



# Power Efficient Compressor based MAC Architecture for DSP Applications

Mothukuri Ramalakshmi Bala<sup>1</sup>, Chodiseti L S S PavanKumar<sup>2</sup>

PG Scholar<sup>1</sup>, Assistant Professor<sup>2</sup>

Department of ECE

AKRG College of Engineering & Technology, Nallajerla, A.P, India

## Abstract:

DSP operations are very important part of engineering as well as medical discipline. For the designing of DSP operations Multiplication play an important role to perform signal processing operations. Multiplier is one of the critical components in the area of digital signal processing and hearing aids. In this paper, efficient hardware architecture of MAC using a modified Wallace tree multiplier is proposed. The proposed MAC uses multiplier with novel compressor designs and adders as primitive building blocks for efficient application. Further, the Verilog-HDL coding of 32 bit MAC architecture and their FPGA implementation by Xilinx ISE 14.4 Synthesis Tool on Virtex7 kit have been done. The proposed compressor and adder based architecture used to be applied to MAC unit and in comparison to the previous design MAC unit and verified that the proposed architecture have reduce in terms of area, delay and power.

**Keywords:** Adders, Compressor, Data path DSP, Low Power VLSI, Multiply Accumulate

## I. INTRODUCTION

A DSP processor is designed to support fast execution of the repetitive, numerically intensive computations characteristic of digital signal processing algorithms. The most often cited of these features is the ability to perform a multiply-accumulate operation (often called a "MAC") in a single instruction cycle. A single-cycle MAC operation is extremely useful in algorithms that involve computing a vector dot product, such as digital filters. Such algorithms are very common in DSP applications. To achieve a single-cycle MAC, all DSP processors include a multiplier and accumulator as central elements of their data-paths [3-4]. The second feature shared by DSP processors is the ability to complete several accesses to memory in a single instruction cycle. This allows the processor to fetch an instruction while simultaneously fetching operands for the instruction, and/or storing the result of the previous instruction to memory. Typically, multiple memory accesses in a single cycle are possible only under restricted circumstances. To allow numeric processing to proceed quickly, DSP processors incorporate one or more dedicated address generation units. The address generation units operate in parallel with the execution of arithmetic instructions, forming the addresses required for data memory accesses. The address generation units typically support addressing modes tailored to DSP applications. Because many DSP algorithms involve performing repetitive computations, most DSPs provide hardware support for efficient looping. Often, a special loop or repeat instruction is provided which allows the programmer to implement for next loop without expanding any instruction cycles for updating and testing the loop counter and branching to the top of the loop. Finally, to allow low-cost, high performance input and output, many DSPs incorporate one more serial or parallel I/O interfaces, and specialized I/O handling mechanisms such as low-overhead interrupts or DMA. The major concern of portable gadgets is the battery life, which influences the real-time processing applications and their dynamic range of input signals for additive features. It is the high time to explore the challenging

criteria of these emerging low power, low area and high performance digital signal processing chips [1]. An efficient compressor architecture is proposed in this paper is, to scale back the area, delay and power consumption of the MAC architecture due to the fact that the presence of extra quantity of compressors. They have an impact on of the circuit design stage or the data path optimizations is addressed at the MAC degree for DSP functions. In MAC, additionally the carry propagate addition concerned in multiplier and accumulate stages are merged to multiplier and accumulate stages of compressors and adders within the MAC architectures. Designs had been illustrated in FPGA domains as per the standard design methodology.

## II. LITERATURE SURVEY

Saravanan&Madheswaran (2010) analyzed low power high performance Multiply and Accumulate (MAC) unit with Hybrid Encoded Reduced Transition Activity Technique (HERTAT) equipped multiplier and low power 0.13µm adder. The developed low power MAC unit is verified for image processing systems exploiting in significant bits in pixels values and the similarity of neighboring pixels in video streams. The proposed technique reduces dynamic power consumption by analyzing the bit patterns in the input three, and applies the proposed encoding technique, otherwise can make use of Booth technique. The proposed adder cell used in the MAC block consumes less power than the other previous adder techniques. This high performance low power MAC can be used in image processing. Jaina et al (2011) discussed that Real-time signal processing requires high speed and high throughput Multiplier-Accumulator (MAC) unit that consumes low power, which is always a key to achieve a high performance digital signal processing system. In this paper, design of MAC unit is proposed. The multiplier used inside the MAC unit is based on the Sutra "UrdhvaTiryagbhyam" (Vertically and Cross wise) which is one of the Sutras of Vedic mathematics. Vedic mathematics is mainly based on sixteen Sutras and was rediscovered in early twentieth century. In

ancient India, this Sutra was traditionally used for decimal number multiplications within less time. The same concept is applied for multiplication of binary numbers to make it useful in the digital hardware. Here, the coding is done in VHDL and synthesis is done in Xilinx ISE series. The combinational delay obtained after synthesis is compared with the performance of the "Modified Booth Wallace Multiplier" and "High speed Vedic multiplier" presented by Ramesh Pushpangadam. They proposed Vedic multiplier seems to have better performance. Elguibaly (2000) presented a dependence graph (DG) to visualize and describe a merged multiply-accumulate (MAC) hardware that is based on the modified Booth algorithm (MBA). The carry-save technique is used in the Booth encoder, the Booth multiplier, and the accumulator sections to ensure the fastest possible implementation. The DG applies to any MAC data word size and allows designing multiplier structures that are regular and have minimal delay, sign-bit extensions and data path width. Using the DG, a fast pipelined implementation is proposed, in which an accurate delay model for deep submicron CMOS technology is used. The delay model describes multi-level gate delays, taking into account input ramp and output loading. Based on the delay model, the proposed pipelined parallel MAC design is three times faster than other parallel MAC schemes that are based on the MBA. The speedup resulted from merging accumulate and the multiply operations and the wide use of carry-save techniques.

### III. PROPOSED SYSTEM

#### A. Compressor

Compressors are the digital circuits which have the potential to add 3/5/6/7 bits at a time and for this reason referred to as column compressors. A traditional five input compressor is illustrated on this temporary. It takes 4 regular inputs and 1 intermediate carry-in input and generates 1sum bit, 1 carry-out bit and one more intermediate carry bit. Intermediate carry-bits are the carry-in and carry--outs (referred to as horizontal carry propagation) from previous and to subsequent stage compressors carry-out (also called as vertical carry) bit is final carry generated along with the sum bit. Seeing that compressors varieties the basic and vital add-ons for multipliers and large-input adders, a number of compressors architectures were developed in the past to tackle several constraints. High speed multipliers use 3-2 and 4-2 lower the latency of partial product reduction part. Compressors are used to minimize delay and area which leads to increase the performance of the overall system. Compressors are generally designed by XOR-XNOR gates and multiplexers the existing compressors architecture described up to now are proven in Fig. 1 and Fig. 2 [8, 9].

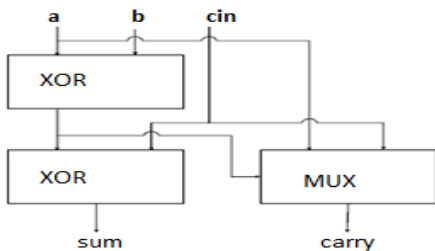


Figure.1. CONVENTIONAL 3:2 COMPRESSOR

In fig 1 A 3-2 compressor has three inputs a, b and  $c_{in}$  and generates two Outputs they are sum and the carry bits. The sum output is generated by the second XOR and carry output is generated by the multiplexer (MUX). Fig. 2 shows the compressor architecture developed utilizing lesser fan-in gates. Common sense implementation with lesser fan-in gates results

in more number of interconnects which has big effect on glitch power & delay. In lower technological nodes the interconnect power is dominant than the gate power, for that reason the architecture of [9] results in excessive power consumption.

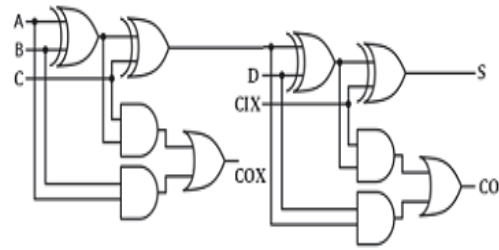


Figure. 2. david harris compressor cell [9]

Fig. 3 suggests the proposed compressor structure. The proposed compressor architecture is developed with higher fan-in gates and in addition utilizing separate logics for sum and carry paths. In the sum path four 2 input XOR cells are changed via two three input XOR cells and within the carry path two 2input AND cells & one 2 input OR cells are changed through one 6 input AND-OR (AO222) common logic cell. Larger fan-in gates covers enormous part of the logics and helps in minimizing the quantity of gates required for implementation. Lesser gates lead to smaller area and minimum interconnect delays. Thus the proposed compressor structure helps in lowering the power consumption.

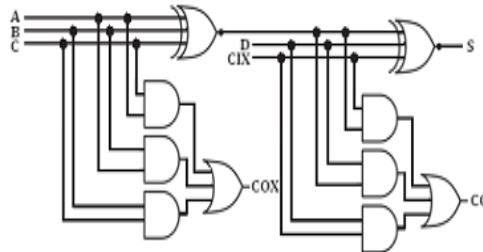


Figure.3. Proposed Compressor Cell

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

#### B. Multiply-Accumulate Unit

MAC is the basic and most frequently used component in DSP to perform filtering, convolution and etc to accelerate the FIR or FFT computations in communications [2]. Regular MAC unit contain multiplier, adders and registers as shown in Fig. 4, where the previous output of the MAC unit is added with the multiplier output and accumulated.

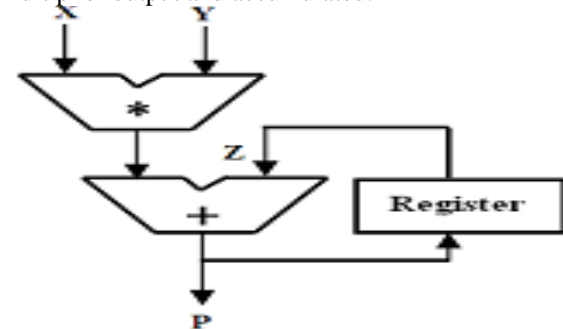


Figure.4. Regular MAC Architecture

Multipliers are implemented in three stages namely:

- a) Partial product generation,

- b) Partial product reduction and
- c) Carry propagate addition.

Regular architectures utilize the half and full adders in the partial product stages, but due to its performance limitation compressor cells were utilized. Some of the past architecture's reduced the number of reduction steps in the partial product reduction stage by introducing Wallace tree in the partial product generation stage, to reduce overall delay [3 - 5]. Use of compressors in the multiplier will shrink the quantity of gates for implementation which in turn reduces the number of interconnects. This outcome in decreased interconnect extend and system faults associated with-it, yielding efficient design. Thus the effective multiplier will strengthen the efficiency the MAC unit. For instance the usage of proposed efficient compressors and adder structure improves the area, delay and power effectively and suits for DSP applications. To illustrate the effect of compressors and adder architecture a MAC unit structure which comprises extra number of compressors is chosen from [2]. In [2], author has used the 4:2 compressor and 3:2 compressor and half adders in multipliers in the partial product reduction and in accumulation stage of the MAC unit, the place the carry propagate stage of the multiplier is merged with the input of accumulate add stage. Fig. 5 indicates the cutting-edge MAC architecture.

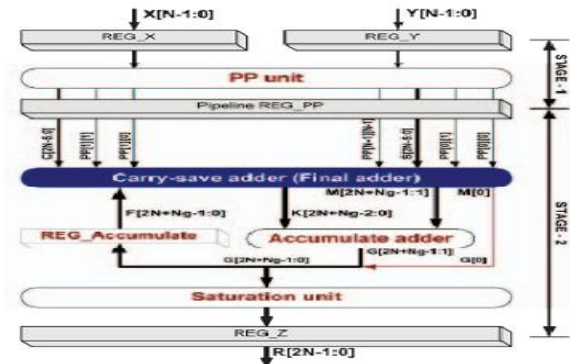


Figure. 5: State of The Art MAC Architecture [2]

IV. RESULTS AND ANALYSIS

Both the regular and proposed architectures at the compressor and MAC unit level were designed and verified. Results of the compressors and MAC units were benchmarked as per the standard design methodology for FPGA domain. The circuit level design optimization was also illustrated in the FPGA design and the synthesis results are tabulated in Table 1. In FPGA domain the designs were targeted to Virtex 7 family. In FPGA domain the logics are mapped to up tables (LUTs). Table 1 shows that the proposed architecture has better results than the existing architectures in FPGA domain.

Table.1. Comparisons of Proposed MAC with Existed MACs in terms of delay, power & area

8 bit MAC Design	DELAY (ns)	AREA	POWER (mw)
MAC using full adder based compressor	9.728	Slices: 237 LUTs :230 FFs:32	Tp: 597.18 Sp: 547.35 Dp: 49.83
MAC using 4:2 conventional compressor	8.858	Slices: 182 LUTs :181 FFs:32	Tp: 596.13 Sp: 546.84 Dp: 49.29
Proposed	5.913	Slices: 32 LUTs :149 FFs:32	Tp: 159.92 Sp: 142.92 Dp: 17.00

Table.2. Comparisons between Proposed 8 bit MAC and 32 bit MACs in terms of delay, power & area

PROPOSED DESIGN	DELAY (ns)	AREA	POWER (mw)
8 bit design	5.913	Slices: 32 LUTs :149 FFs:32	Tp: 159.92 Sp: 142.92 Dp: 17.00
32 bit design	23.138	Slices: 32 LUTs :149 FFs:32	Tp: 170.00 Sp: 143.00 Dp: 27.00

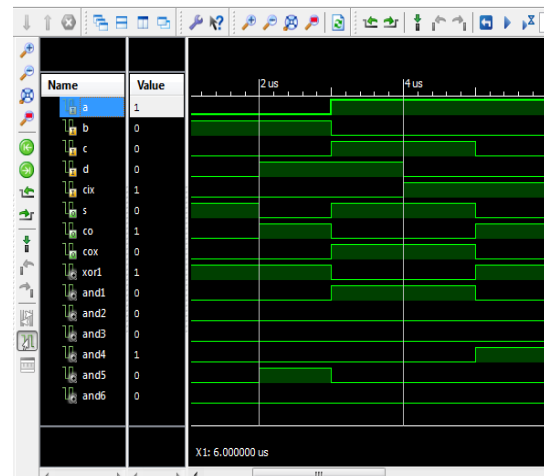


Figure.6. Proposed 4:2 Compressor Output

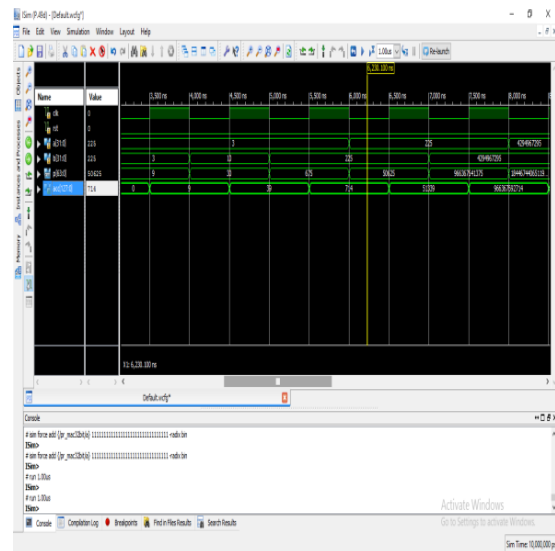


Figure.7. MAC output

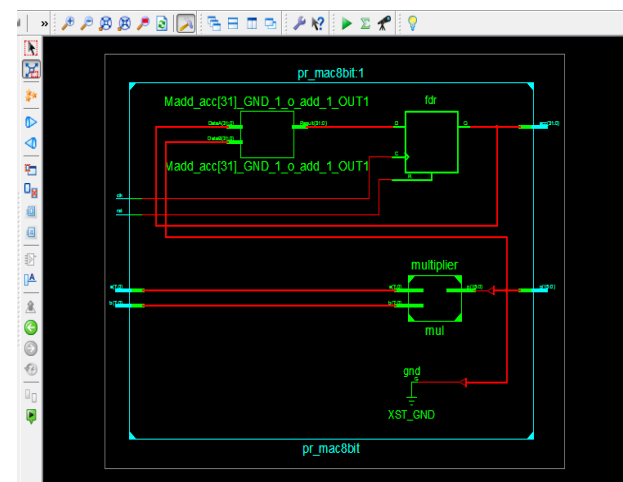


Figure. 8. MAC RTL Schematic

## V. CONCLUSION

Design and domain specific an efficient compressor and adders based MAC architecture has been demonstrated in this work. Thus we propose a new high speed, low power and area efficient MAC architectures which will be an improvement over the existing architecture by replacing conventional 4:2 compressor with proposed 4:2 compressor. The proposed architectures have yielded better efficient results in terms of area, delay and power in the FPGA domain.

## VI. REFERENCES

- [1]. Chang, Chip-Hong, Jiangmin Gu, and . "Ultra-low-voltage low-power CMOS 4-2 and 5-2 compressors for fast arithmetic circuits." *Circuits and Systems I: Regular Papers, IEEE Transactions on* 51.10(2004): 1985-1997.
- [2]. Tung Thanh Hoang; Sjalander, M.; Larsson-Edefors, P., "A High-Speed, Energy-Efficient Two-Cycle Multiply Accumulate (MAC) Architecture and Its Application to a Double-Throughput MAC Unit," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol.57, no.12, pp. 3073,3081, Dec. 2010.
- [3]. Chen Ping-hua; Zhao Juan, "High-speed Parallel 32×32-bit Multiplier Using a Radix-16 Booth Encoder," *Intelligent Information Technology Application Workshops, 2009. IITAW '09. Third International Symposium on*, vol., no., pp.406,409, 21-22 Nov. 2009
- [4]. Kiwon Choi; Minkyu Song, "Design of a high performance 32×32-bit multiplier with a novel sign select Booth encoder," *Circuits and Systems, 2001. ISCAS 2001. The 2001 IEEE International Symposium on*, vol.2, no., pp. 701,704 vol. 2, 6-9 May 2001.
- [5]. Rajput, R.P.; Swamy, M.N.S., "High Speed Modified Booth Encoder Multiplier for Signed and Unsigned Numbers," *Computer Modelling and Simulation (UKSim), 2012 UKSim 14th International Conference on*, vol., no., pp.649,654, 28-30 March 2012.
- [6]. Yangbo Wu; Weijiang Zhang; Jianping Hu, "Adiabatic 4-2 compressors for low-power multiplier," *Circuits and Systems, 2005. 48th Midwest Symposium on*, vol., no., pp.1473,1476 Vol. 2, 7-10 Aug. 2005.
- [7]. Jaina, D.; Sethi, K.; Panda, R., "Vedic Mathematics Based Multiply Accumulate Unit," *Computational Intelligence and Communication Networks (CICN), 2011 International Conference on*, vol., no., pp.754,757, 7-9 Oct. 2011.
- [8]. Aliparast, Peiman, Ziaadin D. Koozehkanani, and Farhad Nazari. "An Ultra High Speed Digital 4-2 Compressor in 65-nm CMOS." *International Journal of Computer Theory & Engineering* 5.4 (2013).
- [9]. N. Weste and David Harris, "CMOS VLSI Design- A Circuits & System Perspective", Pearson Education, 2008.
- [10]. Chandra Mohan U, "Low Power Area Efficient Digital Counters", *Proceedings of the 7th VLSI Design and Test Workshops, VDAT, August 2003*