

A 65nm Parallel Prefix High Speed Tree based 64-Bit Binary Comparator

Er. Deepak sharma
Student M-TECH,
Yadavindra College of Engineering,
Punjabi University, Guru Kashi
Campus, Talwandi Sabo.

Er. Parminder singh
Assistant Professor,
Yadavindra College of Engineering,
Punjabi University, Guru Kashi
Campus, Talwandi sabo.

Abstract- We present a high speed and lower power consumption 64-bit comparator based on parallel prefix tree based structure that leverages the final result outcome of the most significant bit, proceeding bitwise toward the least significant bit only when the input bits are equal. This method reduces complexity of design and dynamic power dissipation by eliminating unnecessary transitions. Simulation results show this proposed work is much faster and less power consumption design compare to previous comparator. The minimum measured time delay of proposed 64-bit comparator is 231ps which is 62% and 49% less compare to implemented previous 64-bit comparators[7], [12] respectively and the average power consumption during operation of this 64-bit comparator is $2.37\mu\text{w}$ which is 30% and 56% less compare to implemented previous 64-bit comparators[7], [12] respectively. All simulations are done on Tanner EDA V13 using 65nm technology.

I. INTRODUCTION

The comparison of two bits is one of most widely used in computer systems, device interfaces for equality, scientific computation (graphics and image/signal processing), test circuit applications (jitter measurements, signature analyzers, and built-in self test circuits), and optimized equality-only comparators for general-purpose processor components (associative memories, load-store queue buffers, translation look-aside buffers, branch target buffers, and many other CPU argument comparison blocks). Also it is an important arithmetic operation in DSP applications, digital filter. Digital signal processing (DSP) is one of most important units in electronic devices. Function of basic binary comparator is to compare the two bits suppose and it gives result in terms of A greater than B ($A > B$), A less than B ($A < B$) and A equal to B ($A = B$).

In past decade several comparators architecture are introduced. Mostly of these designs consume more power, take more time to operation and these were large in size. Recently tree-based comparator that utilizes dynamic Manchester adders are introduced [7], [12]. These comparators had showed high performance compared to previous comparators. According to this structure comparator is divided into several building blocks and the output of one's output is given as input to another building block until final result output comes out. In this brief,

a new tree structure based 64-bit comparator is modified from existing architecture [12] to improve its operating speed and

power efficiency from existing comparators is introduced in this work. Moreover, this work provides the highly performance 64-bit comparator as comparison between comparators is shown and performance parameter for comparison taken are time delay, Average power consumption and number of transistor. This work is performed using Tanner EDA V13 as simulation tool software at 65nm technology modal file.

The rest of this brief is organized as follows: Section II reviews previous comparator works. Section III introduces and explains the proposed comparator implementation. IN section IV schematic view of all sets is shown and described. Section V describes the simulation setup and presents the delay and power simulation results. Finally, Section VI concludes with some final remarks and comments.

II. EXISTING COMPARATOR DESIGNS

A brief description of the design principles of existing comparator is provided here. Interested readers are referred to [12] for a detailed overview. The comparison resolution module (which depicts the high-level architecture of this design) is a novel MSB-to-LSB parallel-prefix tree structure that performs bitwise comparison of two N -bit operands A and B , denoted as $AN-1, AN-2, \dots, A0$ and $BN-1, BN-2, \dots, B0$, where the subscripts range from $N-1$ for the MSB to 0 for the LSB. The comparison resolution module performs the bitwise comparison asynchronously from left to right, such that the comparison logic's computation is triggered only if all bits of greater significance are equal.

The parallel structure encodes the bitwise comparison results into two N -bit buses, the left bus and the right bus, each of which store the partial comparison result as each bit position is evaluated, such that

if $A_k > B_k$, then $\text{left}_k = 1$ and $\text{right}_k = 0$
if $A_k < B_k$, then $\text{left}_k = 0$ and $\text{right}_k = 1$
if $A_k = B_k$, then $\text{left}_k = 0$ and $\text{right}_k = 0$.

In addition, to reduce switching activities, as soon as a bitwise comparison is not equal, the bitwise comparison of every bit of lower significance is terminated and all such positions are set to zero on both buses, thus, there is never

more than one high bit on either bus. The decision module uses two OR-networks to output the final comparison decision based on separate OR-scans of all of the bits on the left bus (producing the *L* bit) and all of the bits on the right bus (producing the *R*

performs a specific function whose output serves as input to the next set, until the fifth set produces the output on the left bus and the right bus. All cells (components) within each set operate in parallel, which is a key feature to increase operating speed while minimizing the transitions to a minimal set of leftmost bits needed for a correct decision. This prefixing set structure bounds the components fan-in and fan-out regardless of comparator bitwidth and eliminates heavily loaded global signals with parasitic components, thus improving the operating speed and reducing power consumption. Additionally, the OR-network's fan-in and fan-out is limited by partitioning the buses into 4-b groupings of the input operands, thus reducing the capacitive load of each bus.

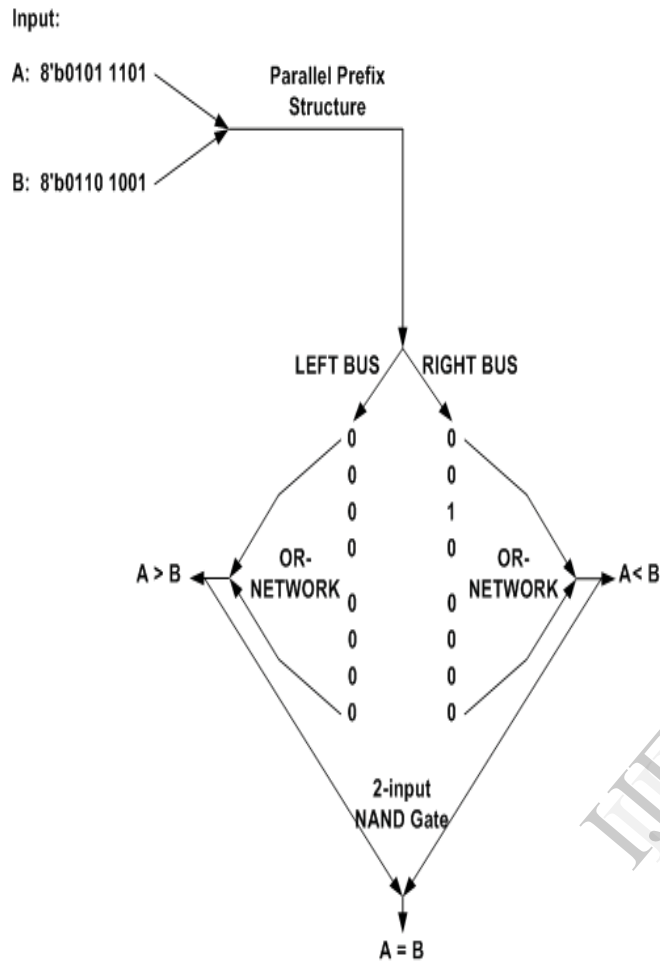


Fig. 1. Example 8-b comparison.[12]

bit). If LR = 00, then $A = B$, if LR = 10 then $A > B$, if LR = 01 then $A < B$, and LR = 11 is not possible. An 8-bit comparison of input operands $A = 01011101$ and $B = 01101001$ is illustrated in Fig. 1. In the first step, a parallel prefix tree structure generates the encoded data on the left bus and right bus for each pair of corresponding bits from *A* and *B*. In this example, $A_7 = 0$ and $B_7 = 0$ encodes as left₇ = right₇ = 0, $A_6 = 1$, and $B_6 = 1$ encodes as left₆ = right₆ = 0, and $A_5 = 0$ and $B_5 = 1$ encodes left₅ = 0 and right₅ = 1. At this point, since the bits are unequal, the comparison terminates and a final comparison decision can be made based on the first three bits evaluated. The parallel prefix structure forces all bits of lesser significance on each bus to 0, regardless of the remaining bit values in the operands. In the second step, the OR-networks perform the bus OR-scans, resulting in 0 and 1, respectively, and the final comparison decision is $A > B$.

The structure of this comparator is partitioned into five hierarchical prefixing sets, as depicted in Fig. 2, with the associated symbol presentations in Tables I, where each set

TABLE I
SYMBOL NOTATION AND DEFINITIONS [12]

Symbol (Cells)	Definition
<i>N</i>	Operand bitwidth
<i>A</i>	First input operand
<i>B</i>	Second input operand
<i>R</i>	Right bus result bit
<i>L</i>	Left bus result bit
Σ	Bitwise AND
Π	Bitwise OR
$T\{*\}$	Logic function of cell type*
$COMP\{*\}$	Complement function of set

This is based on using a novel parallel prefix tree (Tables I contain symbols). Each set or group of cells produces outputs that serve as inputs to the next set in the hierarchy, with the exception of set 1, whose outputs serve as inputs to several sets. Set 1 compares the *N*-bit operands *A* and *B* bit-by-bit, using a single level of Ψ type cells. The Ψ type cells provide a termination flag D_k to cells in sets 2 and 4, indicating whether the computation should terminate. These cells compute (where $0 \leq k \leq N - 1$)

$$\Psi: D_k = A_k \oplus B_k. \tag{1}$$

Set 2 consists of Σ_2 -type cells, which combine the termination flags for each of the four Ψ type cells from set 1 (each Σ_2 -type cell combines the termination flags of one 4-b partition) using NOR-logic to limit the fan-in and fan-out to a maximum of four. The Σ_2 -type cells either continue the comparison for bits of lesser significance if all four inputs are 0s, or terminate the comparison if a final decision can be made. For $0 \leq m \leq N/4 - 1$, there is a total of $N/4$ Σ_2 -type cells, all functioning in parallel

$$\Sigma_2: C_{2,m} = COMP(\sum_{i=4m}^{4m+3} D_i) \tag{2}$$

Set 3 consists of Σ_3 -type cells, which are similar to Σ_2 -type cells, but can have more logic levels, different inputs, and carry different triggering points. A Σ_3 -type cell provides no comparison functionality; the cell's sole purpose is to limit the fan-in and fan-out regardless of operand bitwidth. To limit the Σ_3 -type cell's local interconnect to four, the number

of levels in set 3 increases if the fan-in exceeds four. Set 3 provides functionality similar to set 2 using the same NOR logic to continue or terminate the bitwise comparison

activity. If the comparison is terminated, set 3 signals set 4 to set the left bus and right bus bits to 0 for all bits of lower significance. For 0 <=

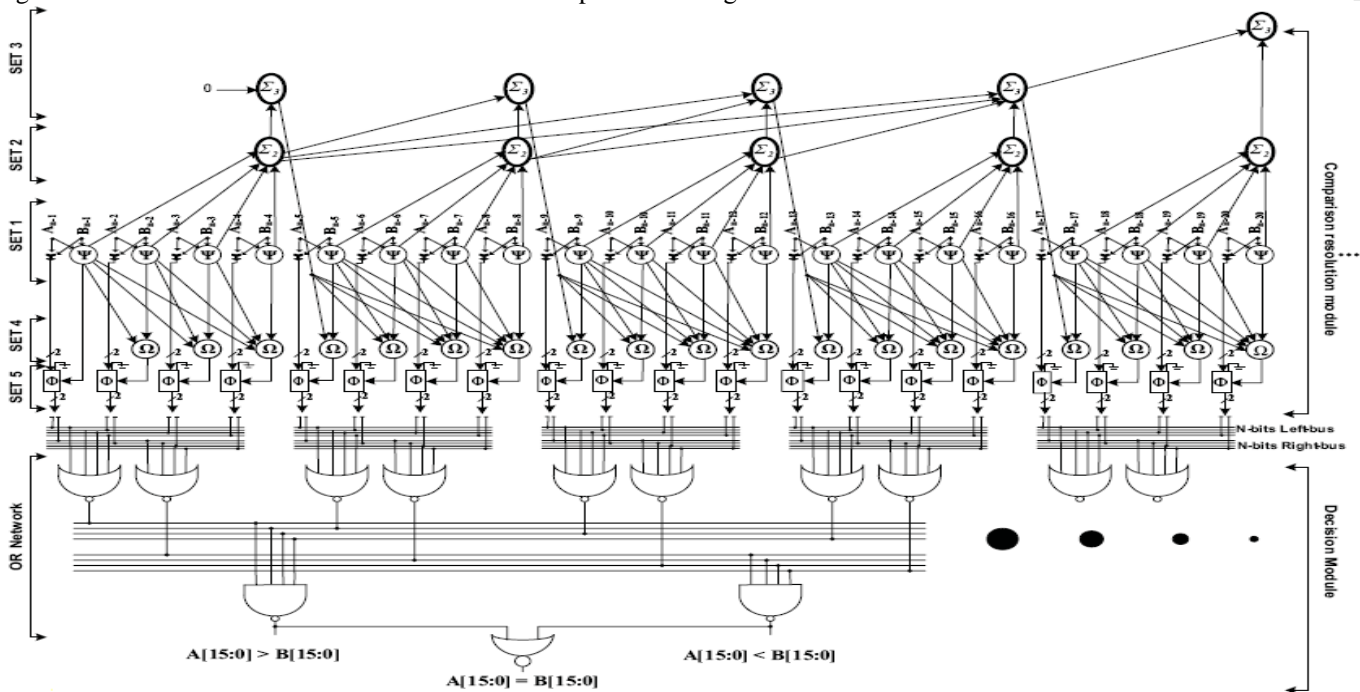


Fig. 2. Implementation details for the comparison resolution module (sets 1 through 5) and the decision module of existing comparator design [12].

$m \leq N/4 - 1$, there is a total of $N/4$ $\Sigma 3$ -type cells per level, with cell function and number of levels as

$$\Sigma 3: C_{3,m} = \text{COMP} \left(\sum_0^m C_{2,i} \right) \quad (3)$$

TABLE II
LOGIC GATE REPRESENTATIONS FOR SYMBOLS
SED IN FIG. 2.[12]

Symbols (Cells)	Logic Gate	Maximum Fan-in/Fan-out And (Transistor Counts)
		2 / 4 (12)
		4 / 4 (8)
		5 / 1 (20)
		3 / 2 (12)

MUX-Logic
TG: Transmission Gate

From left to right, the first four $\Sigma 3$ -type cells in set 3 combine the 4-b partition comparison outcomes from the one, two, three, and four 4-b partitions of set 2. Since the fourth $\Sigma 3$ -type cell has a fan-in of four, the number of levels in set 3 increases and set 3's fifth $\Sigma 3$ -type cell combines the comparison outcomes of the first 16 MSBs with a fan-in of only two and a fan-out of one.

$$\Omega: Y_k = C_{3,[k/4]-1} D_k \prod_{i=4}^{k-1} \bar{D}_i \quad (4)$$

The number of inputs in the Ω type cells increases from left to right in each partition, ending with a fan-in of five. Thus, the Ω type cells in set 4 determine whether set 5 propagates the bitwise comparison codes. Set 5 consists of N Φ type cells (two-input, 2-b-wide multiplexers). One input is (A_k, B_k) and the other is hardwired to "00." The select control input is based on the Ω type cell output from set 4. We define the 2-b as the left-bit code (A_k) and the right-bit code (B_k), where all left-bit codes and all right-bit codes combine to form the left bus and the right bus, respectively. The Φ type cells compute (where $0 \leq k \leq N-1$)

$$\Phi: F_k^{1,0} = Y_k \times M_k + \bar{Y}_k \times 00 \quad (5)$$

The output $F_k^{1,0}$ denotes the "greater-than," "less-than," or "equal to" final comparison decision

$$\begin{aligned} \text{If } F_k^{1,0} &= 00, \text{ for } & Ak = Bk \\ &= 01, \text{ for } & Ak < Bk \end{aligned}$$

$$= 10, \text{ for } Ak > Bk.$$

Essentially, the 2-b code $F_k^{1,0}$ can be realized by OR-ing all left bits and all right bits separately, as shown in the decision module (Figs. 2 and 3), using an OR-gate network in the form of NOR-NAND gates yielding a more optimum gate structure

$$GL_{1,j}^1 = \sum_{k=4j}^{4j+3} F_k^1 \quad (6)$$

$$GR_{1,j}^0 = \sum_{k=4j}^{4j+3} F_k^0 \quad (7)$$

The superscripts "1" and "0" in (6) and (7) denote the summation of the left and right bits, respectively, and the subscript "1" denotes the first level of OR-logic in the decision module that receives data directly from set 5. If we limit the fan-in of each gate to four, the number LDM of the OR-gate tree levels for the decision module is given by

$$LDM = \lceil \log_4 N \rceil. \quad (8)$$

III. THE PROPOSED WORK

A novel scalable parallel prefix based structure that assisted the comparison outcome of the most significant bit, further propagate bitwise toward the least significant bit only when the compared bits are equal. This comparator is also divided in five sets similar to previous discussed comparator. The basic design principle is to similar to previous comparator [12]. The changes are done in sets and function of comparator. Changes done in existing architecture are given below:-

a) The previous comparator architecture is divided into group of 4 parallel input bits and it works in group of four and in this proposed work the grouping of bits only 2. By doing this change the number of the maximum FAN IN and FAN OUT become three instead of five. This change helps comparator to work faster and reduce its design complexity.

b) Second change is done in set 1 of this comparator architecture [12]. In set 1 of previous comparator XOR gate is used based on static logic and require 12 transistors to complete circuit and in this proposed comparator another type of XOR gate based on pass transistor gate is used. This XOR gate is taken from previous comparator [7]. In this work [7], XNOR gate is used and for proposed work complementary circuit of XNOR gate i.e. XOR gate is used. It takes only 6 transistors to complete circuit.

c) Due to change in grouping of bits, set 2 is also changed. Now this set becomes two input NOR instead of four input NOR gate.

d) In set 3 of proposed work, the pass transistor logic based AND gate is used instead of NOR gate used. It makes comparator faster, but this change does not change the function of comparator as inverter is removed from next stages.

e) In set 4 similarly due to change in grouping of bits also limit the FAN IN to three and it also reduce the size of this circuit. Now this set work for maximum of 3 inputs instead of maximum 5 inputs used in reference architecture.

f) Finally last modification is done in set 5, as this contains 2-bit wide 2 inputs multiplexer and in this previous circuit shown for both MUX, one input is connected A_k, B_k and other is hardwired with ground or 00 and for this input it require only one NMOS to propagate 0 as NMOS transistor is good conductor to pass 0.

This comparator is based on using a novel parallel prefix tree. These sets or group of cells generate outputs that are given as inputs to the next set in the hierarchy, except set 1, its outputs given as inputs to more than one set. Detail working or function done by these sets is given below.

Set 1 contains the XOR gate. This set compares the N-bit A, B bit-by-bit, with a single level of N Ψ type cells. This type cells basically have a function of termination flag D_k to cells in sets 2 and 4, and these compute (where $0 \leq k \leq N - 1$)

$$D_k = A_k \oplus B_k. \quad (9)$$

Set 2 consists of NOR type cells, this set contains the termination flags from each of the two set 1 (as two outputs of 2-b partition of set 1 will be given as inputs to set 2 as a termination flags of one 2-b partition) using NOR-logic to limit the fan in and fan out to a maximum of two. These type cells either continue the comparison for bits of lesser significance if all two inputs are 0s, or terminate the comparison if a final decision can be made. For $0 \leq m \leq N/2 - 1$, there is a total of N/2 this type of cell, all functioning in parallel. N is supposed number of N bit.

$$C_{2,m} = \overline{D_{2n} + D_{2n+1}} \quad (10)$$

Set 3 consists of AND type cells. This cell has two inputs and one output as a termination flag for set 4. As this cell is AND gate, it purely sensitive to '0'. If any one of input is 0 it gives termination flag to stop process for the set 4. This set will differentiate for MSB from LSB. If in inputs bits MSB shows inequality. The comparison is terminated, set 3 signals set 4 to set the left bus and right bus bits to 0 for all bits of lower Significance. For $0 \leq m \leq N/2 - 1$, there is a total of N/2 this type of cells per level, with cell function and number of levels as.

$$C_{3,m} = C_{2,m} * C_{3,m+1} \quad (11)$$

Levels set 3 = ($\lceil \log_2(N) \rceil$).

Set 4 consists of final decision type cells according to inputs from set 1, previous set1 and previous set4 and produces the select inputs of 2-input multiplexer in set 5, which in turn drive both the left bus and the right bus. For this type cell and the 2b partition to which the cell belongs, bitwise comparison outcomes from set 1 provide information about the more significant bits in the cell's set 5, that compute ($0 \leq k \leq N - 1$)

$$S = C_{3, \lfloor k/2 \rfloor - 1} D_k \prod_{i=2, \lfloor k/2 \rfloor - 1}^{k-1} \bar{D}_i \quad (12)$$

The number of inputs in this set increases from left to

right in each partition, ending with a fan in of five. Thus, set 5 in set 4 determine whether set 5 propagates the bitwise comparison. Set 5 consists of N type MUX (two-input, 2-b-wide multiplexers). One input is (A_k, B_k) and the other is hardwired to "00." The select control input is output from

set 4, the 2-b as the left-bit code (A_k) and the right-bit code (B_k), where all left-bit codes and all right-bit codes combine to form the left bus and the right bus, respectively. This cells compute (where 0 ≤ k ≤ N-1).

$$Y_{0,1} = Y_k \times M_k + \bar{Y}_k \times 00 \quad (13)$$

The output Y_{0,1} denotes the "greater-than," "less-than," or equal to. Essentially, the 2-b code Y_{0,1} can be realized by OR-ing all left bits and all right bits separately, as shown in the decision module, using an OR-gate network in the form of NOR-NAND gates yielding a more optimum gate structure

IV. Schematic view of all sets and circuit of proposed comparator

In this section the schematic view all sets and 4-bit structure of comparator will be shown and discussed.

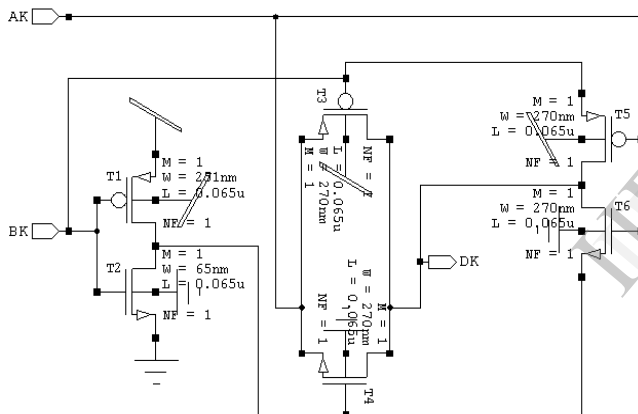


Fig. 3. Schematic view of XOR gate set 1 of comparator

Figure 3 shows the schematic view of first set i.e. XOR gate. This set contains 6 transistors instead of 12 used in previous comparator [12]. It has two inputs A_k, B_k and one output and provides inputs for set 2 and set 5. The relation between inputs and outputs are shown below.

$$D_k = A \text{ XOR } B$$

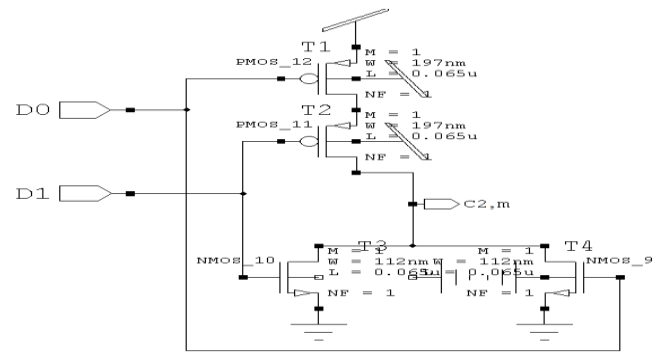


Fig. 4. Schematic view of NOR gate set 2 of comparator

Figure 4 shows the schematic view for set 2 i.e. NOR gate using CMOS logic. This set contains 4 transistors. It has two inputs they are name as D0, D1 and one output provides inputs for set 4. Output of this circuit is

$$C_{2,m} = \overline{D0 + D1}$$

This output is activated when any of input is 1 from set 1. It has maximum of 2 fan in and maximum 2 fan out.

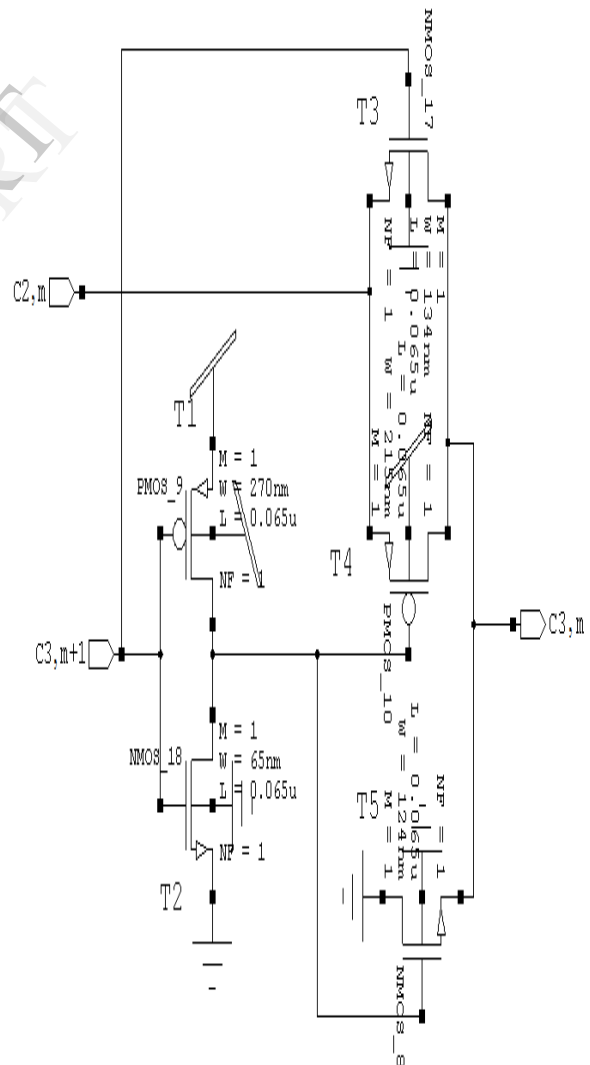


Fig. 5. Schematic view of AND gate set 3 of comparator

Figure 5 shows the schematic view of set 3. It uses AND gate logic using PASS transistor logic of comparator. This set contains 5 transistors. It has two inputs first is $C_{2,m}$ and second is $C_{3,m+1}$. First input $C_{2,m}$ is from set 2 and 'm' consider the number of set 2 is using. This circuit produces one output i.e. $C_{3,m}$. The relation between inputs and outputs are shown below

$$C_{3,m} = C_{3,m+1} \cdot C_{2,m}$$

It has 2 fan in and 3 maximum fan out.

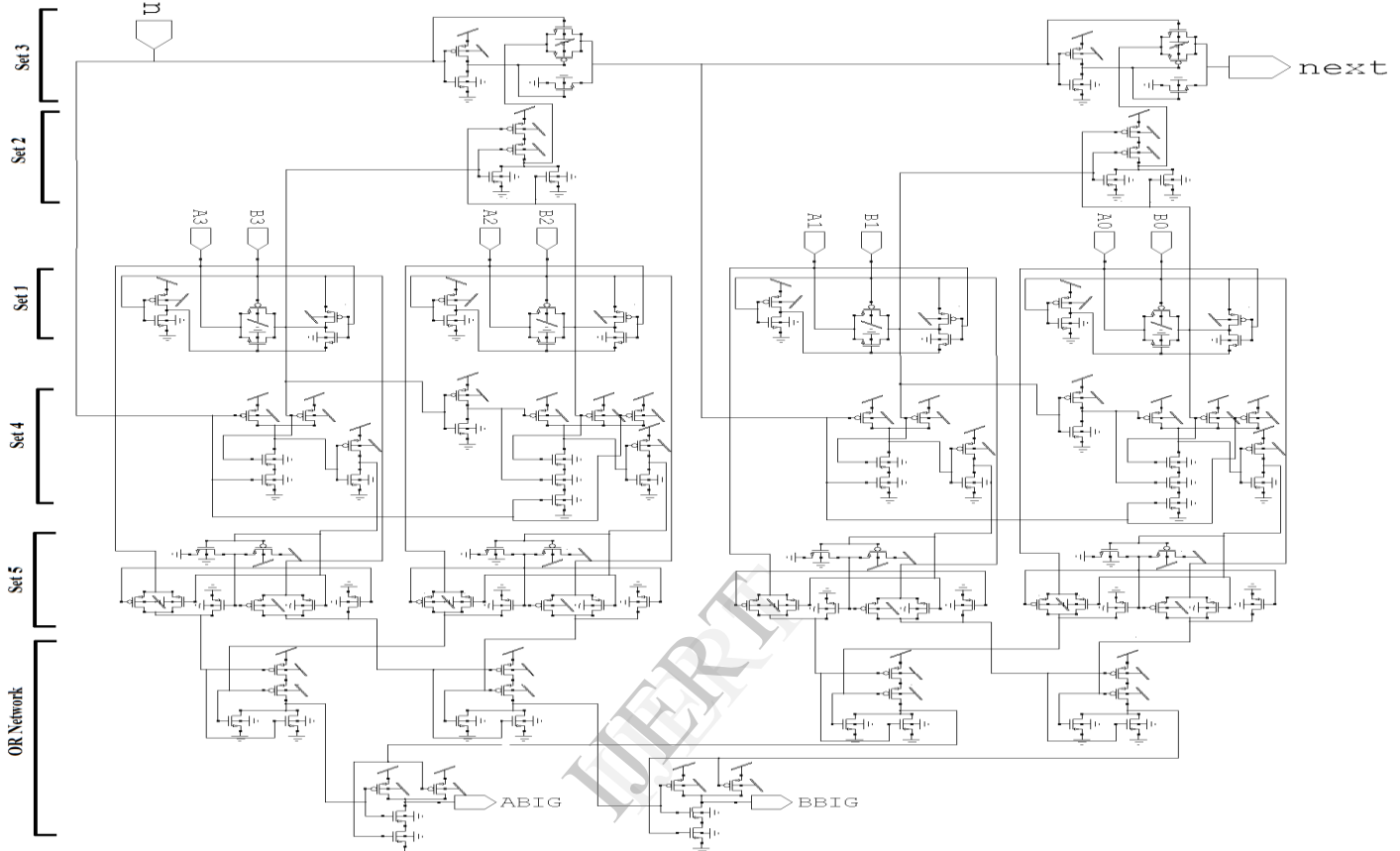


Fig. 6. Schematic view of second proposed 4-bit comparator

Figure 6 shows the schematic view of second proposed 4-bit comparator. 4-bit comparator requires 130 transistors to complete circuit. This circuit is shown for better understanding of architecture of circuit. Further it can be repeated again and again for more bit comparator. Figure 7 shows the schematic view of set 4 it uses NAND gate and inverters to invert the inputs using CMOS logic. This set contains maximum 10 transistors. It has maximum three one of them is inverted inputs named as D_0 , D_1 , $C_{3,m}$ and one output S . Output of circuit is described below

$$Y_0 = S \times A_k + \bar{S} \times 00 \text{ and}$$

$$Y_1 = S \times B_k + \bar{S} \times 00$$

$$S = D_0 + \bar{D}_1 + C_{3,m}$$

The D_1 input is taken from previous set 1, D_0 from current set 1 related to present input bits and $C_{3,m}$ from previous set 4. This set provides input to set 5 as select line for two 2:1 MUX. This circuit has maximum 3 fan in and 1 fan out. Figure 8 shows the schematic view of set 5 i.e. two 2:1 MUX using CMOS logic of comparator. This set contains 8 transistors. It has three inputs A_k , B_k , one select line i.e. from set four and one output which provide output for decision blocks.

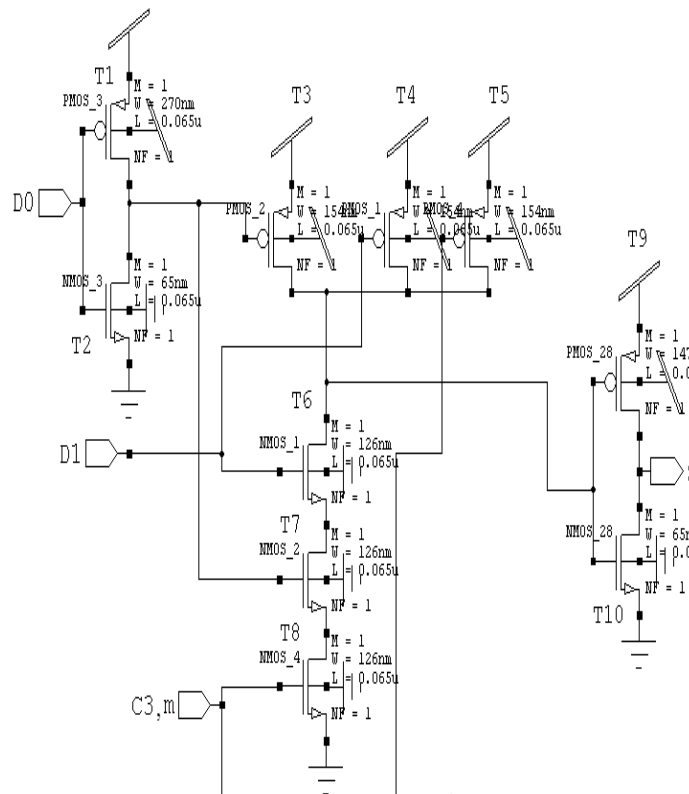


Fig. 7. Schematic view of set 4 of comparator

TABLE III
COMPARISON OF SIMULATION RESULTS OF IMPLEMENTED 64-BIT COMPARATOR DESIGNS.

SR. NO.	Comparator type	VDD (V)	Tech. file	Avg. power Consumption (μ w)	Delay (ps)	Transistor count
1	P.chuang et al. [7]	0.7	65nm	3.39	632	1206
2	S.A Hafeez et al. [12]	0.7	65nm	5.43	461	2864
3	Proposed work	0.7	65nm	2.37	231	2204

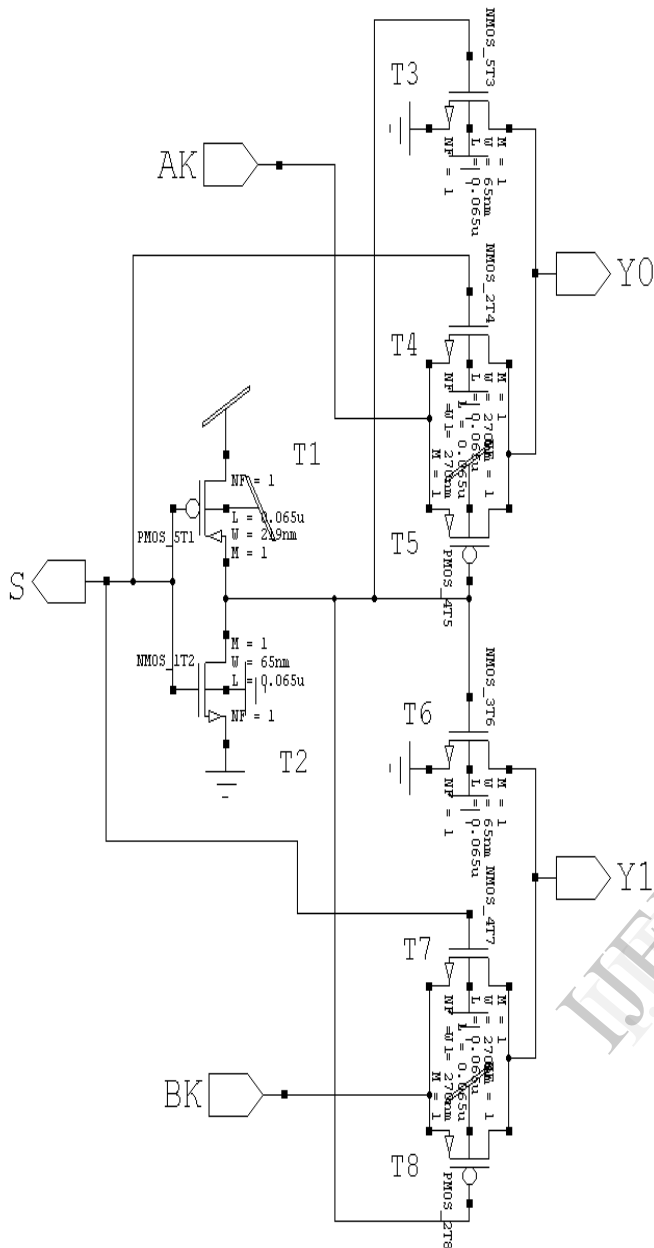


Fig. 8. Schematic view of Set 5 MUX.

This circuit has three PMOS transistors and five NMOS transistors at different channel width. The channel length of all transistors are kept 65nm.

V. Simulation results and waveform

All simulations are done by keeping same channel length of mosfet i.e. 65nm and different channel width which depend on type of circuits as shown in previous schematic views. The output waveform of comparator with respective to initial inputs is shown in figure 9. The simulated results of proposed 64-bit comparator and two existed tree based 64-bit comparators are shown in table III. The simulation of all comparators is done on TANNER-EDA V13 software as a tool. The simulation environment for all implemented comparator is kept same as we use 65nm technology model file based parameter, same VDD supply i.e. 0.7v, same input bits pattern etc. Simulated results shows our comparators

better compare to other simulated comparators. The changes done on comparator makes comparator more efficient as due to change of grouping of bits from 4-bit to 2-bit make comparator more faster and reduce its complexity of architecture design and changes done in various sets reduce power consumption. The simulated results given in table III showed the operating speed of proposed 64-bit comparator is 231ps which is 62% and 49% less compare to implemented previous 64-bit comparators [7], [12] respectively and the average power consumption during operation of this 64-bit comparator is 2.37 μ w which is 30% and 56% less compare to implemented previous 64-bit comparators [7], [12] respectively.

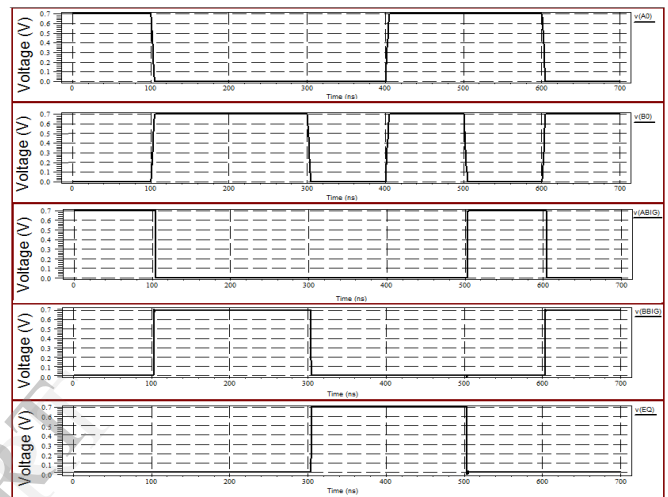


Fig. 9. Waveform of proposed 64-bit comparator

VI. CONCLUSION

A new 65nm parallel prefix tree based 64-bit comparator has been proposed. This design shows very good result and accurate outputs. The simulated results given in table III showed the operating speed of proposed 64-bit comparator is 231ps which is 62% and 49% less compare to implemented previous 64-bit comparators [7], [12] respectively and the average power consumption during operation of this 64-bit comparator is 2.37 μ w which is 30% and 56% less compare to implemented previous 64-bit comparators [7], [12] respectively. This comparator is very simple and easy to design and can be used for electronic circuits for better overall performance.

REFERENCES

- [1] J. L. Hennessy and D. A. Patterson, Computer Architecture a Quantitative Approach. Morgan Kaufmann Publishers, 2002.
- [2] J. Eyre and J. Bier, "DSP processors hit the mainstream," IEEE Computer, pp. 51–59, 1999.
- [3] C. C. Wang, C. F. Wu, K. C. Tsai, "1 GHz 64-bit high-speed comparator using ANT dynamic logic with two-phase clocking" IEE Proc. Computers and Digital Techniques, Volume 145, Issue 6, Nov.1998, pp:433 – 436.
- [4] C. H. Huang, J. S. Wang, "High-performance and power-efficient CMOS comparators" IEEE Journal of Solid-State Circuits, Volume 38, Issue 2, Feb. 2003, pp.:254 – 262.

- [5] H. M. Lam, C. Y. Tsui, "High-performance single clock cycle CMOS comparator" IEE Electronics Letters, Volume 42, Issue 2, Jan. 2006, pp:75 – 77.
- [6] S. Perri, P. Corsonello, "Fast Low-Cost Implementation of Single- Clock-Cycle Binary Comparator" IEEE Transactions on Circuits and Systems II: Express Briefs, Volume 55, Issue 12, Dec. 2008, pp:1239 – 1243.
- [7] Pierce Chuang, David Li, and Manoj Sachdev, Fellow, IEEE "A Low-Power High-Performance Single-Cycle Tree -Based 64-Bit Binary Comparator" IEEE Transactions On Circuits and Systems—II: Express Briefs, Vol.59, No. 2,February 2012.
- [8] S. Perri and P. Corsonello, "Fast low-cost implementation of single-clockcycle binary comparator," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 12, pp. 1239–1243, Dec. 2008.
- [9] F. Frustaci, S. Perri, M. Lanuzza, and P. Corsonello, "A new low-power high-speed single-clock-cycle binary comparator," in *Proc. IEEE Int.Symp. Circuits Syst.*, 2010, pp. 317–320.
- [10] K. Furuya, "Design methodologies of comparators based on parallel hardware algorithms," in *Proc. ISCIT*, 2010, pp. 591–596.
- [11] N. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective.*, 3rd ed. Reading, MA: Addison-Wesley, May 2004.
- [12] S. A. Hafeez, A. G. Ross and B. Parhami, Fellow, IEEE "Scalable Digital CMOS Comparator Using a Parallel Prefix Tree" IEEE Transaction on Very Large Scale Integration (VLSI) Systems September 13, 2012.



Deepak Sharma is currently pursuing M-Tech degree in "Electronics & Communication Engineering" at Department of E&CE of Yadavindra College of Engineering, Talwandi Sabo. His research interests are in high speed and low power digital VLSI circuit.



Er. Parminder Singh Jassal received the M-Tech degree in VLSI Design from C-DAC, Mohali Punjab Technical University in 2006. He is currently working as Assistant Professor in Department of Electronics and Communication Engineering at Yadavindra College of Engineering, Punjabi University Guru kashi Campus, Talwandi Sabo. He was previously working at NIT Hamirpur as a VLSI faculty under special Manpower development in VLSI Area (SMDP-II). His main research interests are in High speed digital VLSI circuits, Low power, synthesis and Simulation of digital circuits and FPGA Implementation.