Implementation of Low Dynamic Power & Area Optimised 16-Bit Reversible ALU

Utkarsh Mishra[1]

***Abstract*— Today's Computer chips are facing a severe problem with power dissipation in addition to that heat generation. Reversible logic reduces power consumption as zero-energy computation has inspired it. It has various areas envisioned for its applications; they are low power CMOS design, Quantum & Optical computing, Nano-Technology, DSP, etc. Using this Reversible logic different combinational circuits are designed, like Decoders & Multiplexers. This proposed design architecture uses less dynamic power, total power and area as compared with a simple ALU.This architecture has been modelled with Verilog in Cadence tools. The reason behind reduced dynamic power is second law of thermodynamics which states that for a reversible process the change in entropy is zero which transforms to the heat exchange is zero. Thereby using reversible gates makes our computations reversible and thus lowering our power consumption.**

I. INTRODUCTION

As the capacity and density of an integrated circuit gradually increase in recent time's power dissipation has become a critical factor in design. Such VLSI circuits need more power so that the power consumption will be high. Thus, it can dissipate more heat causes damage to the reliability ICs. By using Reversible Decoder for designing Combinational circuits power consumption is reduced to an optimum when compared to a conventional decoder based combinational circuits. Reversible Circuits have an equal number of inputs and outputs and has a bijective mapping between them. Hence input vectors can always be uniquely derived from output vectors and vice versa. ALU is the basic building block of CPU. An ALU is a combinational circuit that can have one or more inputs and only one output. ALU output is dependent only on inputs applied at that instant as a function of time and not on past conditions. Basic ALU consists of two operands and one input to select the operation. ALU implemented in this paper does four operations addition, subtraction, multiplication and shifting.

A. Reversible Logics

The concept of reversible logic gates is used for reducing power consumption and loss of data. Reversible computing is the application principle of recycling to the computing. It is because input can be reconstructed from the output. This logic uses the reversible gates which have the same number of inputs and outputs. Some of the cost metrics like garbage outputs, number of gates, Quantum cost, constant outputs are used to estimate the performance of reversible circuits. A Reversible circuit design can be modelled as a Sequence of discrete time slices and depth is a summation of total time slices.

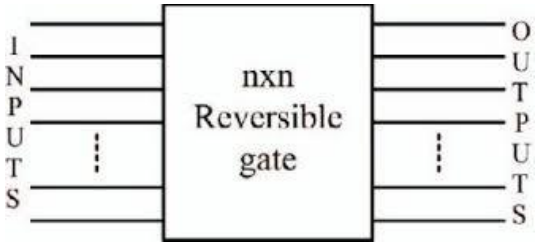

Fig. 1. nxn Reversible Gate

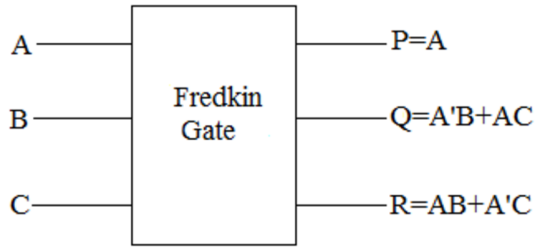

Fig. 2. Example - A Fredkin Gate

Fredkin gates with a combination of inputs serves as not , and, or, xor gates. This property has been used throughout the architecture.

II. DESIGN

A. Decoder (2x4)

Firstly, we have implemented a 2x4 decoder. An 2x4 decoder is implemented as shown in below diagram(see fig 3). Using this we implemented an 3x8 decoder. Below is the RTL of a 2x4 decoder implemented using only fredkin gates as reversible gates.

B. 1-bit Full Adder using Decoder

We then use OR's of min-terms generated through outputs of a 3x8 decoder to calculate the sum and carry. So this how our full adder gets implemented(see fig 4).
$S = \sum(m1, m2, m4, m7)$
$C = \sum(m3, m5, m6, m7)$

C. 16-Bit Adder

16 Bit adder is implemented using full adders from above.Input is are two 16-bit size operands and output is 17 bit sum, where last bit indicates the output carry.

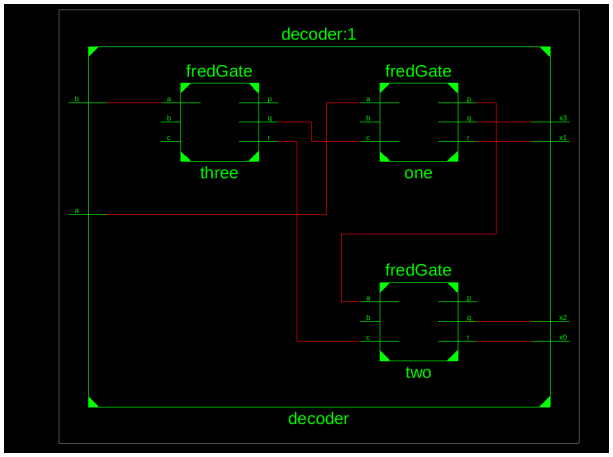

Fig. 3. Decoder (2x4)

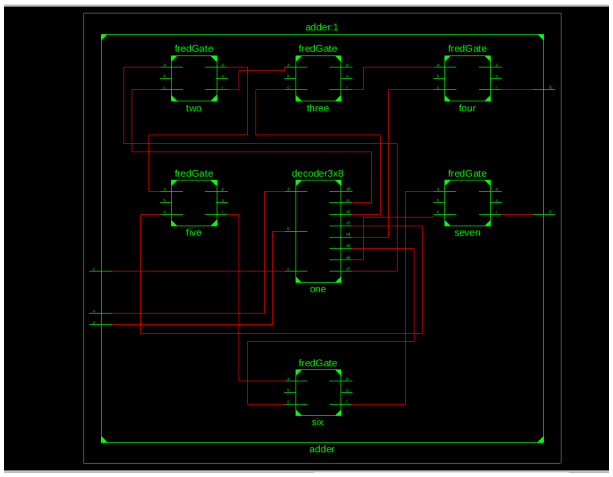

Fig. 4. 1-bit Full Adder using Decoder

D. 16-Bit Subtractor

16 Bit subtractor can be implemented by using the binary compliment of one operand (the one to be subtracted) and adding it to other operand with input carry as 1. The output Sum is the 16-Bit subtraction of the complimented operand from the other operand. Assumption here used is that the number whose compliment is taken is the smaller number.

E. 16-Bit Multiplier

16-Bit Multiplier is implemented using an algorithm. In this algorithm a variable iterates through all 16 bits of an operand1 and stores their product of a bit with operand2 in a buffer1. Then shifting the all bits of buffer1 to add the new product and storing the lost bit in another buffer2. The final answer 32-Bit size , obtained by concatenating both the buffers.

F. 16-Bit Binary Shifter

16-Bit is implemented in a way where one operand is the input on which the shifting operation is to be done and the other operand indicates the value by which the bits are to be shifted. The outputs are two binary strings, left shifted and right shifted, both by an amount operand2.

III. RESULTS

A. Working of ALU

Below is the Timing diagram of an ALU and Reversible ALU(see fig 5).

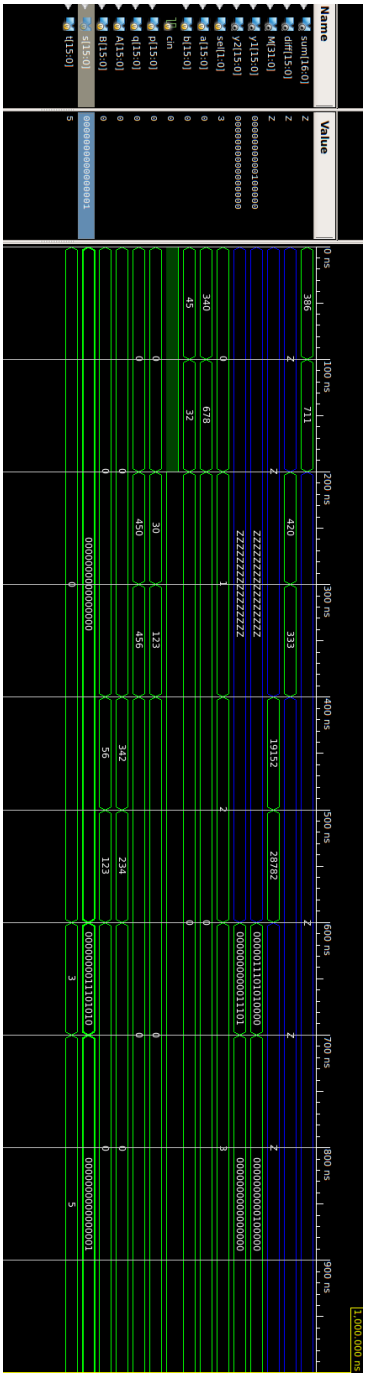

Fig. 5. Timing Diagram of ALU and Reversible ALU

This ALU does four operations depending and outputs the desired depending upon the input of select lines. The

four operations are addition, subtraction, multiplication, and bitwise binary shifting.

B. Power and Area comparison

Refer figure.6 - Genus Power comparison
Refer figure.7 - Innovus Power comparison
Refer figure.8 - Genus Dynamic Power Comparison
Refer figure.9 - Genus Area Comparison
Refer figure.10 - Innovus Area Comparison
Refer figure.11 - Delay reports Comparison

Fig. 6. Genus Power Comparison

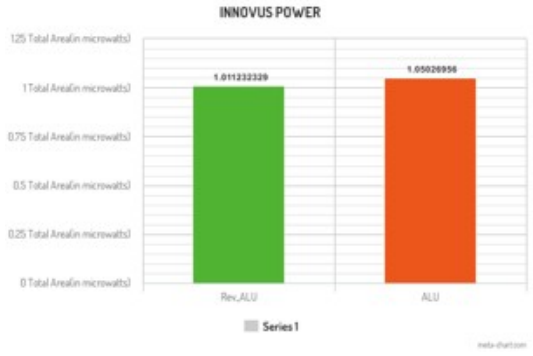

Fig. 7. Innovus Power Comparison

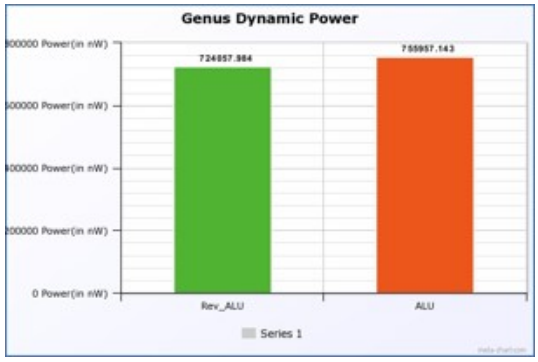

Fig. 8. Genus Dynamic Power Comparison

	Genus Area (in cell units)	Genus Power (in microwatts)	Innovus Power (milliwatts)	Innovus Area (micrometer square)
ALU	10528	814.690707	1.05026956	10491.3909
Rev ALU	9968	777.123397	1.011232329	9966.1023

Fig. 9. Genus Area Comparison

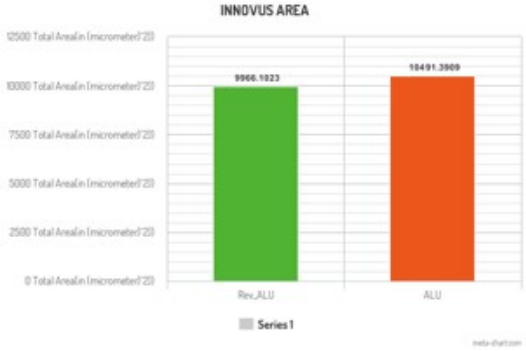

Fig. 10. Innovus Area Comparison

IV. CONCLUSION

Hence, implementing the above architecture is a very good example of applied thermodynamics. Reversible logics and gates can help us to decrease the power consumption marginally without compromising with precision and results of the machine.

V. ACKNOWLEDGEMENT

The notable contribution of professor Azeemuddin Syed and the very helpful Teaching assistants namely Mayank Awasthi and Kunal WAdhwani cannot be expressed in words. Their guidance was priceless and helped me complete the project fully and on time with full understanding of the basic concepts to complex implementation of the project.

VI. REFERENCES

A. Cadence Tools

B. Xilinx Tools

C. International Journal of Applied Engineering Research ISSN - "An FPGA Implementation of Low Dynamic Power Area Optimized 32-Bit Reversible ALU " - By G. Vamsi Krishna, G. Srinivasa Rao, Y. Amar Babu

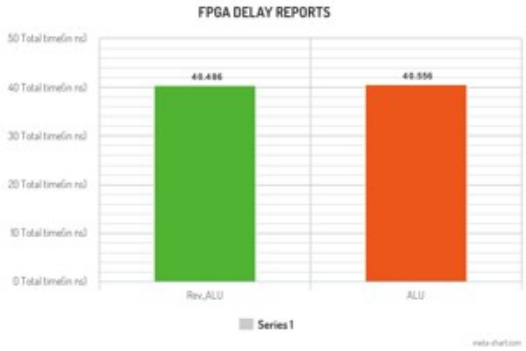

Fig. 11. Delay reports Comparison