# Optimized energy consumption via circuit based Full-Adder Boolean logic

**Marco Bogani**
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

***Abstract*— With heat and energy consumption being the main detrimental factors, where even a hypothetical 10% cut on both can have positive economic and environmental results. As per this analysis, a Full-Adders uses Boolean logic-based circuit provides a reliable solution to this issue and overall beneficial results. MUX (multiplexer) for ADCs (Analog to Digital Converter) within FPGAs (Field Programmable Gate Arrays) provide considerable power improvements. In this paper, we will discuss the analogy between MUXs and Boolean gate arrays and ALUS used in full adders. While concentrating in energy expenditure, four designs for a single ALU instructions and the assembly program will be compared. The methodology of the program is shown in Fig. 1 providing its function and output. Total energy to comparable ALU can be seen in Table 2.**

***Keywords—ALU, Full-Adder, MUX, FPGA, ADC, NVFA, MTJ,MFA***

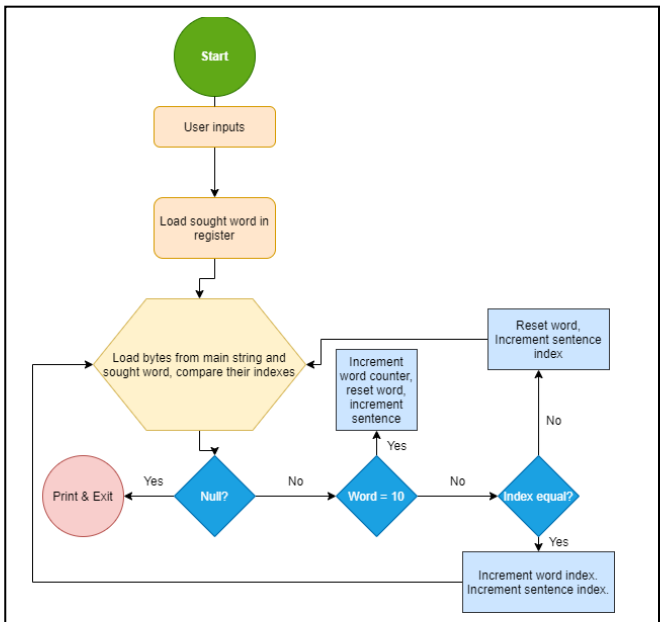
## I. INTRODUCTION

The program starts by taking two simple strings and temporarily storing them in temporary locations. An array is formed from one of the strings to act as comparison array for the other string. Then the program will compare both looking for similarity hits through loops. Once a similarity if found, the program records both its location and occurrence and stores it in different registers. Once similarities are not found the program will print the registers in charge of storing the results.

The assembly program used in this project takes input1 string from the user and stored into a register. An additional input2 string is taken and store at another register. The program should fin both the location index and occurrence of input2 in input1. This is done through an array formed from input1. The sought input2 will also be automatically formatted to the same format as input1 before being compared to the array of input1. The output will be both the number occurrences and indexes of input2 with input1's array.

### *A. Project Design*

The flowchart in fig. 1 gives a general idea of the code's function. In the .data section string stored and utilized to count the occurrence count of the user input. In one loop takes charge of loading bytes from both the specified word and the main




```
#Please type in the first word (up
KniGhts
#Please type in the second word (up
UcF
#Number of times the word 'KniGhts'
6
#Number of times the word 'UcF' was
2
#Please type in the first word (up  to
KNIGHTS
#Please type in the second word (up  to
UCF
#Number of times the word 'KNIGHTS' was
6
```

string, then compares the specified word's index with that of the main string. The main string's index goes through modifications to compare for both lower and upper case that to the sough word. If no matches occur in the indexes, the sought word is reset. But if all indexes in the sought word matches the main string index, a counter is updated. This keeps occurring and the sought word is reset continually until the loop reaches a null termination in the main string.

### *B. Test Cases*

As shown in Fig. 2 three input samples were chosen: KniGhts, UcF, and KNIGHTS. Regardless of the combination of both lower and upper-case words, the programs count for the word as a whole. Additionally, the counts outputs are shown in Fig. 2 as well.

## II. Full-Adder Circuit

As in SLIM-ADC circuits rely in Full-Adder logic to obtain their outputs. This is done through a set of MUX's along with four inputs, three current and one analog. In total the circuit has two MUX, although the MUX0 selects whether it goes into two different modes, either ADC or LIM (logic-in-memory) mode. These combinations of inputs and outputs are similar to that of a Full-Adder. The basis of a full adder is composed of 3 inputs and two outputs accordingly.

Similarly, a Magnetic Full-Adder (MFA) works just like conventional Full-Adder by using magnetic fields as means of bit switching. This type of adder has different advantages over the conventional adder. Area efficiency is greatly increased by up to 13-fold, Idle power efficiency is also increased due to its's non-volatile nature, and its high performing. Additionally, the Non-Volatile Full Adder (NVFA) with the demultiplexing technique and MTJs sharing scheme, the power of the proposed NVFA was greatly reduced. The sensing margin was also improved by removing the paralleled Magnetic Tunnel Junctions (MTJs) in each branch. Other, more performance driven methods are Gate Diffusion Input (GDI) based full adders, in which the three full adders are successfully realized using full swing gates with the significant improvement in their performance. Furthermore, simplicity driven adders like the Ternary Serial Adder (TSA) using carbon nanotubes (CNFETS) providing Multiple-valued logic results in chips with more density, less complexity and high-bandwidth data transfer. Lastly, the efficiency aimed Hybrid Full Adder presents a low voltage and high performance 1-bit full adder designed with an efficient internal logic structure that leads to have a reduced Power Delay Product (PDP).

## III. Results and Discussion

Calculation obtained through:

Energy = (ALU*Table1[ ])+(Jump*3pJ)+(Branch*4pJ)+
(Memory*100pJ)+(Other*5pJ)

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

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

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

| Design | Total Energy Consumption |
|---|---|
| [1] | 4.3E-7 pJ |
| [2] | 5.2E-7 pJ |
| RTM-based [3] | 4.5E-7 pJ |
| STT-based [3] | 4.4E-7 pJ |

## IV. Conclusion

It is notable that design 1 provides the least energy consumption upon being used with the assembly program. Additionally, a trend observation pertained the steady energy consumption of the assembly program with the ALU consumption per design, thus invoking a linear relationship. This shows that different implementation of a Full Adder can impact overall efficiency of a circuit. Radical changes shown in MFAs by mainly focusing in area efficiency, or the RTM based NVFA making overall improvements on both area and speed, and lastly energy saving SLIM-ADC. This comes to show that implementing different ways of performing a specific task can greatly affect the surrounding variables while producing the same output

## 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, W. S. Zhao, J. O. Klein, Y. Zhang, D. Ravelsona, and C. Chappert, "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] Mohan Shoba "GDI based full adders for energy efficient arithmetic applications" in Engineering Science and Technology, an International Journal, vol 19, Issue 1, pp. 485-496, Jun 2015

[5] Mohammad Hossein Moaiyeri "An efficient ternary serial adder based on carbon nanotube FETs" in Engineering Science and Technology, an International Journal, vol 19, Issue 1, pp. 271-278, May 2015

[6] Pankaj Kumar "Low voltage high performance hybrid full adder" in Engineering Science and Technology, an International Journal, vol 19, Issue 1, pp. 559-565, July 2015