

Design of Parallel Prefix Tree Structure for CMOS Comparator

N.Gurusakthi, B.Thamizhkani, S.Niranchana Devi
Dhanalakshmi Srinivasan College of Engineering and Technology
Chennai, India

Abstract

This paper proposes a comparator sketch using conventional digital CMOS cells featuring wide-range and high-speed operation. The Comparison is most primary arithmetic operation that determines whether or not one variety is higher than, less than or equal to the other number. Our comparator uses a novel scalable parallel prefix structure that leverages the comparison effect of the MSB, intending bitwise toward LSB only when the evaluation bits are equal. This comparator is composed of locally interconnected CMOS gates with a most fan-in of five and fan-out of four, unbiased of comparator bandwidth.

Comparator is most imperative thing that performs evaluation operation. Comparison between modified and existing 8-bit binary comparator designs is calculated by means of simulation carried out at 90nm technology in DSCHEM, Micro wind Tool. The major blessings of our proposed format are excessive velocity and electricity efficiency, maintained over a vast range.

Index Terms— wide-bit comparator, parallel prefix tree structure, High-speed arithmetic, high-speed.

I. INTRODUCTION

Comparator is a simple arithmetic unit that compares the magnitude of two binary numbers, say A and B, and produces output bits: $A > B$ or A vital data-path factor for any general motive architecture as nicely as an crucial device for application-specific and signal processing architectures. Comparators are also used in sorting networks which play a vital role in areas such as parallel computing, multi-access memories and multiprocessing. Comparator varieties a vital element of processors and digital systems. For processors, in order to acquire excessive throughput with quick clock rates, it is imperative that such gadgets have less delay. Consequently, the designing of excessive velocity comparator structure turns into a applicable and vital lookup topic. Previously posted comparator implementations having serial and parallel structure can each be observed in literature. The serial architecture is appropriate for short inputs (i.e. when both the inputs have lesser quantity of bits). For longer inputs (say, 32 bit, sixty four bit inputs), the circuit complexity and the combinational prolong enlarge drastically. As a result, parallel approach is commonly desired for comparators with longer inputs. The comparator designs introduced in this paper are primarily based on parallel approach.

II. COMPARATOR ARCHITECTURAL OVERVIEW

The contrast resolution module in Fig. 1 (which depicts the high-level structure of our proposed design) is a novel MSB-to-LSB parallel-prefix tree shape that performs bitwise comparison of

two N-bit operands A and B, denoted as $A_{N-1}, A_{N-2}, \dots, A_0$ and $B_{N-1}, B_{N-2}, \dots, B_0$, where the subscripts vary from N-1 for the MSB to zero for the LSB. The evaluation resolution module performs the bitwise contrast asynchronously from left to right, such that the evaluation logic's computation is caused solely if all bits of increased significance are equal. The parallel shape encodes the bitwise evaluation results into two N-bit buses, the left bus and the right bus, every of which keep the partial contrast result as every bit function is evaluated, such that

if $A_k > B_k$, then left ok = 1 and right okay = 0

if $A_k < B_k$, then left okay = 0 and proper ok = 1

if $A_k = B_k$, then left okay = zero and proper k = 0

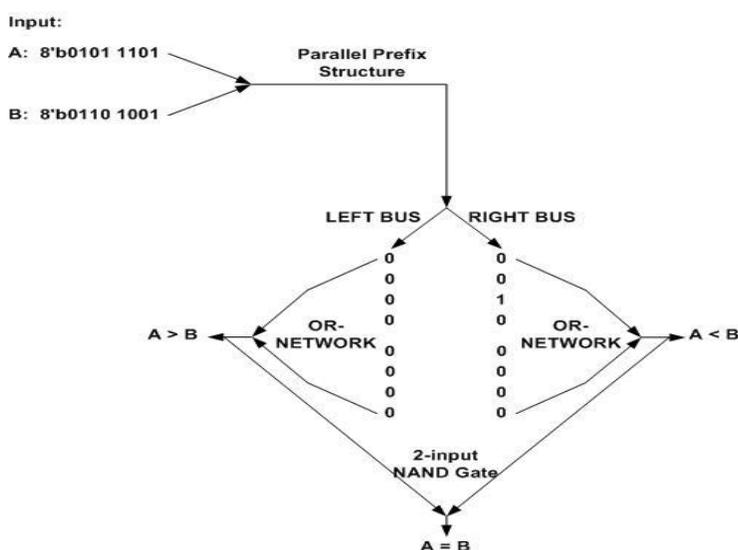


Fig. 2. Example 8-b comparison

In addition, to decrease switching activities, as quickly as a bitwise contrast is not equal, the bitwise evaluation of each bit of decrease value is terminated and all such positions are set to zero on both buses, thus, there is never extra than one excessive bit on both bus. The choice module uses two OR-networks to output the

final evaluation choice primarily 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 bit). If LR = 00, then A = B, if LR = 10 then $A > B$, if LR = 01 then $A < B$, and LR = eleven is now not possible.

An 8-bit comparison of input operands $A = 01011101$ and $B = 01101001$ is exemplified in Figure 2. In the first step, a parallel prefix tree structure generates the encoded data on the left bus and right bus for every pair of corresponding bits from A and B. In this example, A_7

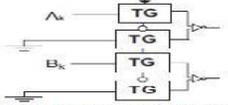
$= 0$ and $B_7 = 0$ encodes as $left_7 = right_7 = 0$, $A_6 = 1$, and $B_6 = 1$ encodes as $left_6 = right_6 = 0$, and $A_5 = 0$ and $B_5 = 1$ encodes $left_5 = 0$ and $right_5 = 1$. At this point, for the reason that the bits are unequal, the contrast terminates and a closing evaluation choice can be made based on

the first three bits evaluated. The parallel prefix structure forces all bits of lesser value on each bus to 0, regardless of the remaining bit values in the operands. In the 2nd step, the OR-networks perform the bus OR-scans, ensuing in zero and 1, respectively, and the ultimate comparison selection is $A \geq B$. We partition the shape into 5 hierarchical prefixing sets, as depicted in Fig. 3, with the related image representations in Tables I and II, where each set performs a unique characteristic whose output serves as input to the next set, till the fifth set produces the output on the left bus and the proper bus.

TABLE I - SYMBOL NOTATION AND DEFINITIONS

Symbol (Cells)	Definition
N	Operand bitwidth
A	First input operand
B	Second input operand
R	Right bus result bit
L	Left bus result bit

TABLE 2 - LOGIC GATE DEPICTION FOR SYMBOLS USED IN FIG. 3

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

TG: Transmission Gate

All cells (components) within each set function in parallel, which is a key feature to enlarge operating velocity whilst minimizing the transitions to a minimal set of leftmost bits wished for a correct decision. This prefixing set structure bounds the components' fan-in and fan-out regardless of comparator bitwidth and eliminates heavily loaded world indicators with parasitic components, accordingly enhancing the working pace and decreasing electricity consumption. Additionally, the OR-network's fan-in and fan-out is restrained by means of partitioning the buses into 4-b groupings of the enter operands, accordingly decreasing the capacitive load of each bus.

III. COMPARATOR DESIGN DETAILS.

In this section, we detail our comparator's sketch (Fig. 3), which is notably based on using a novel parallel prefix tree (Tables I and II include symbols and definitions). Each set or crew of

cells produces outputs that serve as inputs to the subsequent set in the hierarchy, with the exception of set 1, whose outputs serve as inputs to quite a few units

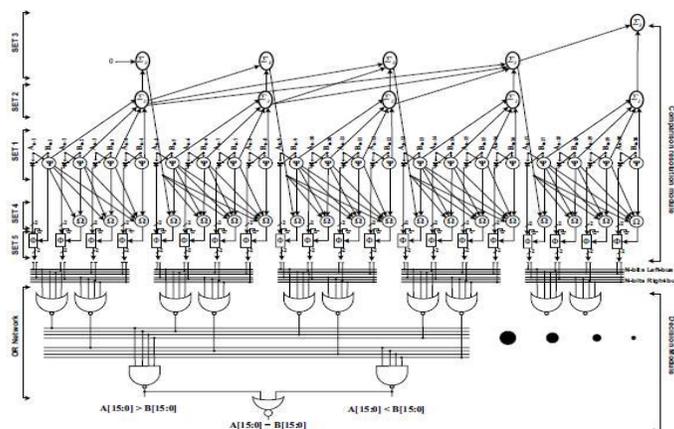


Fig. 3. Implementation details for the comparison resolution module (sets 1 through 5) and the decision module.

Set 1 compares the N-bit operands A and B bit-by-bit, the usage of a single degree of N -type cells. The N-type cells supply a termination flag D_k to cells in sets 2 and 4, indicating whether the computation ought to terminate.

Set 2 consists of $\Sigma 2$ -type cells, which mix the termination flags for every of the four Ψ -type cells from set 1 (each $\Sigma 2$ -type phone combines the termination flags of one 4-b partition) the use of NOR-logic to limit the fan-in and fan-out to a most of four. The $\Sigma 2$ -type cells both continue the contrast for bits of lesser magnitude if all 4 inputs are 0s, or terminate the evaluation if a final decision can be made.

Set 3 has $\Sigma 3$ -type cells, which are similar to

$\Sigma 2$ -type cells, however can have extra common sense levels, specific inputs, and lift unique triggering points. A $\Sigma 3$ -type telephone gives no assessment functionality; the cell's sole cause is to limit the fan-in and fan-out regardless of operand bit width. To restrict the $\Sigma 3$ -type cell's nearby interconnect to four, the variety of stages in set 3 increases if the fan-in exceeds four. Set 3 offers functionality comparable to set 2 the usage of the same NOR good judgment to proceed or terminate the bitwise contrast activity. If the comparison is terminated, set three alerts set four to set the left bus and proper bus bits to 0 for all bits of decrease significance.

Set 4 consists of Ω -type cells, whose outputs control the pick out inputs of Ω -type cells (two-input multiplexors) in set 5, which in flip drive each the left bus and the right bus. For an Ω -type phone and the 4-b partition to which the mobile belongs, bitwise evaluation results from set 1 provide facts about the greater great bits in the cell's Ω kind cells.

The range of inputs in the Ω -type cells increases from left to proper in each partition, ending with a fan-in of five. Thus, the Ω type cells in set 4 determine whether or not set 5 propagates the bitwise comparison codes. Table III indicates a sample 16-b assessment to make clear (5) using(1)–(4). Set 5 encompasses of N Ω -type cells (two-input, 2-b-wide multiplexers). One input is (A_k, B_k) and the other is hardwired to "00." The pick out manage enter is primarily

based on the Ω -type mobilephone output set four We outline the 2-b as the left-bit code (A_k) and the right-bit code (B_k), the place all left-bit codes and all right-bit codes mix to form the left bus and the proper bus, respectively. The output $F_{1,0}$ okay denotes the “greater-than,” “less-than,” or “equal to” last evaluation selection Essentially, the 2-b code $F_{1,0}$ ok can be realized by using OR-ing all left bits and all proper bits one after the other as proven in fig2.

IV. PROPOSED 8 BIT COMPARATOR:

In this section, The Proposed comparator design is identical as the selection module. The regulate the comparison resolution module as proven in fig.4

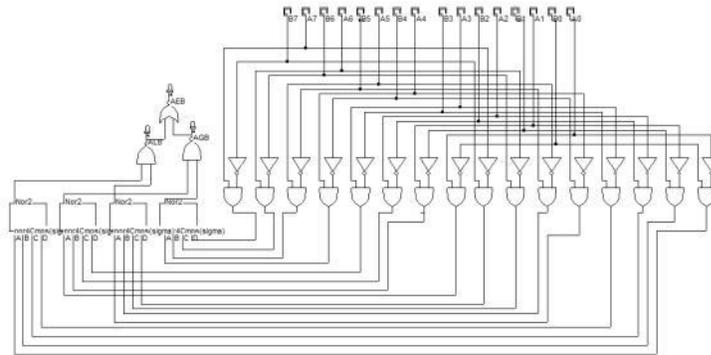


Fig. 4 Design of projected 8 Bit Comparator using

V. SIMULATION RESULTS

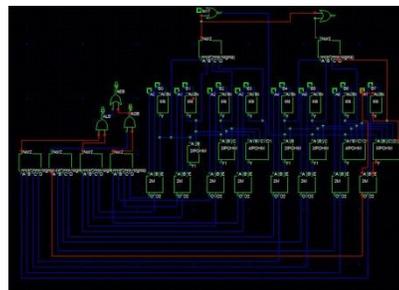


Figure 5.1 Design of 8 Bit Comparator Using a Parallel Prefix Tree using in DSCH Tool.

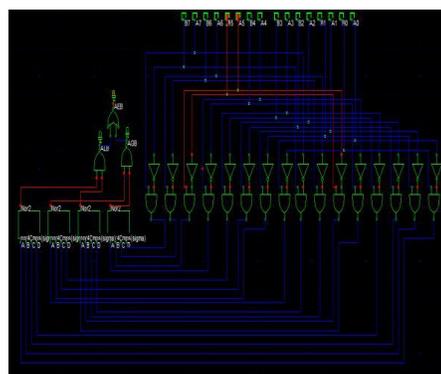


Figure 5.2 Design of Proposed 8 Bit Comparator using DSCH Tool

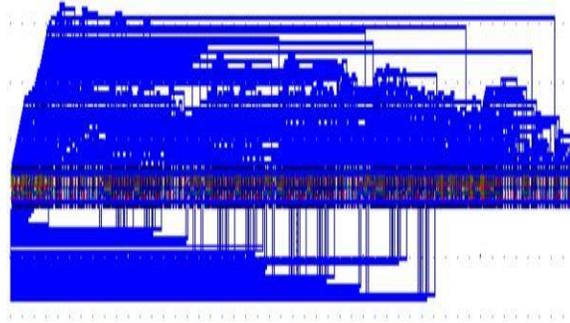


Figure 5.3 Design of 8 Bit Comparator by means of a Parallel Prefix Tree layout via in Microwind Tool in 90nm.

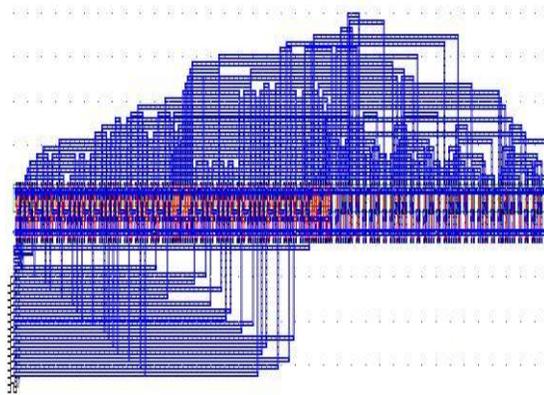


Figure 5.4 Design of projected 8 Bit Comparator Using a layout using in Microwind Tool in 90nm.

**TABLE III- COMPARATOR WITH 8 BITS AT
 DIFFERENT TECHNOLOGY**

	Scalable 8 bit Comparator			Proposed 8 bit Comparator		
	0.18 μ m 1.95 V	0.120 μ m 1.95 V	90 μ m 1.95 V	0.18 μ m 1.95 V	0.120 μ m 1.95 V	90 μ m 1.95V
Area	38211	8796.5	6108.7	12322	2838.8	1971.4
Power	1.342 mW	0.193 mW	93.67 μ W	0.217 mW	20.633 μ W	16.716 μ W

V. CONCLUSION

In this paper, we introduced a scalable high-speed low-power comparator the use of normal digital hardware buildings consisting ABDEL-HAFEEZ et al.:

SCALABLE DIGITAL of two modules: the comparison decision module and the selection module. These modules are structured as parallel prefix bushes with repeated cells in the form of simple ranges that are one gate level deep with a most fan-in of 5 and fanout of four, independent of the input bit width. This regularity permits easy prediction of comparator characteristics for arbitrary bit widths and is eye-catching for endured technology scaling and logic synthesis.

VI. REFERENCES

- [1] H. J. R. Liu and H. Yao, High-Performance VLSI Signal Processing Innovative Architectures and Algorithms, vol. 2. Piscataway, NJ: IEEE Press, 1998.
- [2] Chandrasekhar, S., and J. Lakshmi Prasanna. "128 bit Parallel Prefix Tree Structure Comparator." International Journal of Research in Engineering and Science (IJRES) 3 (2015).
- [3] Suzuki, Hiroaki, Chris H. Kim, and Kaushik Roy. "Fast tag comparator using diode partitioned domino for 64-bit microprocessors." IEEE Transactions on Circuits and Systems I: Regular Papers 54.2 (2007): 322-328.
- [4] Suzuki, Hiroaki, Chris H. Kim, and Kaushik Roy. "Fast tag comparator using diode partitioned domino for 64-bit microprocessors." IEEE Transactions on Circuits and Systems I: Regular Papers 54.2 (2007): 322-328.
- [5] Peiravi, Ali, and Mohammad Asyaei. "Robust low leakage controlled keeper by current-comparison domino for wide fan-in gates." Integration, the VLSI journal 45.1 (2012): 22-32.
- [6] Peiravi, Ali, and Mohammad Asyaei. "Current-comparison-based domino: New low-leakage high-speed domino circuit for wide fan-in gates." IEEE transactions on very large scale integration (VLSI) systems 5 (2013): 934-943.
- [7] Chan, Antonio H., and Gordon W. Roberts. "A jitter characterization system using a component-invariant vernier delay line." IEEE Transactions on Very Large Scale Integration (VLSI) Systems 12.1 (2004): 79-95.
- [8] Xia, Tian, et al. "Self-refereed on-chip jitter measurement circuit using Vernier oscillators." VLSI, 2005. Proceedings. IEEE Computer Society Annual Symposium on. IEEE, 2005.