Single-chip Integration of SRAM and Non-volatile Memory using Bit-line Sharing

David Choi*, Eui Pil Kwon†, Hyaeryoung Lee†, John Chang†, Kyu Choi†, and John Villasenor*
*EE Dept. - University of California, Los Angeles CA
†O2IC Co. Ltd, Santa Clara CA

***Abstract*— A new memory architecture integrating SRAM and Flash within the same array is presented and demonstrated in a standard 0.25μm CMOS process with a memory access time of 20ns. Differential pair Flash cells with low programming current share the same bit-lines as SRAM cells within the same array. This enables row-to-row transfer of data between Flash and SRAM cells as well as access data through I/O directly, in return improving speed and lowering power. Area is saved through the shared usage of column decoding, sense-amplifier, and write-driver circuitry.**

I. Introduction

Many of today's system architectures, including those used for cellular phones, digital cameras, and mobile platforms, utilize both volatile memory and non-volatile memory. Volatile memories, such as SRAM or DRAM, have the advantage of having relatively faster access times, while non-volatile memories such as Flash have the advantage of retaining data when the power is turned off.

An "ideal" memory technology would have the fast speed of SRAM, the non-volatility of Flash, and the high density of DRAM, and thereby eliminate the need for multiple types of memories. There are a number of emerging memory technologies, such MRAM, FeRAM, and PC-RAM, with the potential to deliver these qualities. However, while these technologies are promising, there are still a number of challenges and issues associated with each technology [1]–[3], thus necessitating the usage of two or more types of memories in many systems.

The inherent disadvantage of having multiple types of memory in the same system lies in the frequent need to transfer data among the memory types, often incurring penalties in terms of speed, power, manufacturability, and in the additional logic required to facilitate the transfer. High voltages and high programming currents exceeding 30μA prohibit sharing of the Flash source/drain nodes with any node of a low voltage RAM cell, as this would cause damage and degradation to the RAM cell. In addition, floating gate based Flash devices require a different manufacturing process than RAM, and require a large number of additional mask steps to fabricate. Thus Flash and RAM are generally implemented separately.

In the present paper, we describe a new architecture in which both SRAM and Flash are integrated on the same array and demonstrated in standard 0.25μm CMOS. This integration is enabled by the use of polysilicon-oxide-nitride-oxide-silicon (SONOS) technology in combination with a split gate structure for reducing programming currents, and is demonstrated using an architecture in which the SRAM and Flash devices on the array share the same set of bit lines.

Inclusion of both Flash and RAM within the same memory array provides significantly higher bandwidths between the two memories, faster speeds, lower power, lower effective chip area, reduced board area, practical manufacturing, lower costs, and greater flexibility than previously realized.

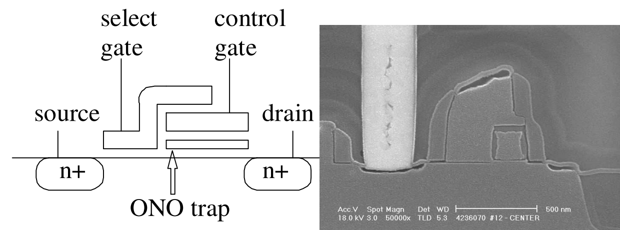

Fig. 1. Cross section of the non-volatile device.

II. Non-volatile Device

The integrated memory utilizes a SONOS device with a split-gate structure. Split-gate structures have been previously shown to provide high efficiency source side injection [4], in which low programming currents are made possible [5]. The non-volatile device utilizes this principle in a self-aligned split gate SONOS structure with improvements to support practical manufacturing [6], [7]. The SEM cross section is shown in Fig. 1. An ONO (oxide-nitride-oxide) dielectric lies under a control gate, which in turn lies next to a self-aligned select gate. The select gate is used for source side injection, thereby increasing the threshold voltage.

Programming of the non-volatile device is performed by applying the following voltages: 8.0V - 9.5V to the control gate (Vcg), 1.0V - 1.5V to the select gate (Vsg), 4.5V - 7.5V to the drain (Vd, VPP), and 0V to the source (Vs). The measured threshold shift with respect to programming time is shown in Fig. 2 for Vs=0V and different control gate voltages.

The threshold shift was found to have little dependence on the drain bias. Programming of the device is however dependent on the source bias. If the source is biased to a voltage larger than 1.5V, the select gate voltage Vgs will be less than the threshold voltage of the select gate, so there will not be any source side injection current and the non-volatile device will not be programmed. This is shown in Fig. 3, where the curves for Vs=0V, 1.0V, and 2.0V are

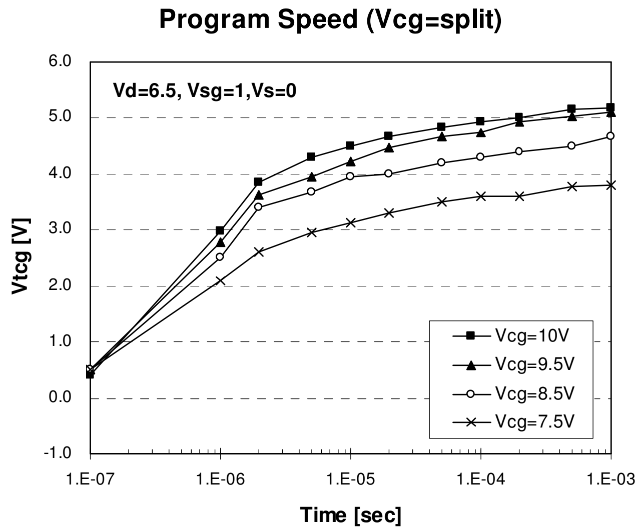

Fig. 2. Threshold voltage shift vs. program time, at different control gate voltages.

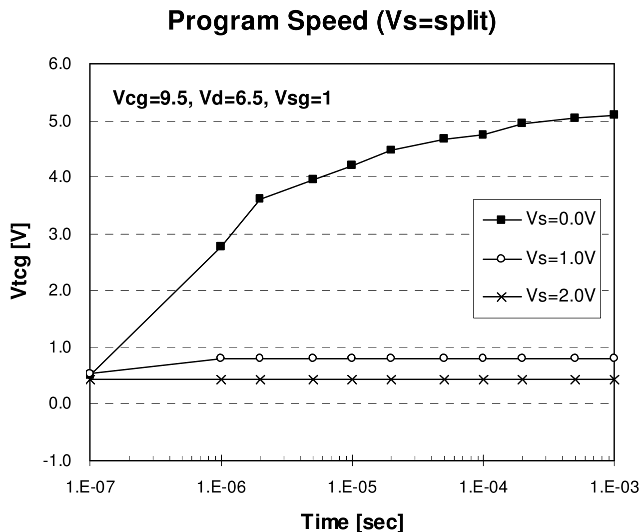

Fig. 3. Threshold voltage shift vs. program time, at different source voltages.

drawn, and Vcg=9.5V, Vd=6.5V, and Vsg=6.5V. The Vs=1.0V curve shows very little shift in the threshold voltage, and the Vs=2.0V curve shows no shift in the threshold voltage.

The source node of the non-volatile device can be tied to a bit-line node. Thus, in order to program the device, the bit-line should be biased to 0V. If the bit-line is biased high, e.g. to 3V, the device is not programmed, even when programming voltages are applied to the control gate, the select gate, and the drain.

The charge in the trapped ONO layer can be erased by tunneling through the bottom oxide layer. This is accomplished by introducing a high negative voltage on the control gate, and applying ground to the bulk region. In the system and experimental results described here, tunneling erase is used

TABLE I
PROGRAM, READ, AND ERASE VOLTAGES

	Vcg	Vsg	Vpp	bit-line
Program	9.5V	1.2V	6.5V	0V
Read	2.0V	3.0V	3.0 V	0V
Erase	-9.5V	-1.5V	floating	floating

and is performed by applying -9.5V to the control gate. It was found that applying a tunneling erase operation for 100ms results in a 3.4V threshold drop.

The use of the select gate prevents over-erase problems. Traditionally, specific measures must be taken in the design process to avoid this. However, in the split-gate structure, a select-gate prevents any drain current, thereby circumventing the over-erase problem.

Reading is accomplished by applying 3V to the select gate, 2V to the control gate, and 1.0V - 3.0V to the Vpp/drain. The device current is 40μA for the erased state and less than 1.0μA in the programmed state. Optimum conditions for program, erase, and read are summarized in Table I.

III. Non-Volatile Device Integration With SRAM

There are many possible methods for integrating the non-volatile device with RAM. One approach, storage node sharing (“SN sharing”) involves attaching the non-volatile device inside RAM via an internal node. Another approach is to share the bit line for both Flash and RAM.

The BL sharing approach is applied here, using two non-volatile devices that are arranged in a differential pair configuration. The drain of both non-volatile devices are connected a global Vpp line, while the control gates are connected together and the select gates are connected together. The source of one device connects to a bit line while the source of the other connects to the complementary bit line.

Programming of the differential pair Flash cell occurs by first biasing the bit-lines to a low voltage on one side and a high voltage on the other through either SRAM or external I/O. Then programming voltages are applied to both devices so that one side is programmed and the other side remains in the erased state. Note that an erase step must be performed before programming.

To read the contents of a programmed differential pair Flash cell, the bit-lines are first discharged to ground. Then the Flash read voltages are applied to the differential pair. The programmed non-volatile device does not conduct current and the erased non-volatile device conducts current according to the read bias conditions.

IV. Bit-line Sharing Test Chip Architecture

A test chip combining differential pair Flash and SRAM, integrated in the manner of Fig. 5, was fabricated. For the array, a combination of 8 rows of SRAM and 128 rows of Flash was implemented, with each row containing 32 columns,

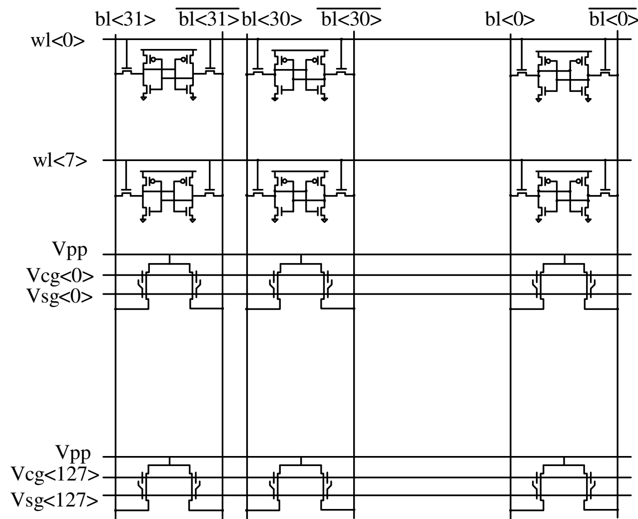

Fig. 4. Circuit diagram of the bit-line sharing array.

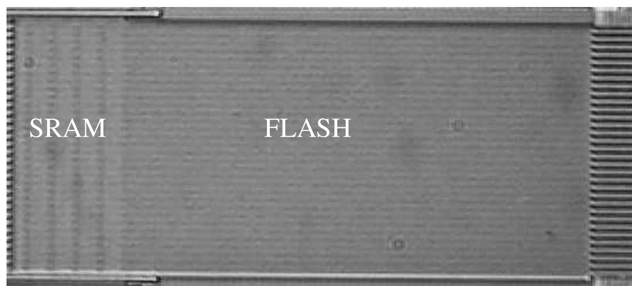

Fig. 5. Die photo of the bit-line sharing array.

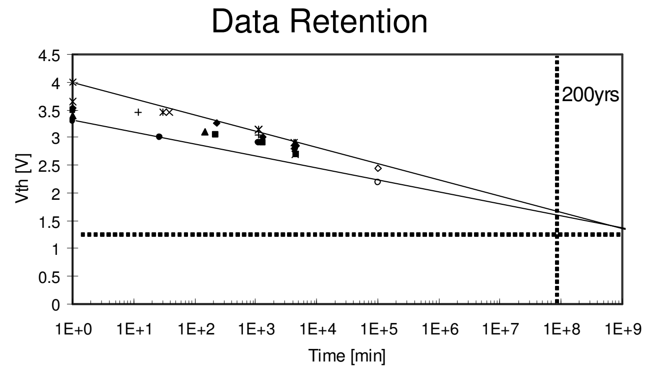

Fig. 6. Data retention of the non-volatile device.

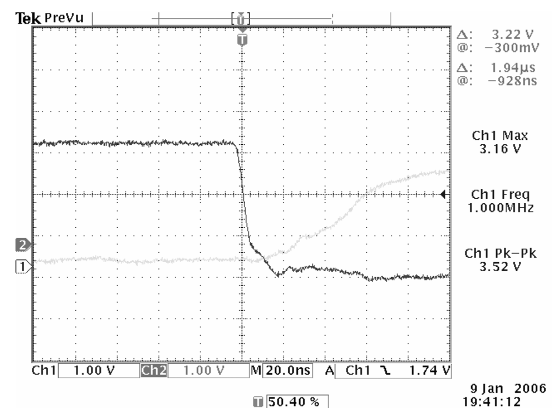

Fig. 7. Access Time Waveform.

for a total of 256 SRAM cells and 4096 Flash cells. The processed array is shown in Fig. 4.

The column decoder, sense amplifier, and write driver circuitry were used for both the SRAM and Flash arrays, thus saving area. The following operations were implemented using the bit-line sharing approach: SRAM read, SRAM write, Flash read, Flash program, Flash Erase, Flash to SRAM transfer, and SRAM to Flash transfer.

The SRAM read and SRAM write operations are the same as the normal SRAM operations. The Flash read and Flash program operations are also bit level operations that read and write to a single Flash cell.

The Flash to SRAM transfer operation entails the simultaneous transfer of data from one selected row of Flash cells to one selected row of SRAM cells. Any row of Flash and any row of SRAM can be independently selected for the transfer. This transfer is performed without the use of a sense-amplifier or the use of a write driver, and the transfer occurs simultaneously across the entire row. The SRAM to Flash transfer operation moves data from one selected row of SRAM cells to one selected row of Flash cells simultaneously. The row-based transfer operations have the advantage of allowing programming an entire row at the same time and in parallel, without having to perform the reading and programming of individual cells serially. The Flash erase operation is an array-based operation and causes all Flash cells within the array to be erased at the same time. Additional control logic was implemented to manage the address decoder, sense amplifier, and write driver circuitry during these various operations.

V. Fabrication and Validation

The bit-line sharing architecture was successfully demonstrated in a test-chip in a 0.25μm CMOS logic process with an appropriate process modification. The micrograph of the array is shown in Fig. 5. The modification of the process was made as a non-volatile process module to retrofit the logic process. One additional mask step was used for the non-volatile memory devices, and three additional mask steps were used for the high voltage gates.

The size of the non-volatile device was 10.8F^2. Gate oxide thickness for the logic device was 55Å. The non-volatile device can be scaled to a .04μm CMOS logic process with the appropriate process modification; e.g. the device structure does not impose any new scaling limitations. Data retention at room temperature is estimated to be over 200 years, as shown in Fig. 6. At 85C it is anticipated to be at least 20 years.

Results from the measurements on integrated memory array

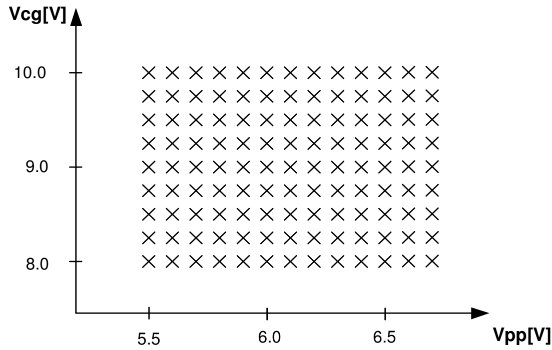

Fig. 8. Programming Voltage Range.

TABLE II
TIMING RESULTS

	Time	Vcg	Vpp
Program	10 μs	9.5V	6.5V
Erase	20 ms	-9.5V	0V
Access	20 ns	3.0V	0V

TABLE III
OPERATING VOLTAGE RANGE

	Voltage Range
Programming	
Vcg	8.0V to 9.5V
Vpp	4.5V to 7V
Erase	
Vcg	-8.5V to 9.5V
SRAM to Flash Transfer	
Vcc	2.5V to 3.7V
Flash to SRAM transfer	
Vcc	2.5V to 3.7V

for the typical bias case are presented in Table II. Programming current per cell was measured at 1.2μA. The programming time of the differential pair Flash cell using the external write driver was 10μs. Moreover, the programming time when performing simultaneous programming of 32 Flash cells via the SRAM-to-Flash transfer was also 10μs, corresponding to a normalized equivalent programming time of 0.31μs per cell. The read access time of the SRAM was measured at 20ns, as shown in the access time waveform in Fig. 7. The erase time was found to be 40ms. The operating range of voltages of the bit-line sharing memory array for programming operations were measured and are shown in Fig. 8.

Simultaneous data transfer of 32 bits between SRAM and Flash was also verified. Moreover, due to the low programming current of $\sim$1μA, it is possible to design an architecture that performs a simultaneous data transfer operation on a much larger number of cells, limited only by the charge pumping capability. For example, if 4096 cells are programmed simultaneously with a charge pumping circuit capable of 4mA, the equivalent normalized programming time per cell would be about 25ns, compared with approximately 500ns for conventional Flash.

The operating voltage range of the SRAM to Flash transfer and the Flash to SRAM transfer was also measured, and is given along with the Programming and Erase operating voltage ranges in Table III.

VI. CONCLUSIONS

The integration of SRAM and Flash cells on the same array using bit-line sharing has been described and demonstrated using an array fabricated in a 0.25μm CMOS logic process. The bit-line sharing architecture concept has the advantage of allowing parallel transfer of data between SRAM and Flash for fast operation. Through a low Flash programming current of 1μA per 1 bit, the application of the bit-line sharing architecture concept can enable programming of over 4096 Flash cells in parallel with a charge pumping circuit of $\sim$4mA. In the presented test chip architecture, parallel transfer of 32 bits between SRAM and Flash was successfully demonstrated. In addition, the architecture has the advantage of being able to operate as independent memories of SRAM and Flash, with SRAM read/write and Flash read/write operations. The Flash access time through SRAM was measured to be 20ns.

Area is saved by sharing the sense-amplifier, write driver, and column-decoding circuitry across both memory types. Furthermore, by using bit line sharing between Flash and SRAM, the chip area can be greatly reduced, in return reducing the production cost. Thus, potential uses lie in embedded memory applications and in system on chip (SoC) designs, with potential applications in a range of devices including cellular phones, digital cameras, and other platforms requiring multiple memory types and the ability to move data between them.

REFERENCES

[1] B. Prince, “Trends in scaled and nanotechnology memories,” in *Non-Volatile Memory Technology Symposium*, Nov. 2005, pp. 55–61.
[2] G. Muller, T. Happ, M. Kund, G. Y. Lee, N. Nagel, and R. Sezi, “Status and outlook of emerging nonvolatile memory technologies,” in *Int’l Eletron Devices Meeting Tech. Dig.*, Dec. 2004, pp. 567–570.
[3] G. T. Jeong, Y. N. Hwang, S. H. Lee, S. Y. Lee, K. C. Ryoo, J. H. Park, Y. J. Song, S. J. Ahn, C. W. Jeong, Y. T. Kim, H. Horii, Y. H. Ha, G. H. Koh, H. S. Jeong, and K. Kim, “Process technologies for the integration of high density phase change RAM,” in *IEEE Conf. on Integrated Circuit and Technology*, May 2005, pp. 19–22.
[4] K.-T. Chang, W.-M. Chen, C. Swift, J. M. Higman, W. M. Paulson, and K.-M. Chang, “A new SONOS memory using source-side injection for programming,” *IEEE Electron Devices Letters*, vol. 19, no. 7, pp. 253–255, July 1998.
[5] T. Ogura, N. Ogura, M. Kirihara, K. Park, Y. Baba, M. Sekine, and K. Shimeno, “Embedded twin MONOS Flash memories with 4ns and 15ns fast access time,” in *IEEE Symp. VLSI Circuits Dig.*, June 2003, pp. 207–210.
[6] K. H. Choi, “Non-volatile memory device,” U.S. Patent 6 965 145, Nov. 15, 2005.
[7] K. H. Choi, “Method of manufacturing self-aligned non-volatile memory device,” U.S. Patent 6 972 229, Dec. 6, 2005.