Different Technologies to Design Full based on Non-volatile condition

Houda Alaafya & Peirce Connell
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

***Abstract*— As technology advances, the demand for faster devices increase. Therefore, the search competition for new devices that can replace or compliment CMOS (Complementary Metal Oxide Semiconductors) technology has increased in the research industry with the intention of identifying different approaches to reduce leakage power increases. And this is where the magnetic tunnel junction (MTJ) comes. Since the introduction of the spin-based devices become popular, new designs of full adders have been discovered using different methods to solve the different issues such as reducing the leakage power increases. Integrating ADCs in the sensors due to the area of the analog circuits, the increase of the static energy consumption, and the decrease of the switching high energy, and the delay issues.**

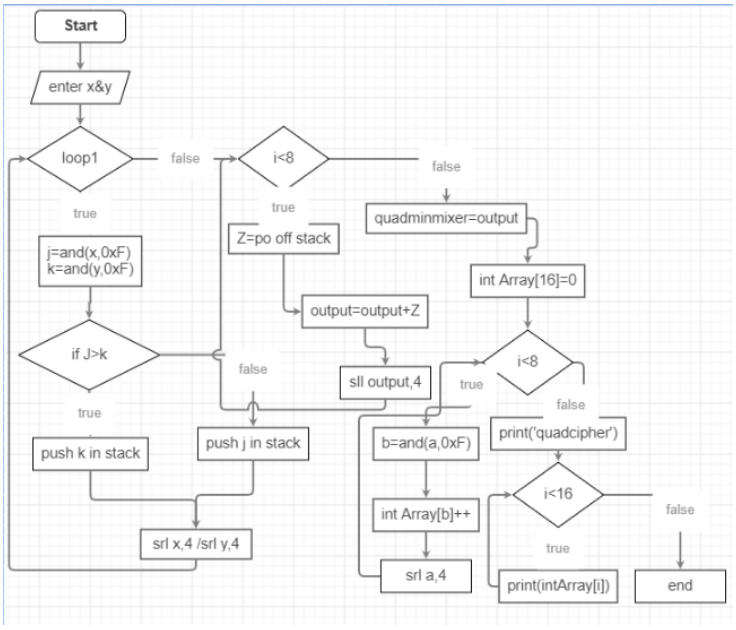
Keywords_ CMOS, MG, MTJ, non-Volatile, NVFA, NVM, logic memory, RTM, PCSA, DW, MA...

I. INTRODUCTION

A. Project Design

Our code first reads in both user inputs then calls the QuadMinMixer (x, y) subroutine using both the user inputs as arguments. The QuadMinMixer subroutine then iterates through the and compares the 32 bits of the integers in groups of four. It does this by masking a group of four bits in each iteration starting with the smallest four bits and ending with the largest four. Through each iteration of the look the subroutine compares the corresponding four bits of both numbers and pushes the minimum of the two onto the stack. After the subroutine has iterated through all eight groups of 4 bits the subroutine then pops each set off the stack and add them to the output number left shifting the output number between each pop to provide the requisite space. The subroutine then prints its output in hexadecimal and returns the output to the main function. The main function then iterates through the number returned from the subroutine in the same way the subroutine did, by masking sets of four bits at a time. However, in the main function once a set of four bits is masked the corresponding index of an array of integers is incremented in order to track its occurrence. This is done by left shifting the set of masked bits two places in order to multiply by four then adding that number to the address of the beginning of the integer array, loading the resultant address, incrementing it and then storing it again. The program then loops through the integer array printing the value stored at each index in decimal.

Our approach to designing the code for this project was modularity and reusability, we designed each section of the code in isolation before combining them together as one fully functioning program. This made testing our code much easier as it allowed us to quickly isolate what was causing our code not to work rather try to find it in one large code block. Our use of the stack was a solution to the problem of starting with the smallest bits when comparing in the subroutine, but needing to print the largest bits first. Because stacks operate on the principle of First In, Last Out pushing our comparisons to the stack was an elegant solution. Finally we designed our code with reusability in mind. Given both the subroutine and the main program involved a lot of the same actions being performed on integers we were able to reuse the sections of our code that dealt with iterating through the bits of the integers saving us time in the long run.



B. Test Cases

Three sets of inputs for our given function that we used for testing were as follows: 1. x = 1792801454, y = 2016082984 2. x = 2100000000, y = 154698 3. x = 123456789, y = 123456789. We chose the first set because it is the benchmark for instruction count and cache hit rate, so we felt this exemplified normal operation for our program. The second set was chosen because 2100000000 is the maximum expected input and we felt this would be a good stress test for our program. Finally, the third set was chosen because we wanted to make sure our program still functioned given the edge case of both user inputs being the same number.

```
1792801454
2016082984
QuadMinMixer = 0x682afa28
QuadBitCipher = 1000020201000200
-- program is finished running --
```

```
2100000000
154698
QuadMinMixer = 0x00025500
QuadBitCipher = 0000000000200105
-- program is finished running --
```

```
123456789
123456789
QuadMinMixer = 0x075bcd15
QuadBitCipher = 0011100010200011
-- program is finished running --
```

II. Full-Adder Circuit

As a non-volatile, near zero stability energy, and high-density device, MTJs overcomes the power and delay issues as well as its ability to be used as a CMOS alternative with more advantages. As the need of integrating the signal acquisition and processions and the rapid parallel data conversion in sensor nodes has increased, the demand for energy and area efficient Analog to digital converters (ADCs) increases. However, the design of such sensors in CMOS know many challenges such as the large area of analog circuits, the increase if static energy consumption, and the decrease of reliability cause by the high process. As a solution to these challenges. A framework for efficiency acquisition of digital signals that used spin-based devices was discovered. The architecture is called Spin Based Logic in Memory Analog Digital Converter (SLIM_ADC); it used the Spin Hall Effect drives Domain Wall Motion (SHE_DW) devices for a faster quantization of analog signals in a novel energy efficient fashion in addition to the realization of logic operations. The SLIM_ADC can reduce the energy consumption during the on/off without considering the backup storage. The MTJ devices can be used as different function such as OR gates, Majority Gates (MG) and AND gates. Combining 3 input MG will give a 1bit Full adder. See fig below.

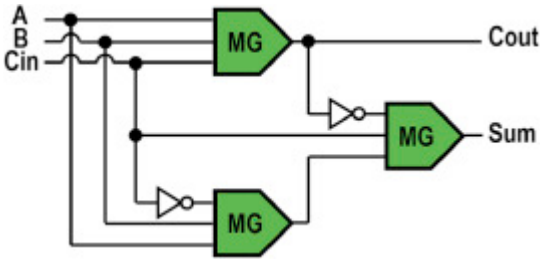


FIG 1: implementation of a 1-bit MG-FA circuit utilizing the SLIM-ADC devices. [1]

Using non- volatile, fast switching, and high-density devices such as MTJs m have solved the power and delay issues by making new designs and architectures. one of the designs is the key of knew components of a multibit full adder. The MTJ technique together with demultiplexing approach is used in a Non-Volatile Full Adder (NVFA) to reduce the area and power as well as improving the speed and sensing margin. The NVFA scheme can also be used in other types of NV- memory (NVM). By integrating the memories on the top of the logic circuits, NVM technologies have provided solution for the high power and interconnection delay issues. The NVM can be store the computing results in addition to recalling for different operations by not only shorting the length difference between the memory and logic but also by using a smaller number of transistors for a higher density. In this paper, the processors which is a multi-bit full adder will be analyzed. The FA will have the input and output data stored in perpendicular magnetic anisotropy (PMA), domain Wall (DW) and racetrack memory (RTM). Therefore, the circuit will include the new Logic in memory architecture based NVFA circuit using racetrack memory (RTM). The structure of the full adder would be as follower. The non-volatile storage advantage of the WTM is that a ferromagnetic strip is used to store ta sequence of binary bits. The current generated by the conducting wire below the strip create several domains that are called writers or write head. PMA MTJ function as a writer and a reader of the RTM. once the data is written, the train of domains moves along the strip synchronously via the Spin Torque Transfer (STT) effected by the steady DC current throw the strip. And to read the data from the strip, an MTJ reader placed at a different strip location sense the magnetic moment directions of the domains. The stored data in the ferromagnetic strips can be directly used in the operations. The convenient of separating the computing architecture, logic and memory is the performance of logic operations. Both the instruction and data should be read from the memory units for example cache and

main memories to the logic units. However, using full adders based on RTM have limitations. The first is the need to update (the large consummation of energy) the 10 input MTJ's in the SUM circuit, and the 6 ones in the CARRY circuit. The second is the serial write of the Complementary MTJ pair require a VDD or low DW velocity from the low to high resistance or vice versa which extend the switching period. And the third and last one issue is the small difference between the resistance in the SUM circuit branches which may produce a sensing result error. For the MJT being the basic element of the magnetic RAM (MRAM). The MRAM has nowadays been discussed in the industry. The 1bit MFAs based on the MTJ were considered to increase the standby power and the high dynamic power to move data between the computing core or buses and the memories. However, to build a MA, multiple 1bit MFAs are required otherwise multi bit NVFF (NV flip flops). In either case, the area of the circuit will be very large. the new 1bit MFA design without data switching is composed of two pre-charge sense amplifiers (PCSAs) used for data sensing, and MTJs as input data named A, B, and a CARRY C. the two opposite MTJs connect the branches to the PCSA. If the resistance of the left branch is less than the right one, then the output will be low for both SUM and CARRY otherwise if the left resistance is greater than the right resistance the output will produce a high logic value. The Boolean expression of the full adder can me expressed at the XOR of A, B, and C. the DW FA has many advantages first the increase of the area efficiency, the nonvolatile data (input and output) as they get stored in the DW shift registers.

III. Results and Discussion

In order to calculate the Energy consumption of the MIPS code, using the energy consumption per instruction found in the references [1, 2, and 3]. In order to access the Mips code dynamic instruction; Tool => Instructions statics.

- In order to find the energy consumption of the Mips code the following equation is used, Considering the following energy per instruction:

1) ALU = different ALUs for different tecnhologies see Table 1
2) Branch = 3 pJ
3) Jump = 2 pJ
4) Memory = 100 pJ
5) Other = 5 pJ

Energy= (Code ALU%*(ALU)) +(Code Jump%*2) +(Code Branch%*3) +(Code Memory%*100) +(Code Other%*5).

- Since the Mips instruction statistics of our code only consumes 50% ALU and 50% other. The energy consumed is shown in the second table.

Table I: Energy consumption for a single ALU Instruction in the designs provided in [1-3].

Design	Energy Consumption For Each ALU Instruction
[1] SLIM_ADC	0.6pJ
[2] NVM	6.3pJ
RTM-based [3]	1.67pJ
STT-based [3]	1.61pJ

- The different energy consumption results depending on different ALU technologies are shown in the Table II below.

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

Design	Total Energy Consumption
[1] SlIM_ADC	25.25pJ
[2] NVM	28.33pJ
RTM-based [3]	25.83pJ
STT-based [3]	25.79pJ

IV. Conclusion

This paper presents the different ways of designing a Full Adder using different technologies. The main purpose of this paper is to compare which technology overcomes the main three issues of a Full adder which are power consumption, the increase of static energy consumption, and the decrease reliability. The results show that using the SLIM_ADC, it consumes 0.2uW leakage power compared to 1uW of the CMOS. and since SLIM-ADC is used to implement thee (MG-FA) it is one of the technologies that can replace CMOS while building a full adder. The second technology is the NVFA, the Non-Volatile Full Adder is used in other non-volatile devices such as Non-Volatile Memories (NVMs). Comparing the MFA (magnetic full adder) with the NVFA, the NVFA reduces the power by 6 times and the area by 50%. Not forgetting that the speed by 10% and the sensing margin by 66%.

References

[1] S. Salehi and R. F. DeMara, "SLIM-ADC: Spin-based Logic-In-Memory Analog to Digital Converter Leveraging SHE-enabled Domain Wall Motion Devices," Microelectronics Journal, Vol. 81, pp. 137–143, November 2018.

[2] H. P. Trinh et al., “Magnetic adder based on racetrack memory,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 60, no. 6, pp. 1469–1477, Jun. 2013.

[3] K. Huang, R. Zhao and Y. Lian, "A Low Power and High Sensing Margin Non-Volatile Full Adder Using Racetrack Memory," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 62, no. 4, pp. 1109-1116, April 2015.

[4] Sverdlov, Viktor, et al. “CMOS-Compatible Spintronic Devices.” *Ieeexplore.ieee.org*, IEEE

[5] IEEE Transactions on Circuits and Systems I: Regular Papers (Volume: 62 , Issue: 7 , July 2015)