International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Design of MAC-Unit with High Speed and Low Power Using Network Stimulation Software

Chinthapalli Baba Shankar[a*], R. Vamshi Vardhan Goud[b], Nader Babo Dukhun Albaire[c]

[a] *Babashankar64@gmail.com, Department of ECE (Digital Systems & Computer Electronics), MTech, JNTUH College of Engineering, Hyderabad 500085, INDIA*

[b]*vamshivardhan217@gmail.com, Department of ECE (Systems & Signal Processing), MTech, JNTUH College of Engineering, Hyderabad 500085, INDIA*

[c]*Eng.naderbabo@gmail.com, Department of ECE (Digital Systems & Computer Electronics), MTech,, JNTUH College of Engineering, Hyderabad 500085, INDIA*

A B S T R A C T

Popularity of MAC-unit is increasing day by day with the advancement of new technology .In this fast updating world speed and power place a promenient role. Thus a fast and efficient MAC-unit is required in various technologies and gadgets. In this paper, we desiged a MAC-unit with more speed and more efficient in terms of power usage then the previous existing MAC-units using VLSI design language. Here we used VLSI language to design a MAC-unit with modified internal circuits.

Keywords: MAC-unit, Design, VLSI

1. Introduction

In this paper Low Power Compressor Based Multiplier Accumulator is designed using VLSI Software. The increasing demand for portable systems and the need to limit power consumption and heat dissipation in very high density chips have led to rapid developments in low power design during recent times. The battery lifetime is additionally a priority on the general power consumption of the portable system. Hence, reducing the facility dissipation of integrated circuits through design improvements may be a major challenge in portable systems design. The need for low power design is additionally a problem in high performance digital systems, like microprocessors, digital signal processors (DSPs), and other applications. In digital VLSI circuits, computation is that the critical part and it decides the facility consumption and operating speed of the designs. For computations, arithmetic circuits involve adders and multipliers; which are the foremost copiously used components. Digital signal processors performing filtering, convolution, etc, relies on the efficient implementation of these adders, multiplier, and MAC units. Low power compressor architecture is proposed during this brief to scale back the facility consumption of the MAC architecture since the presence of more compressors. The impact of the circuit design level or the info path optimizations is addressed at the MAC level for DSP applications. Additionally, the carrier addition involved in the multiplier and accumulate stages are merged together to increase the number of compressors within the MAC architectures .

2. MAC UNIT

The Multiplier-Accumulator (MAC) operation is the key operation not only in DSP applications but also in multimedia information processing and various other applications. MAC unit consist of multiplier, adder and register/accumulator.The MAC Unit take inputs from the memory location such as RAM and given to the multiplier block. This is very useful in 8 bit digital signal processor. The inputs which is being fed from the memory location is 8

Corresponding Author E mail : babashankar64@gmail.com

bit. When the input is given to the multiplier it starts computing value for the given 8 bit input and hence the output will be 16 bits. The multiplier output is given as the input to adder which performs addition.

Then, the output of adder is given to the accumulator register. The accumulator used is designed with Parallel in Parallel out (PIPO) Type. Because the accumulator produces output in parallel form and also the bits are huge. PIPO register is used where the input bits are given in parallel and output is taken in parallel. The output of the accumulator register is taken out or fed back as one of the input to the CSA.

3. Block Diagram

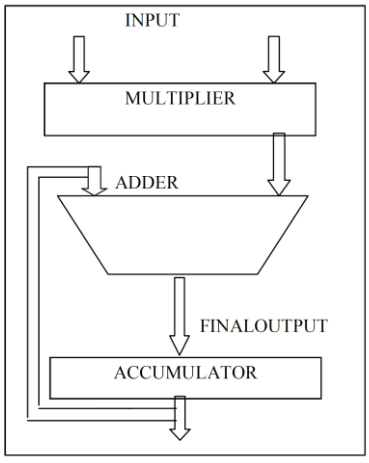

Fig 1. Block Diagram of MAC architecture

4. Multiplier

Multiplier takes input and performs multiplication and gives the output to adder. The multiplication logic involved in multiplier is discussed below.

Multiplication Logic :

Considering an example of 8 bit multiplication in which 8 bit input is X7 X6 X5 X4 X3 X2 X1 X0 and multiplier isY7 Y6 Y5 Y4 Y3 Y2 Y1 Y0. The multiplication process is shown in fig.2. There is the requirement of 64 AND logics. First Y0 is multiplied with X7 X6 X5 X4 X3 X2 X1 X0 and results X0Y0, XIY0, X2Y0, X3Y0, X4Y0, X5Y0, X6Y0 and X7Y0. After it Y1 is multiplied with X7 X6 X5 X4 X3 X2 X1 X0 and results X0Y1, XIY1, X2Y1, X3Y1, X4Y1, X5Y1, X6Y1 and X7Y1. Similarly all multiplications are taken place. In each step there is one binary shifting the resultant logic.

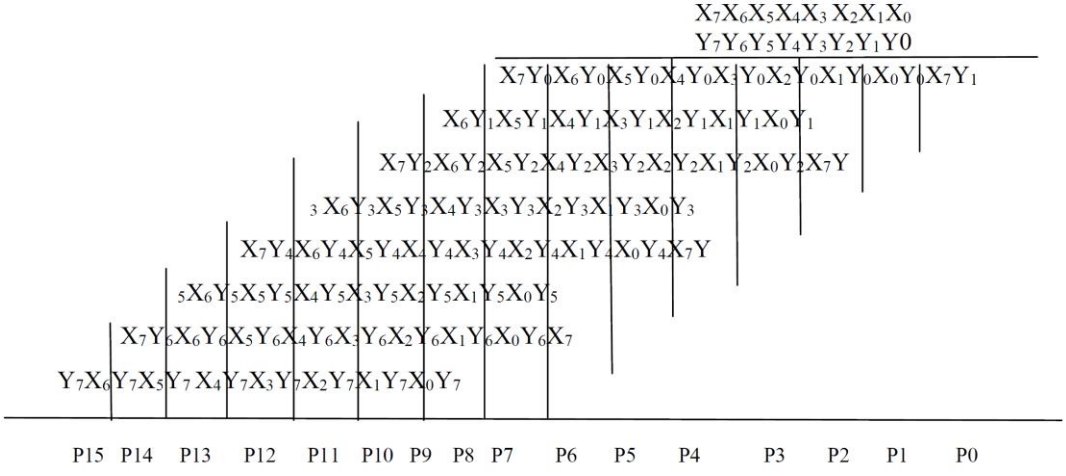

Fig 2: Multiplication Of 8bits

5. Adder

An **adder** is a digital circuit that performs addition of numbers. In many computers and other kinds of processors adders are used in the arithmetic logic units or ALU. They are also used in other parts of the processor, where they are used to calculate addresses, table indices, increment and decrement operators, and similar operations.

Although adders can be constructed for many number representations, such as binary-coded decimal or excess-3, the most common adders operate on binary numbers. In cases where two's complement or ones' complement is being used to represent negative numbers, it is trivial to modify an adder into an adder–subtractor. Other signed number representations require more logic around the basic adder.

Table1:Addition operation

Inputs		Outputs	
A	**B**	**C**	**S**
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	0

6. Accumulator

An *accumulator* is a register where arithmetic and logic operations are performed and their results stored (*accumulated*). Usually one operand was already stored in the accumulator while the other came from memory, the immediate value field of an instruction, or if the computer has them another register.

Many early computers had only one accumulator and a small number of index registers. On such machines the accumulator register itself is the *ALU*, it has all the special circuits inside itself to perform those operations and modify its own contents. The flip-flops that store the data in an accumulator are usually *JK-flip-flops*, which simplify the design of those special circuits.

Later when computers began to use multiple *general purpose registers* the concept of the accumulator fell out of favor. On such machines *one separate ALU was shared* with all the registers, which were reduced to just a small bank of very *high speed RAM* with no special circuits like accumulators usually have.

7. Compressor Cell

Compressors are the digital circuits which have the capability to add five/six/seven bits at a time and hence called as column compressors. A typical five input compressor is illustrated in this brief. It takes 4 regular inputs and 1 intermediate carry-in input and generates 1 sum bit, 1 carry-out bit and another intermediate carry bit. Intermediate carry bits are the carry-in and carry-outs (called as horizontal carry propagation) from previous and to next stage compressors. Carry-out (also called as vertical carry) bit is final carry generated along with the sum bit. Since compressors forms the basic and critical components for multipliers and large-input adders, several compressors architectures were developed in the past to address several constraints. Some of the compressor architectures described in the past are shown in Fig.

Compressor architecture shown in Fig. 3 is built using the full-adders. This architecture has only two cells and will have minimum interconnects but each of the cell needs to generate the sum and carry path and one of the path is dependent on the other. This requires larger drive strength to drive the chain of compressors and hence the power consumption will be higher. The higher drive strength will significantly have the reduced delay. Fig. 4 shows the compressor architecture built using lesser fan-in gates. Logic implementation with lesser fan-in gates leads to more number of interconnects which has significant impact on glitch power & delay. In lower technological nodes the interconnect power is dominant than the gate power, hence the architecture of [9] leads to high power consumption. Fig.4.

Fig. 5 shows the proposed compressor architecture. The proposed compressor architecture is built with larger fan-in gates and also using separate logics for sum and carry paths. In the sum path four 2 input XOR cells are replaced by two 3 input XOR cells and in the carry path two 2 input AND cells & one 2 input OR cells are replaced by one 6 input AND-OR (AO222) logic cell. Larger fan-in gates covers large part of the logics and helps in minimizing the

number of gates required for implementation. Lesser gates lead to smaller area and minimum interconnect delays. Thus the proposed compressor architecture helps in reducing the power consumption.

Thus the proposed compressor architecture enables new features like design specific/constraint specific architectures and allows utilizing for low power applications. Optimizations provided in the proposed architectures are,

- Minimum interconnect in sum-path reduces the interconnect delay and associated glitches
- Reduced power consumption with minimum interconnects
- Independent carry logic to reduce the horizontal carry delay.

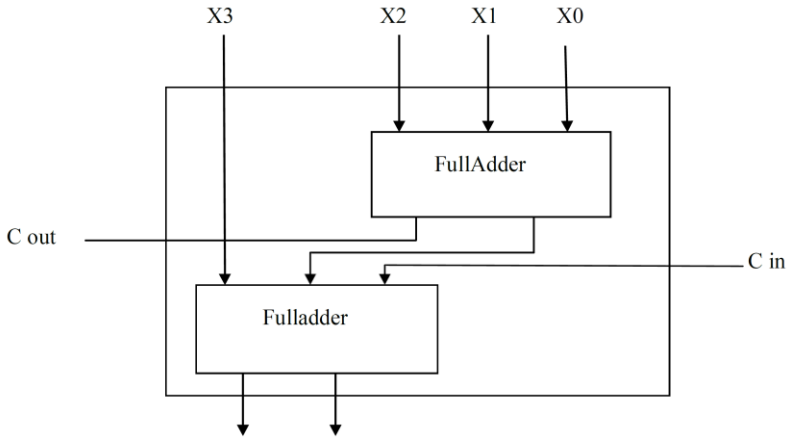

Fig3: Full Adder compressor cell

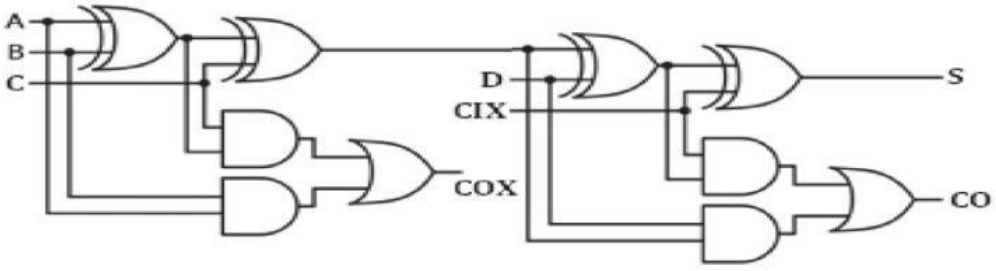

Fig4: David Harris Compressor cell

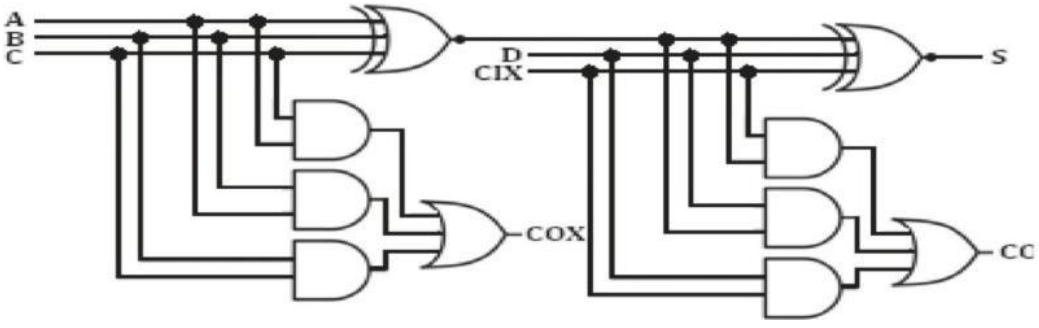

Fig5:Used compressor cell

8. Power Comparison

Power can be calculated by connecting to FPGA and can be calculated by usingTotal power= Dynamic power + Static power**.**

Table 2: Comparison Table of CompreSSOrS

Design	Using Full Adder	Compressor
Area	1086.12	1083.24
Delay	2.546	2.874
Total Power	76.782	73.324

As more number of compressors are required in the MAC architecture, the proposed MAC architecture requires less number LUTs and it constitutes to lesser interconnects and resulted in the reduced delay against the existing MAC architecture with compressor architecture. Since the numbers of LUTs are higher in existing MAC architecture and as per the relation larger the area; higher will be power consumption, the power consumption of the existing MAC architecture is higher than the proposed MAC architecture. More number of interconnects also contributes to power consumption.

9. Block Diagram

The output block diagram of Low power compressor based mac architecture is shown below. Inputs given to the MAC are a7-a0 and b7 –b0, clock , reset and maco(15:0) is obtained as output. Internal multiplication operation is performed and then added and accumulated and output is obtained as maco(15-0)

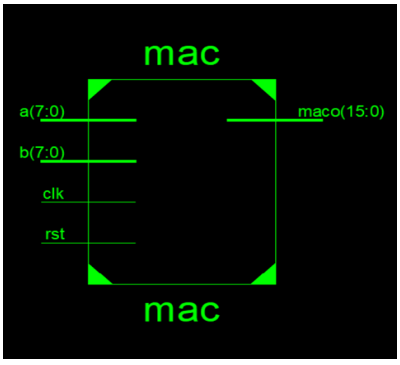

Fig 6. Block diagram of MAC architecture

10. Internal Diagram

The internal schematic diagram of low power compressor based MAC multiplier accumulator is shown below. MAC has two blocks multiplier and accumulate. Multiplier has a7-a0,b7-b 8-bit as inputs and o(15:0) is obtained as output. This output is given as input to accumulator. Clock and reset are also given as input to accumulator. Maco(15:0) is obtained as output.

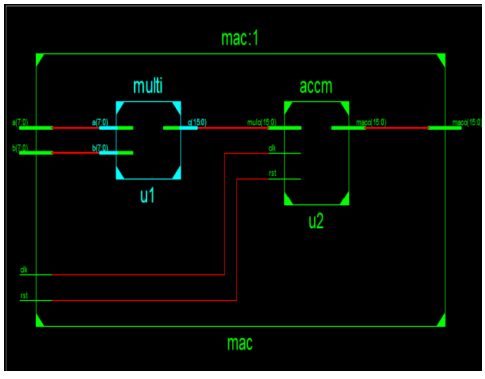

Fig 7. Internal diagram of MAC architecture

11. RTL Schematic Diagram

The multiplier has many internal blocks which perform multiplication process. Each block includes compressor cell. The green blocks in the figure indicate the compressor block. Left most inner green line indicates multiplication operation and adder is also present in the multiplier block only. The right green line is accumulator.

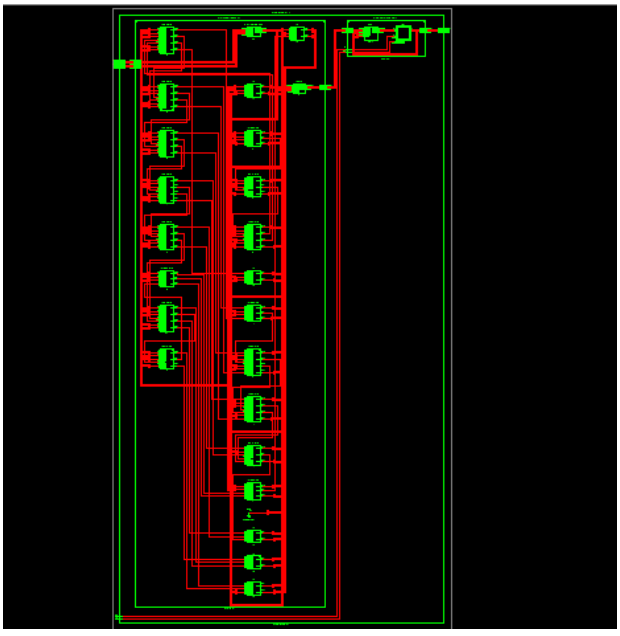

Fig 8. RTL schematic diagram

12. Compressor Cell

The internal block present in multiplier has the compressor cell. This compressor cell performs the multiplication operation and all such obtained outputs are given as inputs to accumulator.

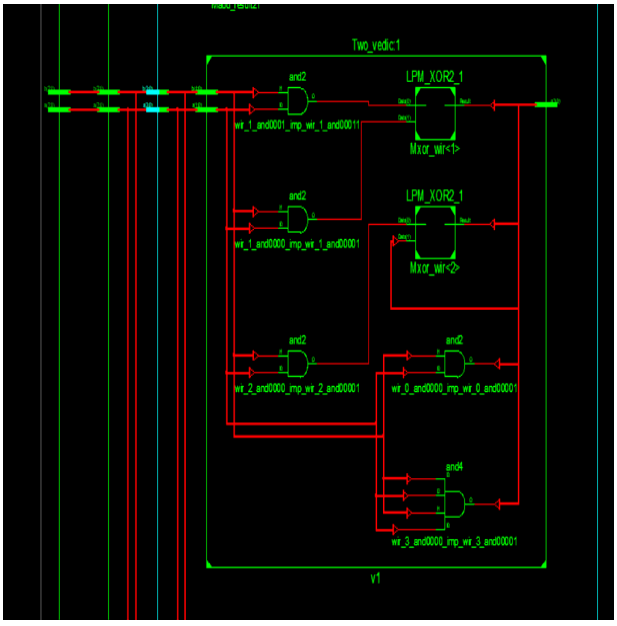

Fig 9. Compressor cell

13. Simulation Output Waveforms

The output waveforms are as below shown. The first green graph indicates the mac output. Second line indicates the clock input. Third line indicates the reset. The fourth and fifth line indicates the inputs given to the mac. Whenever the reset is high what ever the input is, the output is zero. When reset is low, at positive clock edges, multiplication operation is performed and the output is added with previous output and accumulated and that accumulated output is obtained as mac output.

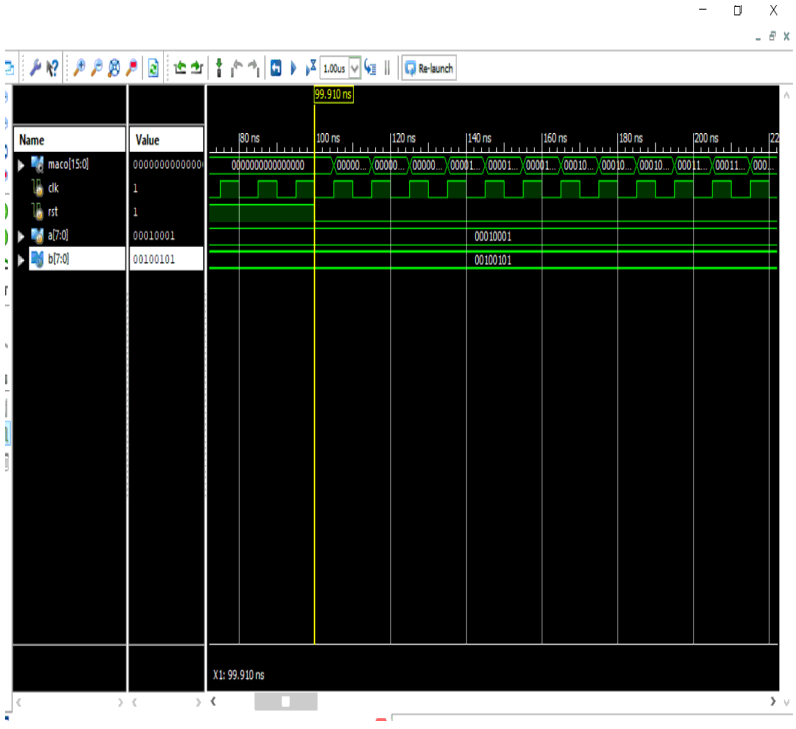

Fig 10. Simulation output waveforms

14. Conclusion

Low power compressor based 8-bit MAC architecture is designed using multiplier, adder and accumulator. Multiplication is performed using multiplication logic and the output is added using adder and then it is accumulated simultaneously.

High Performance 8 bit MAC Unit is designed and implemented using compressor , multiplier and Adder and accumulator. When compared to all other MAC Units which are developed earlier using different combinations of multipliers and adders, the designed compressor offers High Performance with Less Delay, Less Power Dissipation which further increases the overall speed of MAC Unit. This MAC Unit is designed using Verilog - HDL and Synthesized using Xilinx 14.3 ISE.

REFFRENCE

- Vaijyanath Kunchigi, Linganagouda Kulkarni, Subhash Kulkarni International Journal of Scientific and Research Publications, Volume 3, Issue 2, February 2013 ISSN 2250-3153
- Ramalatha, M.Dayalan, K D Dharani, P Priya, and S Deoborah, High Speed Energy Efficient ALU designusing Vedic multiplication techniques, International Conference on Advances in Computational Toolsfor Engineering Applications, 2009. ACTEA '09.pp.600-3, July 15-17, 2009.
- Jagadguru Swami Sri Bharati Krishna Trithaji Maharaja, "Vedic Mathematics", Motilal Banarsidas Publishers Pvt. Ltd Delhi, 2009. International Journal of Digital Applications & Contemporary Research (Volume 5, Issue 3, October 2016).