Energy Efficient Comparator Design

Meenu Shrivastava[1] and Shweta Agrawal[2]
[1]Research scholar,[2]Assistant Professor,
Electronics and Comm. dept., SRCEM Banmore, Morena, India

***Abstract*- The energy efficient computations is the requirement for all portable battery operated devices. Along with the arithmetic operation, comparison is a frequency used operation with the computation. This paper explores the existing magnitude comparator design techniques and proposes a new binary comparator that provides significant reduction in the power and area. The effectiveness of the proposed technique over the existing is evaluated by coding in Verilog, synthesized and simulated on Xilinx tool chain and design metrics are evaluated. The simulation results on FPGA show that the proposed comparator provides more than 19.3% reduction in power over the best-known comparator.**

Keywords: **Comparator, High speed integrated circuits, VLSI, Low-power design, FPGA.**

I. INTRODUCTION

The rapid development in the modern very large scale integration (VLSI) technology is allowing us to integrate billions of transistors on the same chip. Although, this development in the integrated circuit design technology made it possible to implement more and more functionality on chip with each new technology, it brings several challenges to the VLSI designer such as maintaining area, power and delay within the desirable limits. Further huge transistor on the same chip further increases the complexity in terms of testing; process and other variability issues may become more severe. This integration of huge functionality is imposing several challenges to the VLSI designer as increasing functionality increasing failure probability. As energy consumption is the serious issue for these chips, an energy efficient design is the focus of research in the modern VLSI designs.

In addition to the basic arithmetic operations, comparison of two numbers is the prime operation performed in the most processing units for performing functionalities such as instruction decoding, flag generation etc. [1]. In order to perform comparison efficiently, there is the demand of high performance low power comparator. The performance of the comparator significantly affects the overall performance of these processing units [2]. Significant efforts have been given to improve area, power and delay parameters of the comparator at different level of abstraction [3-6]. The existing architectures of the magnitude comparator are not power and performance efficient [3], thus, demanding novel comparator architecture that provides highly energy efficient comparison of two numbers. This paper presents an energy efficient comparator for different signal processing application.

The rest of the paper is organized as follows: Section II presents the work done to achieve efficient comparator and further analyses these designs. Section III details proposed low power comparator architecture whereas its effectiveness using via simulation is given in Section IV. Finally Section IV concludes the paper.

II. COMPARATORS ARCHITECTURES

The section explores different comparator architectures in details.

2.1 Comparator Basics

The magnitude comparator exhibits two inputs and three outputs [3]. The comparator compares two inputs A and B and provides output as equal (Eq), greater (G) or small (S) given by the following equations. For example, a 2-bit comparator compares two 2-bit binary numbers A and B, and gives three outputs. Let input A and B have bits $A_1A_0$and B_1B respectively. The logical expression for the 2-bit comparators output G, S and Eq are given by the equations 1-3 below.

$$Eq = (A_1 \odot B_1).(A_0 \odot B_0)$$
$$G = A_1\overline{B_1} + (A_1 \odot B_1)A_0\overline{B_0}$$
$$S = \overline{A_1}B_1 + (A_1 \odot B_1)\overline{A_0}B_0$$

The logical diagram for the G, S and Eq are shown in Figure 1.

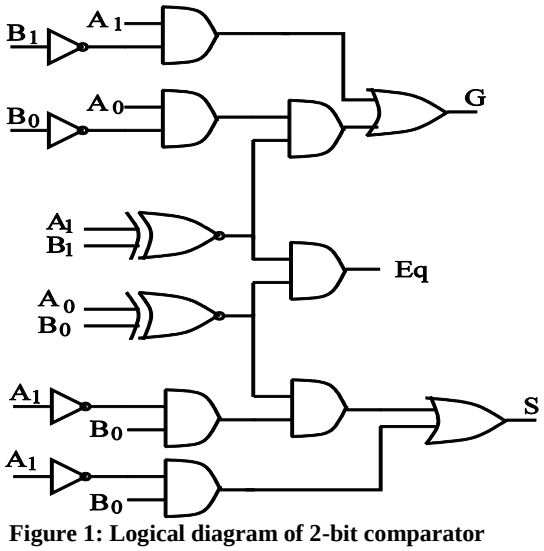

Figure 1: Logical diagram of 2-bit comparator

It is observed that as the input bit-width increases the complexity of the designs increases significantly. Therefore direct implementation of the higher bit-width comparator is costly in terms of area, power and delay as increasing complexity increases the power and delay. Hence, 4-bit comparator is used to design higher bit-

width comparator. The direct implementation of large bit-width comparator is very complex and costly [7-10]. Therefore, we can implement extensive bit-width comparator using small bit-width comparator.

2.2 Priority Based Comparator (PBC)

Priority based comparator [13] has three stages to compute output as is shown in Figure 2. First stage identify 1's in each input that may cause the number may be greater over the other number. The second step identify most significant in each number. The third step identifies which number have 1's at more significant position over the other.

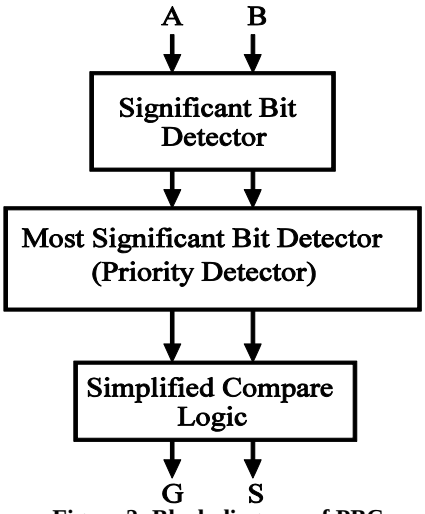

Figure 2: Block diagram of PBC

2.3 Look-ahead Comparator (LAC)

The LAC [13] is based on the concept of look-ahead adder where carry-in is calculated in advance to eliminate carry dependency. In the LAC, a look-ahead logic computes the bits which decides the which number is greater/smaller. The block diagram of the look-ahead block accepts two four bit numbers and generates 4-bit comp output. Only one bit of the comp out will be high if the number is greater or smaller else are zero it reflect that numbers are equal.

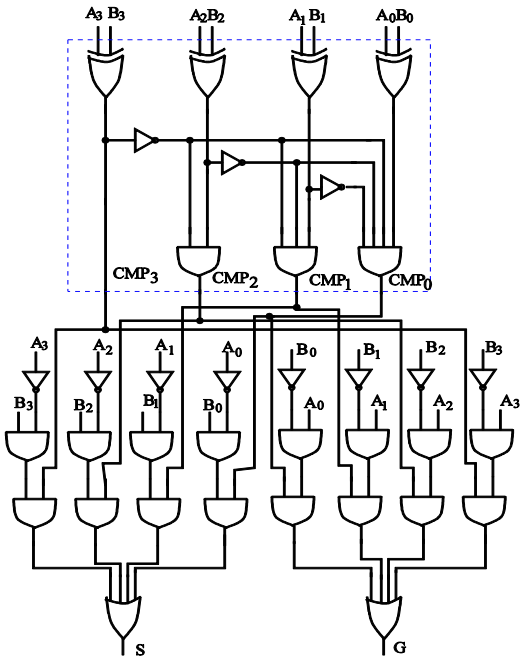

Figure 3: Block diagram of LAC

The logical diagram of the look-ahead logic [14] as shown in Figure 3 requires four XOR gate to find 1's which corresponds to the '0' in the other number and then provides ultimate compare output based on the value of the XNOR out. If the most significant XOR is at logic '1' it reflect that this bit will cause output will be greater/smaller, if this bit is on logic '0', other significant XOR will be searched. The LAC as shown in Fig. 3 employs look-ahead logic with some additional logic.

2.4 Subtractor Based Comparator

This comparator utilizes the subtractor to compare two numbers [15-16]. When the number A is smaller than number B, result of the subtractor will be negative which can be easily detected by the sign bit. If sign bit is '1' it reflect negative difference which in turn shows that number A is smaller than B. On the other hand, when the two numbers are equal it will provide zero output, in this case all bits of the difference will be zero that can be detected by simple OR gate. If all bits of the OR gate are zero then output will be logic '0' else it will be logic '1'. The equal and the small signals are further used to generate greater signal.

III. PROPOSED COMPARATOR

The block diagram of the proposed area efficient comparator is shown in Figure 4. It can be observed from the block diagram that proposed comparator elimiates the second stage of the PBC. Therefore it significantly reduces the area, power and delay over the PBC.

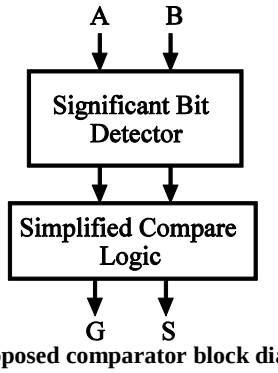

Figure 4: Proposed comparator block diagram

The proposed encoder works in the similar manner as the priority encoder. The proposed comparator also eliminates the 1's corresponding to the 1's in the other number in the same bit position. It does not eliminates least significant 1's if the number existing 1's in the most significant bit position as it done in the priority based comparator architecture. It is not necessary to remove these 1's as it requires more hardware.

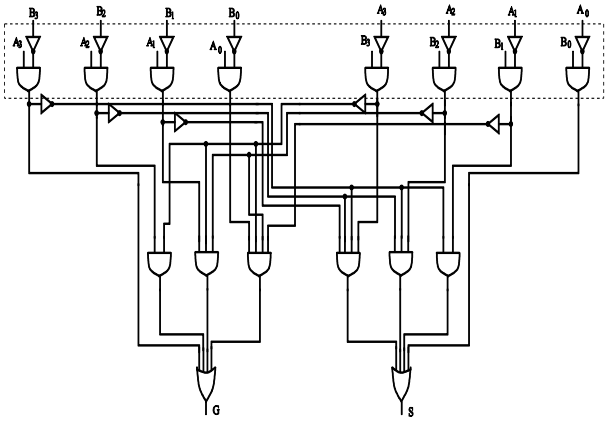

Figure 5: Circuit diagram of proposed comparator

The circuit diagram of the proposed comparator as shown in Figure 5 computes greater/smaller signal using the first stage output. Therefore significantly reduces the area, power and delay. The simulation results in the next section shows efficacy of the proposed comparator over the existing comparator architectures.

IV. SIMULATION RESULT ANALYSIS

In order to evaluate the efficacy of the proposed comparator over the existing, all the comparator designs are implemented in Verilog. These Verilog designs are synthesized using Xilinx ISE 14.5. Test bench for all the designs are created and simulated to verify the functionality of each designs. Further, design metrics such as area, power and delay are extracted and compared. Following subsections provides the simulation results and their analysis for the proposed comparator over the existing comparators.

The design metrics for 4-bit comparators and 16-bit comparators are shown in the Table 1.

Table 1: Metrics of the 4-bit comparators

Technique Comparator		Area (#LUT)	Delay (nS)	Power (mW)
4-bit Comparators	Traditional	3	1.455	30
	Priority Based	4	1.444	31
	Look-ahead	4	1.522	36
	Subtractor based	5	1.511	35
	Proposed	3	1.221	25
16-bit Comparators	Traditional	20	2.556	151
	Priority Based	20	2.466	153
	Look-ahead	29	3.025	158
	Subtractor based	18	2.922	163
	Proposed	17	2.444	131

It can be observed from the simulation results that proposed 4-bit comparator requires 19.35% reduced power over priority comparator. Further, it can also be observed that proposed comparator requires 25%, 25% and 20% reduced area over PBC, LAC and Subtractor based comparator respectively as shown in Figure 7. These 4-bit comparators are used to design 16-bit comparators whose simulation results are also mention in the Table 1.

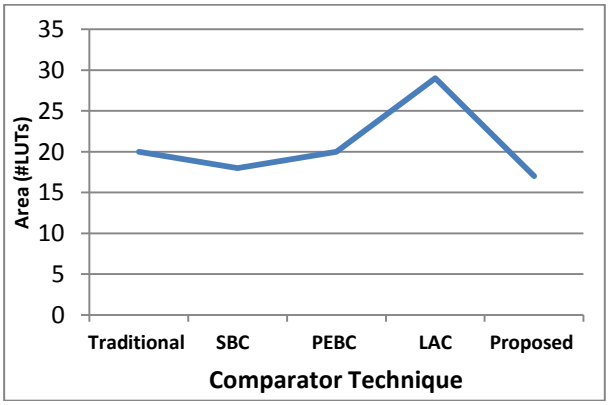

Figure 6: Area of various 16-bit comparators

Figure 7 illustrate that the proposed 16-bit comparator exhibits smaller delay over the look-ahead. The proposed comparator requires 19.2% and 16.3% less delay over LAC and subtractor based comparator respectively.

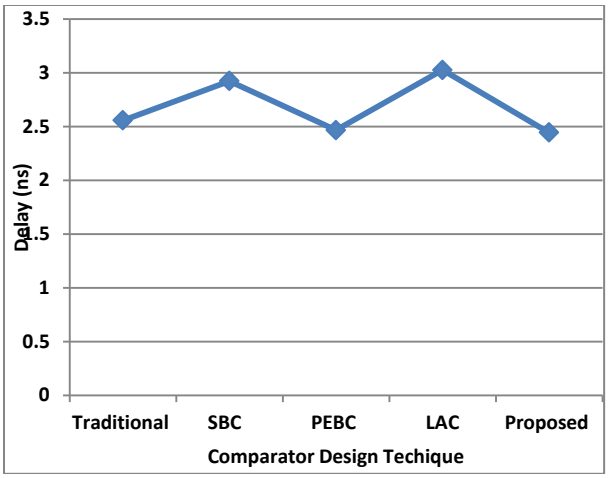

Figure 7: Delay of various 16-bit comparators

Further, Figure 13 shows that the proposed comparator exhibits smaller power over the existing. The proposed comparator shows 13.2%, 14.3%, 17.3%, and 19.6% reduced power over the traditional, priority based, look-ahead and subtractor based comparator, respectively.

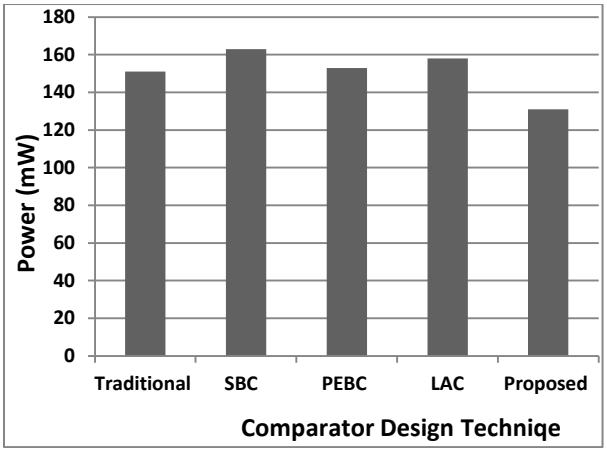

Figure 8: Power of 4-bit comparators

All these results shows efficacy of the proposed comparator over the existing architectures.

V. CONCLUSION

In this paper the existing comparator architectures are explored and presented a new comparator that shows significant improved performance over the existing comparator architectures. The proposed and existing comparator are implemented in Verilog and processed with Xilinx ISE tool chain. Further the designs are synthesized and post synthesis results are extracted. The extracted metrics are compared. The simulation results show nearly 13.2% power reduction over the best known architecture. Thus, the proposed comparator can be effectively utilized different signal processing applications such as image/video processing.

REFERENCES

[1] M. Morris Mano, Digital Logic and Computer Design.
[2] J. Eyre and J. Bier, "DSP processors hit the mainstream," IEEE Computer, pp. 51–59, 1999.
[3] J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach. Morgan Kaufmann Publishers, 2002.
[4] T. Suryakala et al. "Circuit Design of Low area 8-bit magnitude Comparator With Low Power by Static CMOS", Int. Journal of Advanced Research in Computer Engg & Tech. Vol 4 Issue 10, Oct 2015, pp 3982-86.
[5] Anjuli et al. "2-Bit Magnitude Comparator Design Using Different Logic Styles", Int. Journal of Engg Science Invention, Vol 2 Issue 1 Jan 2013 PP.13-24
[6] Sharma et al. "A Low Power 8-bit Magnitude Comparator with Small Transistor Count using Hybrid PTL/CMOS Logic" Int. Journal of Computational Engineering & Management, Vol. 12, April 2011, pp 110-115.
[7] Ergin et al. "A Circuit–Level Implementation of Fast, Energy–Efficient CMOS Comparators for High–Performance Microprocessors" IEEE Int. Conf on Computer Design: VLSI in Computers and Processors, pp. 115-118
[8] Chung et al. "High-Performance and Power-Efficient CMOS Comparators", IEEE JSSC, Vol 38, No 2, Feb 2003, pp 254-262.
[9] Lam et al. "A MUX-based High-Performance Single-Cycle CMOS Comparator" IEEE Tran on circuits and systems—II: express briefs, Vol. 54, No. 7, July 2007, pp 591-595
[10] Perri et al. "Fast Low-Cost Implementation of Single-Clock-Cycle Binary Comparator" IEEE Tran. on circuits and systems—II, Vol. 55, No. 12, Dec 2008 pp 1239- 1243.
[11] Frustaci et al. "A New Low-Power High-Speed Single Clock Cycle Binary Comparator", pp 317-320
[12] Pierce Chuang, David Li, and Manoj Sachdev "A Low-Power High-Performance Single-Cycle Tree-Based 64-Bit Binary Comparator", IEEE Tran on circuits and systems—II: express briefs, Vol. 59, No. 2, Feb 2012, pp 108-112.
[13] Deb et al. "High-Speed Comparator Architectures for Fast Binary Comparison" 3rd Int. Conference on Emerging Applications of information Technology, pp 454- 457
[14] Hafiz et al. "Approach to design a compact reversible low power binary comparator", IET Computers & Digital Techniques, 2014, Vol. 8, Iss. 3, pp. 129–139
[15] Vijaya et al. "2-Bit Magnitude Comparator using GDI Technique", IEEE Int Conf on Recent Advances and Innovations in Engineering, May 09-11, 2014, 225-229
[16] Shun-Wen Cheng, "A High-Speed Magnitude Comparator with Small Transistor Count" in Proceedings of IEEE international conference ICECS, 1168 - 1171 Vol.3, Dec 2003.

