Look Up Tables Advancing FPGAs in Many Ways

Mark Escott
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

***Abstract*—The objective of this report is to take a deeper look at Look up Tables and how they can differ from each other to advance our modern day FPGA. The different types of look up tables can improve the FPGA by taking up less space, decreasing power consumption, speed etc. We look into four different types of look up tables that can help us with the advancement of our FPGAs. If you want a large advancement in speed, you would look toward the MTJ-LUT for your fastest. As far as the total energy consumption goes, we find later that the RRAM-based LUT gives us the largest difference, cutting the closest (MTJ) LUT energy consumption by over 2-fold.**

***Keywords*—Spin Hall Effect (SHE), Magnetic Tunnel Junctions (MTJs), Field Programmable Gate Arrays (FPGAs), Resistive Random Access Memory (RRAM), Magnetic RAM (MRAM), Non-Volatile Memory (NVM). Clockless-LUT (C-LUT)**

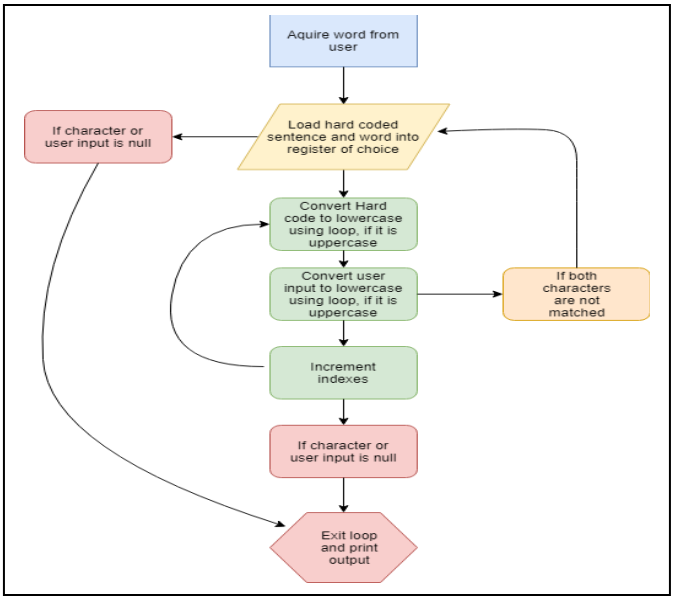
I. Introduction

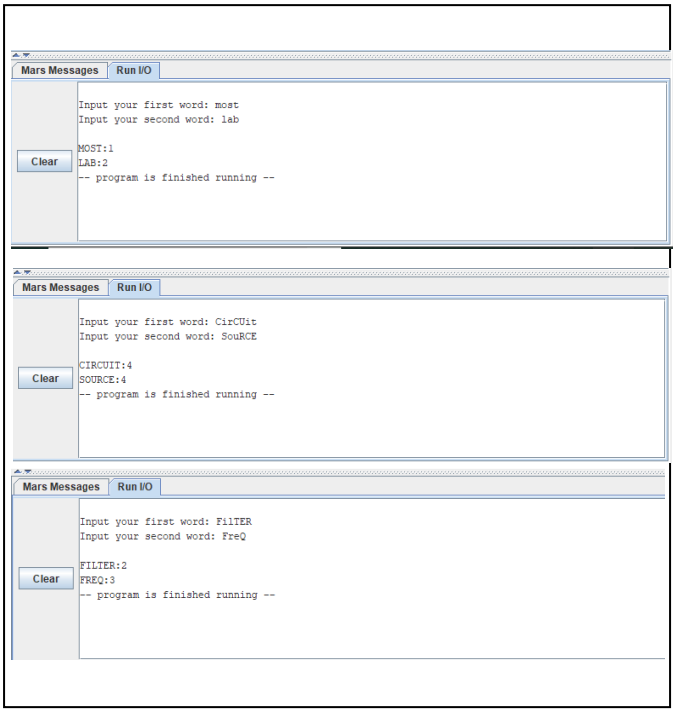
A. Project Design

For the assembly code portion of this project, We are tasked with taking 2 words from the user, and counting how many times they occur within a hard coded statement. It will not matter how the words are input by the user, if its lower case or upper case. Within the code there will be loops included so that the case will not matter. The word that the user wants will be converted so that the code can correctly read it and see that it is a part of the current word in the register. The direction taken in this code is that we will be loading in the word by the user and the character of the statement to see what matches until there is a space. When each character matches after a space in the hard coded statement (while using loops with jumps to change the case of the character) until there is another space, then the counter increases for that word. This is repeated with both codes and flow chart provided for the first and second word.

B. Test Cases

Using 3 ways of testing, I believe it is best to use 3 completely different hard coded sentences with different words every time for the best results. Doing this will provide the best coverage to finding out where bugs are and where there might be mistakes while also showing the best cases possible. Below, are Test cases being used that are not the case given in the project:





Case 1: Most experiments are designed to be one week experiments. The pre-lab with the calculated and simulated results, is to be completed prior to coming to the lab to do the measurements.

Case 2: A DC circuit is an electrical circuit that consists of any combination of constant voltage sources, constant current sources and resistors. The voltages and currents in this circuit are invariant with time, in other words, constant. A DC circuit is usually powered by a DC voltage source or a DC current source.

Case 3: Resonant circuits form the basis for filters that have better performance than first order (RL, RC) filters in passing desired signal or rejecting undesired signals that are relatively close in frequency. The resonance frequency is defined as the frequency at which the impedance of the circuit is purely real, that is, with zero reactance.

II. Look-Up Table Circuit

Look up Tables (LUTs) are used for combinational logic. LUTs are one of the main components of modern day Field Programable Gate Arrays (FPGAs). As seen in reports that FPGAs are a very useful tool for modern day computing and logic, it is also known that they come with their fair share of limitations.

FPGAs have been researched as a device that can be very promising and can effectively increase reliability in case of process-voltage-temperature variation. They can also be very promising in the way that we use modern day computing using logic.

One problem that we see with this is that there are SRAM FPGAs that run into some limitations such as high static power, volatility and low logic density. We know now that using LUTs in a different way, we can bridge the limitations that SRAM FPGAs give us.

We then can see another type of LUT which are spin-based LUTs that use clocks. They are said to offer qualities to help the FPGAs by offering non-volatility, a near zero static power, a high endurance and also high integration density.

FPGAs can suffer in more than just the SRAM FPGAs. In general, they suffer by the fact that more than 40% of the area on the FPGA is taken from the configuration memory bits. Since memory seems to be the problem, a way this has been tackled is by using NVM or non-volatile memory or more specifically the RRAM (Resistive Random Access Memory)-based LUT to take the place of the SRAM.

There is another weakness in current FPGAs that lies with the (CMOS) Complementary metal oxide semiconductor, including long power-on/reboot latency of programmable circuits, loss of data during unexpected power supply interruptions and high leakage currents, resulting in the exponential increase of 'idle' or standby power consumption [3]. These deficiencies can be cause by any of the previously listed problems, register latches, CMOS, SRAM. A new LUT-based FPGA is presented successfully is 2006, the Magnetic RAM (MRAM). This discovery increased the resistance between two states, which greatly improves sensing stability and speed of an FPGA.

III. Results and Discussion

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

Design	Energy Consumption For Each ALU Instruction
STT C-LUT [1]	170.88 fJ
SHE C-LUT [1]	184.08 fJ
RRAM LUT[2]	33.7 fJ
MTJ LUT[3]	134 fJ

Assembly Code

*ALU = 13757 * (Table I)*
*Branch =11154 * 3fJ*
*Jump = 6368 * 2fJ*
*Memory =2573 * 200fJ*
*Other =9953 * 5fJ*

As shown above, when finding the total energy consumption, we multiply the number of instruction of a specific type, by the energy needed to complete that instruction. We then add up the products to get the total energy consumed. Below are the results of each LUT.

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

Design	Total Energy Consumption
STT C-LUT [1]	2,961,359.16 fJ
SHE C-LUT [1]	3,142,951.56 fJ
RRAM LUT[2]	1,074,173.9 fJ
MTJ LUT[3]	2,454,001 fJ

IV. Conclusion

LUTs have been an increasingly interesting topic because of how much I did not understand before this. They are still very complicating and difficult to understand but after reading the reports and seeing what is helping make the FPGAs better has helped tremendously.

I learned the LUTs are a part of the entire FPGA, also that the memory of the FPGA takes up a lot of room. The FPGA can be and is being increasingly more efficient by using the previously stated LUTs from the reports. I also learned of many other deficiencies that come into play with FPGAs like back conductors causing long power-on/reboot latency etc. and other problems with various based LUTs like the SRAM with high static power, volatility and low logic density. With the total energy calculated above, we can see that the RRAM, which cuts down on our ALU energy significantly, makes a huge difference when it comes to total energy consumption.

Increasing how our LUTs and logic works can be very detrimental to how efficient technologies can be in the future.

References

[1] Soheil Salehi, Ramtin Zand, Ronald F. DeMara. 2019. Clockless Spin-based Look-Up Tables with Wide Read Margin. In Great Lakes Symposium on VLSI 2019 (GLSVLSI '19), May 9–11, 2019, Tysons Corner, VA, USA. ACM, Washington, DC, USA, 4 pages.

[2] B. Khaleghi and H. Asadi, "A Resistive RAM-Based FPGA Architecture Equipped With Efficient Programming Circuitry," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 65, no. 7, pp. 2196-2209, July 2018.

[3] Zhao, Weisheng, Eric Belhaire, Claude Chappert, François Jacquet, and Pascale Mazoyer. "New non‐volatile logic based on spin‐MTJ." physica status solidi (a) 205, no. 6 (2008): 1373-1377.