One string, two string, energy efficient magnetic RAM, the new thing

David Brown
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

***Abstract* —** For this project an algorithm was generated which takes two input strings (string1 and string2) and searches for the number of occurrences of string2 within string1 as well as the index of each occurrence. This algorithm was then implemented using MIPS assembly language. Energy consumption as a function of the algorithm's instruction set was analyzed in order to compare and contrast the energy efficiency of the read operations of four different memory cells. The analysis presented here finds the magnetic random access memory (STT/SHE-MRAM) pre-charge sense amplifier (PCSA) design has the lowest energy consumption for read instructions at 74375.60 fJ for the entire workload.

Keywords — Pre-charge sense amplifier (PSCA), merge sense amplifier (MSA), memory bit cells, magnetic random access memory (MRAM), read circuit, non-volatile memory (NVM), STT/SHE-MRAM, Process Variation (PV) resilient architecture, magnetic tunnel junction (MTJ) devices.

I. Introduction

A. Project Design

The string comparison algorithm starts by asking the user for a string, which is saved in buffer allocated for up to 10 characters. Once user input is saved into the registers, the program loops through a paragraph of text which has been hardcoded into the program, as well as the user input string, changing all uppercase letters to lowercase. Next, the length of the user input string is determined and saved. Finally, the hardcoded string is looped through counting the number of occurrences of the user input string as well as the index of each occurrence.

B. Test Cases

The following test cases were used as inputs to demonstrate the reliability of the program for a range of different inputs based on case sensitivity and location within an index of words. For test case 1, the user input with alternating text, "KnIgHt" was used to demonstrate ability of program to deal with case sensitivity. Test case 2 is the same string input, but without capitalization. The third test input is ucf, to demonstrate the locations of the index are accurate.

II. Memory Bit-Cells

Memory can be broken up into two categories; volatile and non-volatile memory. The basic difference is volatile memory requires power to maintain information in memory, while non-volatile does not. Common examples of volatile memory would be

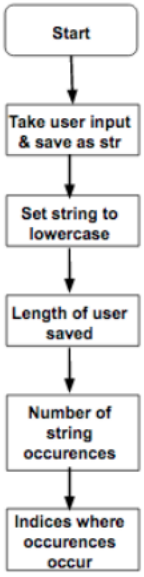

Fig.1: Flowchart of the assembly program.

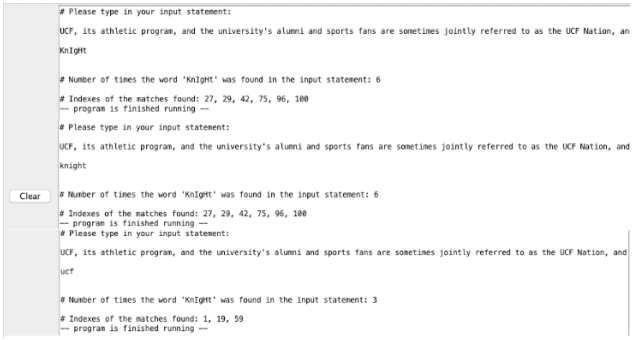

Fig.2: Sample outputs of assembly program.

static random access memory (SRAM) and dynamic random access memory (DRAM). Both memory modules store information in bits using memory cells.

Information stored within memory cells can be accessed by activating a "word-line" which points to some address in memory. A "bit-line" is connected to each memory cell pointed to by the word-line and dictates whether data should be read (bit-line LOW) or written (bit-line HIGH) to the memory cell.

Memory cell designs contain multiple bit lines with opposite values. This makes it easier to read the values in the cell, since the difference in voltages across the two cells can be read and interpreted as the information stored in the cell. One way to do this is using Comparators or Sense Amplifiers (SA), which are very sensitive to difference in voltage across two terminals, and amplify the difference into a digital signal. These are attractive for this purpose due to low power consumption (2).

Magnetoresistive random-access memory (MRAM) works by storing information in the magnetic field of each memory cell. A MRAM memory cell, called a magnetic tunnel junction (MTJ) device, contains three layers; a fixed layer, a free layer, and a magnesium oxide (MgO) layer sandwiched between the free and fixed layers. The fixed layer has a magnetic field with a constant orientation, while the free layer can vary in the direction of the magnetic field orientation. It faces either the same direction as the fixed layer (parallel orientation) or opposite the magnetic field (anti-parallel orientation). If the fields are anti-parallel then the resistance across the cell is high. If the fields are parallel in orientation, the resistance is low. Despite using a different mechanism for storing bits of information, these memory cells are similar to other designs in that they also contain word-, bit-, and source-lines.

There are two types of MRAM memory cell designs; spin transfer torque (STT-MRAM) and spin hall effect (SHE-MRAM). The general design of STT-MRAM was explained in the previous paragraph. SHE-MRAM is an alternative to STT-MRAM in that it generates a less stochastic write operation.

SHE-MRAM write operations work by adding a "write" current, which dictates the direction of the magnetic field orientation in the free layer. These designs are attractive alternatives to standard memory modules because they are non-volatile memory units (thus data is saved without power), have extremely low energy consumption when not active, and have large memory density per area (1).

However, a few problems exist with these memory modules. For instance, STT-MRAM consumes a large amount of energy when the module remains on after read or write operations. Attempts to lower this energy consumption have been successful by developing a bitwise operation that turned off the module after read/write operations (3).

Another non-trivial issue is Process Variation (PV) present in the fabrication process of MTJ devices. In an attempt to solve both of these problems, Soheil Salehi et al used two different STT/SHE-MRAM devices, one of which was energy efficient (Pre-charge sense amplifier; PCSA) and one which was resilient to process variation (Separated Pre-charge sense amplifier; SPCSA). They found that using their approach, which assigned the two modules based on the need for energy efficiency or resilience to Process Variation, they could drastically reduce power consumption as well as the bit-error rate from accessing MTJ memory cells (1).

Table II: Energy consumption for the assembly program using designs provided in [1-3].

Design	Energy Consumption For Each Bit-cell's Read Operation
MSA-PCSA [1]	74375.60 fJ
MSA-SPCSA [1]	75396.40 fJ
Comparator [2]	121906.60 fJ
STT-PSCA [3]	74579.76 fJ

III. Results and Discussion

The energy consumption of the aforementioned algorithm, when implemented in MIPS assembly code, was analyzed for the four separate memory cells (1,2,3). The instruction set can be broken down into five categories; ALU, Branch, Jump, Memory (Write and Read), and Other. The energy consumption per instruction for all instructions was constant with the exception of the Memory Read instruction, whose energy consumption varied across each memory cell. Energy consumption per instruction is as follows; ALU: 1fJ, Jump: 2fJ, Branch: 3fJ, Memory Write: 50fJ, Other: 5fJ. The variance in the energy consumption per Memory Read instruction for each memory cell can be found in Table 1.

Table I: Energy consumption for a single bit-cell read operation in the designs provided in [1-3].

Design	Energy Consumption For Each Bit-cell's Read Operation
MSA-PCSA [1]	0.35 fJ
MSA-SPCSA [1]	1.15 fJ
Comparator [2]	37.6 fJ
STT-PSCA [3]	0.51 fJ

The values for energy consumption per read instruction for all four memory cell designs were used to calculate total energy consumption of the designed string search algorithm. The total number of instructions for the string search algorithm was 7364. This total was subdivided into 3803 ALU instructions, 1651 Branch instructions, 599 Jump instructions, 1276 Memory instructions, and 35 Other instructions. Each of instruction total for each instruction category was multiplied by their respective energy consumption per instruction value. The result of each multiplication was then added to a total which represents the total energy consumption of the string search algorithm for each memory cell (Table 2).

IV. Conclusion

The energy consumption, per read instruction, of four different memory modules was analyzed. Based on the analysis done, it appears the MSA-PCSA design has the lowest energy consumption for read instructions at 74375.60 fJ for the entire workload. However, the STT-MRAM with a bitwise algorithm controlling power consumption, and STT/SHE-MRAM SPCSA designs are very close at 75396.40 fJ and 74579.76 fJ, respectively. Thus, it is important to keep in mind that while these designs consume more energy for read instructions, these designs might be superior to the MSA-PCSA design when compared by other metrics or instruction sets. It has been pointed out that MSA-PCSA is superior for low power applications, but MSA-SPCSA is superior in reliability (1). Therefore, it is important to implement these different memory cells depending on the conditions and requirements of the design.

References

[1] S. Salehi, N. Khoshavi, R. Zand, and R. F. DeMara, "Self-Organized Sub-bank SHE-MRAM-based LLC: an Energy-Efficient and Variation-Immune Read and Write Architecture," Integration, The VLSI Journal, accepted and in-press.

[2] HeungJun Jeon and Yong-Bin Kim. 2010. A low-offset high-speed double-tail dual-rail dynamic latched comparator. In Proceedings of the 20th symposium on Great lakes symposium on VLSI (GLSVLSI '10). ACM, New York, NY, USA, 45-48.

[3] R. Bishnoi, F. Oboril, M. Ebrahimi and M. B. Tahoori, "Self-Timed Read and Write Operations in STT-MRAM," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 24, no. 5, pp. 1783-1793, May 2016.