycles of addresses are written to the device. Next, the ERASE CONFIRM (D0h) command is written to the command register. At the rising edge of WE#, R/B# goes LOW. The READ STATUS REGISTER command can be used to check the status of the ERASE operation. When bit 6 = "1" the erase operation is complete. Bit 0 indicates a pass/fail condition where "0" = pass.

Reset: The RESET command is used to put the memory device into a known condition and to abort a command sequence in progress. RANDOM READ, PROGRAM, and ERASE commands can be aborted while the device is in the busy state. The contents of the memory location being programmed or the block being erased are no longer valid. The data may be partially erased or programmed, and is invalid. The command register is cleared and is ready for the next command.

V NAND FLASH MEMORY CONTOLLER OVERVIEW :

NAND Flash devices use a highly multiplexed 8- or 16-bit bus (I/O[7:0] or I/O[15:0]) to transfer data, addresses, and instructions. The five command pins (CLE, ALE, CE#, RE#, WE#) implement the NAND command bus interface protocol. The additional pins control hardware write protection (WP#), monitor device status (R/B).

Architectural Block Diagram of ONFI 2.0 NAND Flash Memory Controller:

Functional Block Diagram of NAND Flash Memory Controller is shown in figure1. It consists of different modules such as, Command Interface Block, Read FIFO, Address Mapping Block, and FSM.

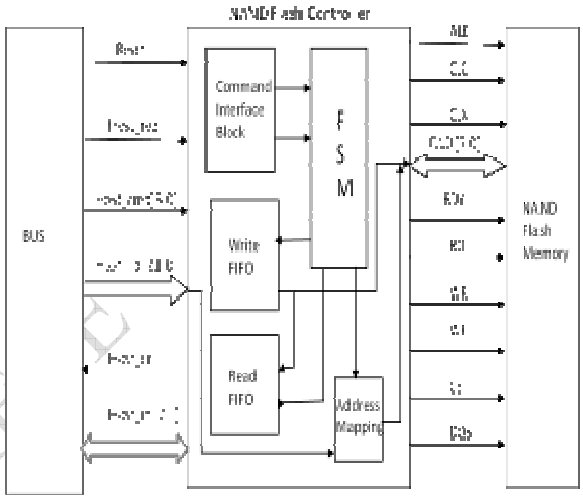

Fig.2 Functional Block Diagram

Control Interface Module: Control Interface Module accepts command from the host and decodes the command and passes the request to the Command module.

FIFO*:* FIFOs are commonly used in electronic circuits for buffering and flow control for rate buffering. In hardware form a FIFO primarily consists of a set of read and write pointers, storage and control logic. In case of Write FIFO, the data to be written to memory are stored. Read FIFO stores data coming from memory after performing operation on memory.

Address Mapping: The NAND Flash memory controller receives NAND Flash memory access requests along with a 22-bit logical address from the host. NAND Flash memory controller uses the logical address to generate an address for the NAND Flash memory. The generated address is given to the control FSM and command generation module.

***FSM*:** Finite State Machine. This module accepts command from the Control Interface Module and address from the address generation module and generates the proper commands to NAND Flash memory.

State Diagram for NAND Flash Memory Controller:

In the state Diagram for ONFI 2.0 Compliant FPGA Based Memory Controller, the controller begins operation at power ON. After that it immediately goes to idle state. From the idle state, controller enters to read state because memory initiates with the Read operations. Controller performs various read operation. At the end of each operation, the controller goes to idle state.

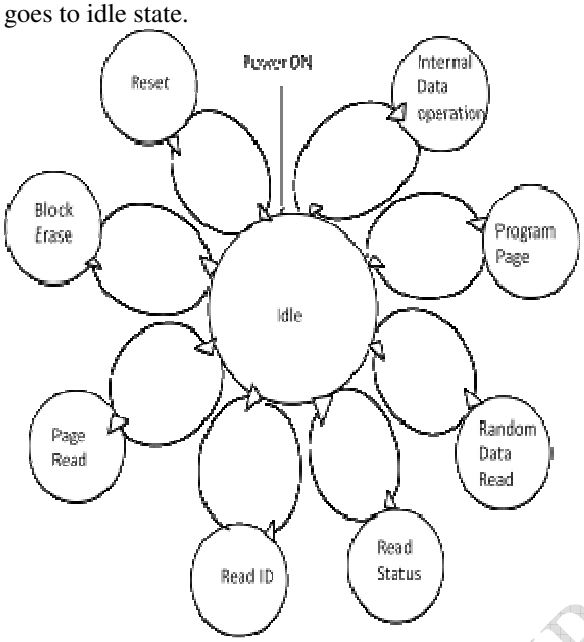

Fig.3 State Diagram of Controller

After completing Read operation, it performs operation like erase, Program page, internal data operation etc.

VI SIMULATION RESULTS:

NAND Flash Memory controller performs certain operations on memory. Based on that, there are certain simulation results which qualify the solution. NAND Flash Memory controller performs operations like Page Read, Block Erase, Read ID, etc.
Simulation result for Block Erase operation is shown in the figure below:

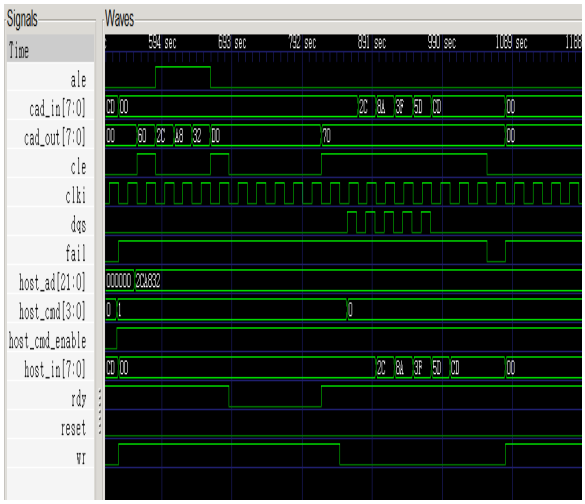

Fig.4 Simulation Results for Block Erase operation

Simulation result for block erase operation from the data sheet of NAND Flash memory controller is shown here:

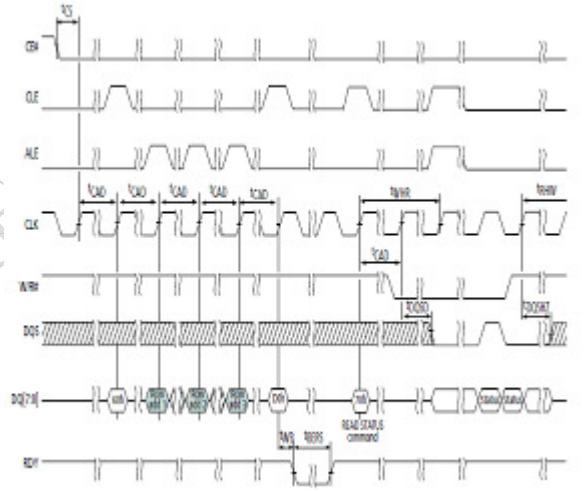

Fig 5 Timing Diagram for Block Erase operation
Courtesy: Datasheet, Micron Company

Simulation result for Read ID operation is shown in the figure shown below:

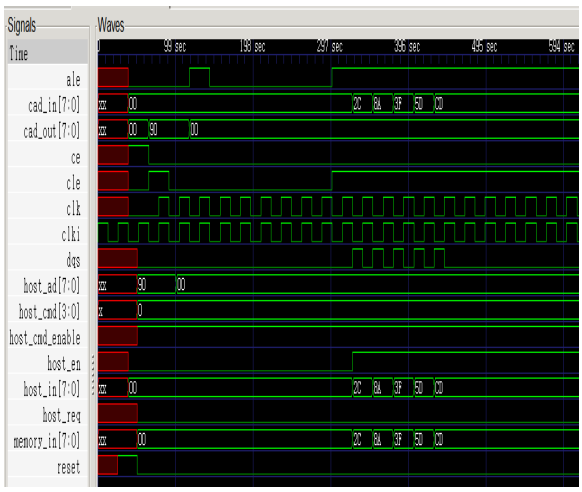

Fig.6 Simulation result for Read ID operation

Simulation result for Read ID operation from the data sheet of NAND Flash memory controller is shown here for reference (in figure 6)

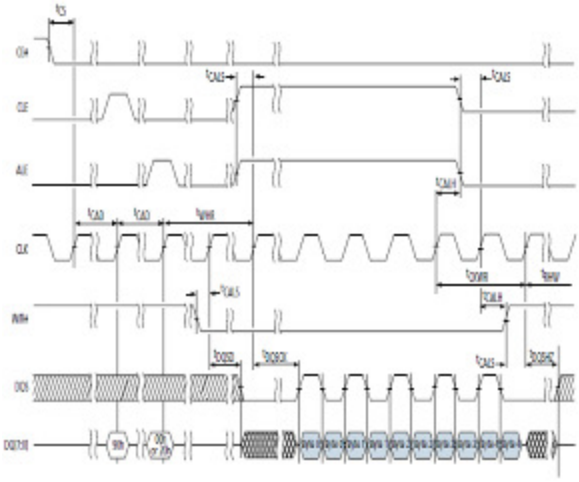

Fig.7 Timing Diagram for Read ID operation
Courtesy: Datasheet, Micron Company

On comparing the simulation result with the timing diagram of the datasheet, the Block Erase operation is verified.

VII CONCLUSION:

The ONFI 2.0 Compliant NAND Flash Memory Controller design, that I undertook, perform various operations that conform to the requirement of ONFI. This can be verified by comparing the simulated results of the design with the ONFI 2.0 Compliant Memory datasheet.

This design can be readily used for FPGA based embedded system having NAND Flash Memory requirement.

The solution that I am providing is extremely useful for the FPGA based electronic systems, in need of high volume, high data rate portable data reading system.

VIII ACKNOWLEDGEMENT:

I hereby would like to acknowledge the invaluable support of my project guide Mr. Shrikanth Jadcherla, CEO of SEER Akademi and faculty members of Gujarat Technological University. I owe my special thanks to all the faculty members of the SEER Akademi and Gujarat Technological University for the support throughout the duration of the project and have been an unending source of inspiration for me.

IX REFERENCES:

[1]J. J. Chang, "Non Volati1e Semi conductor Memory Devices", PROC of th e IEEE , pp. 1039-1059, July 1976.

[2]P. Cappelletti et al., Flash Memories. Norwood, MA: Kluwer Academic, 1999.

[3]Herman E. Maes, Katholieke Universiteit Leuven ESAT LaboratoryKardinaal Mercierlaan 94, B-3030 Heverlee – Belgium "RECENT DEVELOPMENTS IN NON-VOLATILE SEMICONDUCTOR MEMORIES".

[4]Roberto Bez and Numonyx, Italy "Overview of advanced Non-Volatile Memory Technology". In Proc.International Test Conference, pages 37–46, 7–10 Oct. 2002.

[5]G. Campardo, R. Micheloni, and D. Novosel, VLSI-Design of Non-Volatile Memories.

[6] F. Sun, S. Devarajan, K. Rose and T. Zhang " Design of on-chip error correction systems for multilevel NOR and NAND flash memories" IET Circuits Devices Syst., 2007, 1, (3), pp. 241–249 241

[7]NAND Flash Memory Datasheet of Micron for 64Gb, 128Gb, 256Gb, 512Gb Asynchronous/Synchronous NAND.

[8]NAND Flash Memory Datasheet of Micron for 2, 4, 8Gb: x8/x16 Multiplexed NAND Flash Memory

[9]NAND Flash Memory Datasheet of Toshiba for 32 Gb: x8/x16 Multiplexed NAND Flash Memory