

VLSI IMPLEMENTATION OF HIGH PERFORMANCE DIGITAL COMPARATOR FOR ANALOG SIGNAL PROCESSING APPLICATION

Ayee Vinotha J¹, Priyadarshni.S²

¹Assistant Professor, Department of ECE, Coimbatore Institute of Engineering and Technology, Tamilnadu, India

²Assistant Professor, Department of ECE, Coimbatore Institute of Engineering and Technology, Tamilnadu, India

Abstract

A digital comparator is one of the fundamental computational elements in most digital circuit components such as microprocessor and in digital signal processing designs as well. A new and efficient comparator design featuring with reduced area and power is presented here. The comparator executes its comparison operation from most significant bit to the least significant bit only when the number of comparison bits is equal. The comparator architecture is based on cut set algorithm which reduces the silicon area by time multiplexing many operations into single functional units. This type of algorithm reduces the power dissipation by eliminating unnecessary transitions. In addition, the comparator design is simple and flexible since it makes use of combinational circuitry alone. The main advantage of this design is its low power which can be tapped in digital signal processing applications like analog to digital convertor.

Keywords: Comparator Architecture Cut Set Algorithm, Fully Digital Analog To Digital Convertor

1. INTRODUCTION

Comparator is one of the most basic components in all digital system. This can be used in wide range of application like sorting, searching data, graphical and image/signal processing, testing application and in all general processor components. Even though the design of comparator is easy to implement, the extensive use of comparator places a challenging task on area and power consumption. To design a comparator some previous work is made using with certain algorithms and with dynamic logic gates for their increased performance. For example, various design uses subtractor (flat adder components), but these design are slow and area prohibitive even when they are implemented in fast adders. To improve scalability and reduce delay, a combined two's complement and a floating point comparator is designed which supports 32 bit and 64 bit values [6]. However these circuits will have a greater delay and consumes a large area, even when implemented in a magnitude comparator with logarithmic delays.

To reduce area and power, a high speed magnitude comparator is designed. The principle is based on modified 1's complement and a conditional sum adder. One's complement addition is designed in such a way that it has two input multiplexer at each level and a generate-propagate term at the first level. Even for a 64 bit width comparison, six level of logic structure is required and all the structure is active irrespective of the bit width. This leads to high power consumption on the circuit and moreover the structure cannot be used to perform equality operation [3]. With the actual operands' bit width, several designs are proposed in the form of pipelining. The concept used in [10] reduces the switching activity with the help of power down mechanism thereby reducing the power consumption. In order to acquire a large fan in, all-N-transistor (ANT) circuit is designed

with high pipelined throughput. In [4] the evaluation block of a dynamic cell, where rearrangement and reordering of transistor is done thereby reducing the transistor count. An alternative architecture uses multiphase clocking scheme with only three pipeline stages where the clocking scheme may be incompatible for high speed single-cycle processor.

Comparator used in [5] are specifically delineate for priority encoder with the help of priority-encoding (PE) algorithm. To reduce logic depth, some dynamic logic is merged with PE function which results in multiple output domino logic (MODL). Another technique which shortens the path of priority-token propagation is multilevel look-ahead technique. The three techniques used above will limit the clock speed and jitter margin. A 64-bit comparator is designed with the help of multiplexer-based structure. The structure is splitted into two stages; the first stage consists of 8 bit comparison module, where second stage contains the multiplexer structure instead of using the same comparison module. The multiplexer which is used in second stage will further limit the clock speed and jitter margin.

To enhance speed of the comparator, the comparator is design using two phase domino clocking structure. These structures using addition operation as their basic element where it perform two's complement function for the addend bit and carry bit decides the "greater than" or "lesser than" operation. In [2] architecture uses ripple carry adder which can be implemented for a wide range. This dynamically reduces the area and save power. A faster, energy efficient comparator takes the advantages, where only a particular bit slices is executed. The comparison starts from the most significant bit to the least significant bit, where undulate one bit at a time whereby reducing the delay and switching speed.

To reduce propagation delay, transistor count and physical area, a circuit is designed with no arithmetic computation which can be implemented in bitwise competition logic (BCL), pre-encoder and selection logic. To avoid long execution time, the BCL module is divided into block and then the signal is sending for the selection logic. Here the multiplexer acts as selection logic. The main drawback of the BCL circuit is, it limits the maximum achievable operating speed. To avoid this problem a special control logic circuitry should be designed.

2. EXISTING DESIGN

A novel parallel prefix structure contain reconfiguration arithmetic algorithm, with a maximum fan-in and fan-out of five and four respectively. The tree structure is globally partitioned into two pipelining stages and a number of stages at further levels. The overview of the architecture contains two modules. One is the comparison resolution module, which act as a high-level architecture of the design. Another module is the decision module, which contain OR-network.

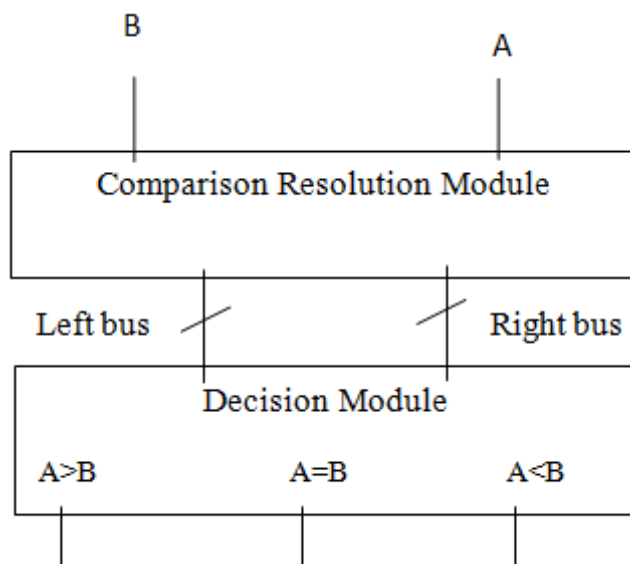


Fig -1: Parallel Prefix Structure

Comparison resolution module compares from the most significant bit to the least significant bit performing bitwise comparison logic. The module contains five sets of logic, where each set has a particular cell type of function to be performed. In order to reduce switching activity inside the circuit, lower bits are terminated once the 1 is detected in the most significant bit. This termination process is done in set three and set two logic type cells.

$$\left\{ \begin{array}{l} \text{If } A > B; \text{ then left bus}=1 \text{ and right bus}=0, \\ \text{If } A < B; \text{ then left bus}=0 \text{ and right bus}=1, \\ \text{If } A = B; \text{ then left bus}=0 \text{ and right bus}=0, \end{array} \right\}$$

Once the bit is terminated, the structure is divided into two parallel buses as left bus and right bus. Depending upon the output in the comparison resolution module, either left bus or right bus is selected.

The decision module contains the OR-network, where the final comparison is done with the help of scanning the left bus and right bus parallel in a separate chain.

3. PROPOSED DESIGN

The novel comparator is designed with the concept of cut set algorithm. A cut set is a group of edge which is partition into sub graph that can be connected at least with a single edge. Here group of edges denotes the logic gates and the sub graph denotes the identical logic gates. In this paper, the identical logic elements are group into a structure and those structures are separated into groups. Each particular group of logic elements is folded back to itself for a maximal usage of the logic gates. Hence the interconnection and power is reduced. Each time the results are stored in a separate register and passed to the new recommencing results. The register which contain the results are feed to an OR network where the comparator outputs are obtained. This type of cut set algorithm will naturally reduce the power.

3.1 Application

Analog to digital convertor (ADC) is a basic building block that encodes the physical quantity to digital number. An analog to digital convertor are used in many more application especially in sensor networks. As sensor networks are analog in nature, it is essential to convertor an analog signal to digital signal form. There are many applications where both analog signal and digital signal are used in a single circuit network. In such case the designer implements the analog circuit in the same digital CMOS technology. It is preferable to have a fully digital analog to digital convertor, which can be implemented in CMOS circuitry. On comparison with analog ADC, fully digital ADC will have a less power and area consumption. In this way a digital comparator which is designed using cut set algorithm is implemented in fully digital analog to digital convertor.

In this paper, a fully digital analog to digital convertor consist of a reference signal, an input signal and a comparator block. The input and the reference signals are sound and sign wave signals respectively. These sign wave and sound signal are generated from MATLAB. The signals which are generated from MATLAB are sample and hold internally and they are generated in the form of 1's and 0's (digital signals). The internal sample and hold operation consists of circuit that contain continuously varying analog signal which is in the form of voltage and locks its value for a constant interval of time. The constant input values are converted into a digital form. The digital signals which are generated by the reference and sign wave signal are feed to the input of the comparator as 'A' and 'B' respectively.

4. SIMULATION RESULT

4.1 Comparator Output

The coding for comparator which is designed using cut set algorithm is done and the output is shown below.

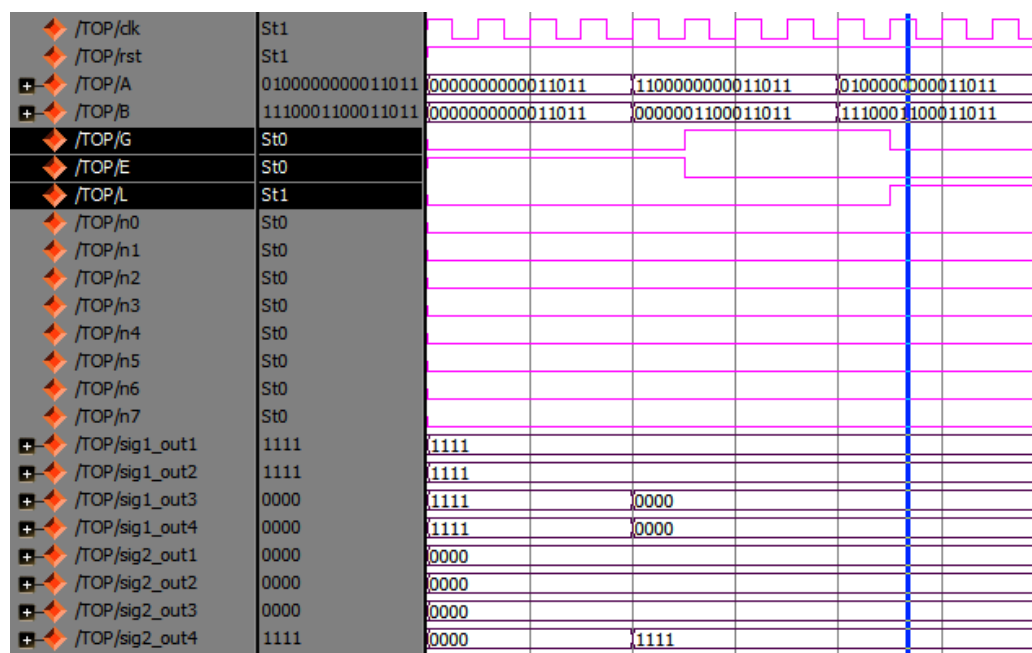


Fig -2: Output Waveform for Comparator

The output results shows that A and B are two input which is of 16 bit width. The output are G, E, L as greater than, equal to and lesser than respectively.

4.2 Fully Digital ADC Output

The output for fully digital ADC is designed and shown below. The result consist of clock and reset as input signal and the discrete signal as the output signal.

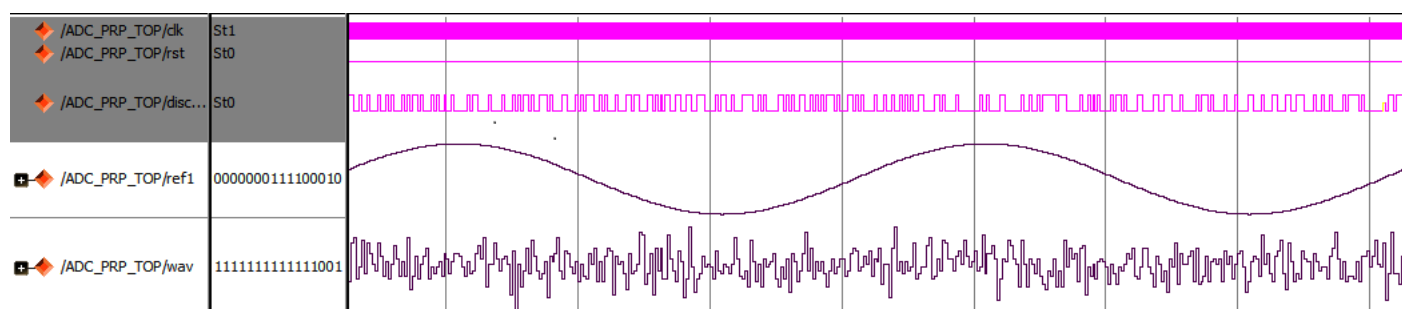


Fig -3: Output Waveform For Fully Digital ADC

The reference and input signal which are generated from MATLAB are as follows.

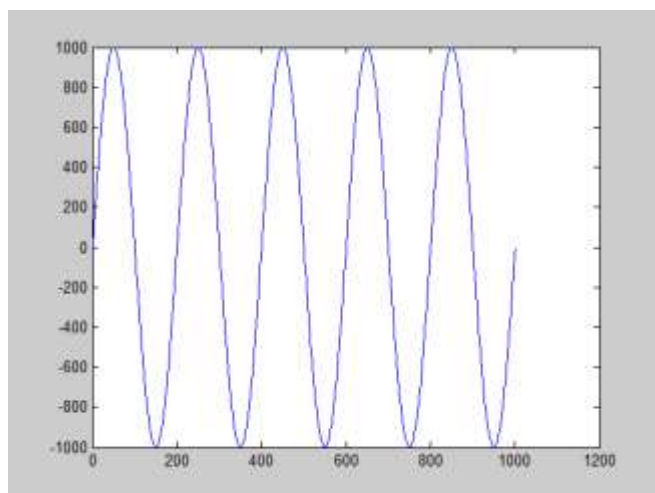


Fig -4: Reference Signal

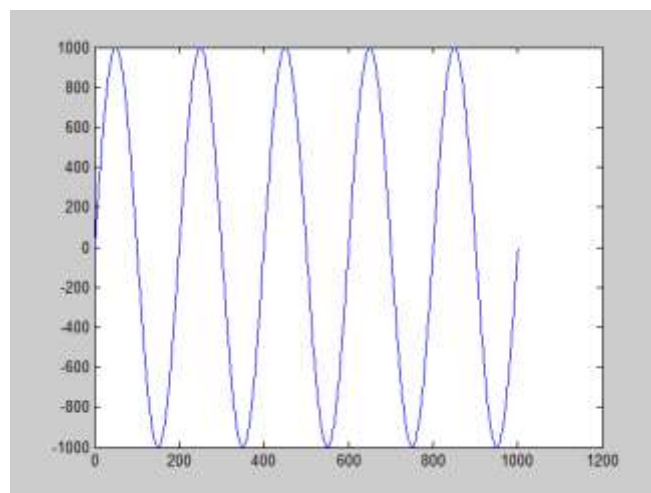


Fig -5: Input Signal

5. PERFORMANCE ANALYSIS

5.1 For Comparator Design

The power analysis is done for both prefix tree structure and the comparator which is designed using cut set algorithm.

Table -1: Power Analysis for Comparator Design

PARAMETER	EXISTING	PROPOSED
POWER	133.52Mw	133.05Mw

5.2 For Fully Digital ADC Design

A comparative power and area analysis is done for digital ADC which is designed using both parallel prefix structure comparator and cut set algorithm comparator.

Table -1: Power Analysis for Comparator Design

PARAMETER	EXISTING	PROPOSED
POWER	518.11mW	337.27mW
AREA	6036	3025

6. CONCLUSION

In this paper, a novel high performance comparator is designed using cut set algorithm where a replicated circuit is partition into sub circuit and finally these sub circuits are connected to generate the result. This type of design tends to reduce power in the circuit. Thus the design provides an efficient way for comparator wide range and high speed application. The comparator which is designed using cut set algorithm is implemented in fully digital Analog to digital convertor in which the power and area are analyzed.

REFERENCES

- [1]. Saleh Abdel-Hafeez, Ann Gordon-Ross and Behrooz Parhami, "Scalable digital CMOS comparator using a parallel prefix tree", IEEE transactions on very large scale integration systems.
- [2]. J. D. Bruguera and T. Lang, "Multilevel reverse most-significant carry computation," vol. 9, no. 6, pp. 959–962, Dec. 2001.
- [3]. S.-W. Cheng, "A high-speed magnitude comparator with small transistor count," vol. 3, Dec. 2003, pp. 1168–1171.
- [4]. C.-C. Wang, P.-M. Lee, C.-F. Wu, and H.-L. Wu, "High fan-in dynamic CMOS comparators with low transistor count," vol. 50, no. 9, pp. 1216–1220, Sep. 2003.
- [5]. C.-H. Huang and J.-S. Wang, "High-performance and power-efficient CMOS comparators," vol. 38, no. 2, pp. 254–262, Feb. 2003.
- [6]. J. E. Stine and M. J. Schulte, "A combined two's complement and floating-point comparator," vol. 1, 2005, pp. 89–92.
- [7]. J. Hensley, M. Singh, and A. Lastra, "A fast, energy-efficient zcomparator," 2005, pp. 41–44.
- [8]. H.-M. Lam and C.-Y. Tsui, "High-performance single clock cycle CMOS comparator," vol. 42, no. 2, pp. 75–77, Jan. 2006.

- [9]. W. Belluomini, D. Jamsek, A. K. Nartin, C. McDowell, R. K. Montoye, H. C. Ngo, and J. Sawada, "Limited switch dynamic logic circuits for high-speed low-power circuit design," *IBM J. Res. Develop.*, vol. 50, nos. 2–3, pp. 277–286, Mar.–May 2006.
- [10]. H.-M. Lam and C.-Y. Tsui, "A mux-based high-performance single-cycle CMOS comparator," vol. 54, no. 7, pp. 591–595, Jul. 2007.
- [11]. J.-Y. Kim and H.-J. Yoo, "Bitwise competition logic for compact digital comparator," Nov. 2007, pp. 59–62.
- [12]. S. Perri and P. Corsonello, "Fast low-cost implementation of singleclock- cycle binary comparator," vol. 55, no. 12, pp. 1239–1243, Dec. 2008.
- [13]. B. Parhami, "Efficient hamming weight comparators for binary vectors based on accumulative and up/down parallel counters," vol. 56, no. 2, pp. 167–171, Feb. 2009.
- [14]. D. V. Ponomarev, G. Kucuk, O. Ergin, and K. Ghose, "Energy efficient comparators for superscalar datapaths," vol. 53, no. 7, pp. 892–904, Jul. 2004.
- [15]. H. Suzuki, C. H. Kim, and K. Roy, "Fast tag comparator using diode partitioned domino for 64-bit microprocessor," vol. 54, no. 2, pp. 322–328, Feb. 2007.
- [16]. Cedric Walravens and Wim Dehaene, "Low-power digital signal processor architecture for wireless sensor nodes" pp.313-321, Mar 2013.
- [17]. Hoorman Farkhani, Maohammad Meymandi-Nejad and Manoj Sachdev, "A fully digital ADC using a new delay element with enhanced linearity" pp.2406-2409, May 2008.