CMOS based Power Efficient Digital Comparator with Parallel Prefix Tree Structure

J. Lakshmi Prasanna1, * V. Sahiti2, E. Raghuveera3, M. Ravi Kumar4

1,2,3,4 Department of Electronics and Communications Engineering
1,2,3,4 Koneru Lakshmaiah Education Foundation Deemed to be University, Greenfields, Vaddeswaram,
1,2,3,4 Guntur, Andhra Pradesh, India - 522502
*Corresponding author E-mail: sahithi.vankalapati@gmail.com

Abstract

A 128-Bit Digital Comparator is designed with Digital Complementary Metal Oxide Semiconductor (CMOS) logic, with the use of Parallel Prefix Tree Structure [1] technique. The comparison is performed on Most Significant Bit (MSB) to the Least Significant Bit (LSB). The comparison for the lower order bits carried out only when the MSBs are equal. This technique results in Optimized Power consumption and improved speed of operation. To make the circuit regular, the design is made using only CMOS logic gates. Transmission gates were used in the existing design and are replaced with the simple AND gates. This 128-Bit comparator is designed using Cadence TSMC 0.18µm technology and optimized the Power dissipation to 0.28mW and with a Delay of 0.87µs.

Keywords: ANT Circuits, Fast Adders, Parallel Prefix Tree Structure, Priority Encoder and Transmission Gates.

1. Introduction

Comparator is a logic circuit that is used to compare the magnitude of two numbers. Comparators are the key design elements in many scientific and mathematical applications. It is used widely in scientific applications, test circuits and analysis of signatures etc. Previous comparators are designed using adder architectures. These architectures suffer from speed and area issues. Multiplexer based comparator architectures a headed the design of comparator architectures that make use of adders. The multiplexer based comparators divide the n-bit input into two n/2 bits and the result of two n/2 comparators is fed to the multiplexer that provide the result of the comparison with un-optimized power consumption [9]-[13]. The comparators are designed using All N Transistor (ANT) circuits, but all the NMOS transistors that are connected in series enter saturation mode during operation which increases over all conductive resistance[13]. Some uses priority encoder architectures. These architectures split the n-bit input into two n/2 bits and the result of two n/2 comparators is taken as input by the priority encoder so that it considers the MSB priority first[11]. In this work, uses Parallel Prefix Structure to develop architecture, the n bits are divided into n/4 modules each module compares 4 bits. The comparison is carried out from MSB to the LSB. The comparison is carried to further bits only when the MSBs are equal. The decision can be taken in the initial module and then the next modules will not perform comparison operation thereby saving the power. The 128-bit comparator is separated into two sub modules i.e., comparison module and decision module as shown in Fig. 1.1. The comparison is performed bit by bit using the comparison module of 128 bits. The input variables A and B are denoted as A127, A126, A125…………..A1, A0 and B127, B126…………B2, B1, B0. Bit wise comparison is advanced from MSB to LSB, so that the comparison is triggered only when the MSBs are equal. The comparison module encodes the comparison bits into two buses that are right bus and left bus such that each bus stores the intermediate result [2]. Each bit is compared such that

If

A0>B0 then right n=1 and left n = 0
A0>B0 then right n=0 and left n=1
A0=B0 then right n=0 and left n=0
The module which makes decision uses OR network to make a conclusion based on the bits that are stored in the bus.

A (128-1)     B (128-1)

Comparing

128-Bit Right Bus

Decision Making

A > B     A = B     A < B

128-Bit Left Bus

Fig.1.1 Block Diagram of 128-bit Comparator [2]

If

rl = 10     A < B
rl = 01     A > B
rl = 00     A = B

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.
2. Architecture

The comparator architecture comprises of the decision module and comparison module. The comparison module performs the comparison for the given input bits. As we are designing a 128-bit comparator the 128 bits are grouped into 32 groups of 4 bits and each four bits are compared in a single 4-bit module and we have 32 modules each comparing 4 individual bits. Each 4-bit comparator module takes two 4-bit input operands one signal from the previous comparator module that enables the comparison of that module. Each single 4-bit comparator module has 4 outputs that are used by the decision module for making decision at the output and one enable output that acts as enable input for the next comparator module that triggers the comparison. The decision module gets its input from the 4-bit comparison module. Each comparison module gives 4 outputs for 4 input bits. In this way, we get a total of 128 outputs from the comparison module each single output for 128 inputs. The distinct four inputs are combined to get a single output[14-22]. This procedure is tracked until we get our last 3 outputs. Each 4-bit comparator segment of the comparison module is again divided into five hierarchical sets that perform the comparison operation in a specified manner. We partition the comparator resolution module structure into five hierarchical prefixing sets.

3. Comparator Design

The comparison module relates each individual bit using a tree structure. In the comparison module, we utilize five Sets of elements

![Fig. 2.1. Architecture of 128-Bit Comparator [2]](image)

Set -1 performs basic comparison of two individual bits of A and B. The output of set-1 acts as the input of set-2. Four set-1 outputs are combined to give a decision regarding the individual four input bits by set-2 element. Set-3 gets the input from set-2 and set-4 gets the input from set-3. The output of set-4 element acts as enable input to set-5 element. The output of set-5 element forms left and right bus bits. The right and left bus bits from the comparison module are given to the decision module which performs operation and makes a final decision.

![Fig. 3.1: Tree structure of comparison module [2]](image)

Set 1 compare A and B bit wise. The set-1 elements provide a output D, to elements in set-2 and set-4 which is used to abort the operation, if it is high the operation is proceeded else the operation is aborted. These results in computing a XOR operation as represented in equation (1).

$$ D_i = A_i \oplus B_i $$  \hspace{1cm} (1) 

Set-2 combines the four outputs from the set-1 by using 4 input NOR gate. The set _2 type elements compares bits of lower order if all the inputs are logically low, otherwise aborts the operation if ultimate output can be determined

$$ S_{2,m} = \prod_{m,i=1,0}^{32,3} D_{m,i} $$  \hspace{1cm} (2) 

Set 3 consists of elements, which resembles set _2 type elements in their functionality but will have more logic levels. The set _3 type elements do not perform comparison. The elements prime function is to bound the fan-in and fan-out irrespective of number of bits in input. Set-3 makes use of AND gates to progress or abort the operation. If the operation is aborted, output from set-3 makes following elements to set the left bus bit to ‘0’ and right bus bit to ‘0’ for all lower order bits.

$$ S_{3,m} = \prod S_{2,m-1} S_{2,m-2} $$  \hspace{1cm} (3) 

Where

- m - module number
- S2, m - output of present set-2 module
- S2, m-1 - output of previous set-2 module

Output of set-4 controls the inputs of set-5 elements, which drives both the left and right bus

$$ S_{4,m} = \prod S_{3,m-1} D_1 $$  \hspace{1cm} (4)
\[ S_{4,m_2} = \prod S_{3,m-1} D_2 \overline{D}_1 \]  
\[ S_{4,m_3} = \prod S_{3,m-1} D_3 \overline{D}_2 \overline{D}_1 \]  
\[ S_{4,m_4} = \prod S_{3,m-1} D_4 D_2 \overline{D}_1 \]  

Set 5 accomplishes the task of a multiplexer. It shows whether the particular bit of ‘A’ is greater or whether the particular bit of ‘B’ is greater and provides 2-bit output. The selection input relies on the output of set-4. We describe the 2-b as the right bit and left bit rli. The output \( F \) denotes greater, less or equal as a final output.

In the existing design the functionality of multiplexer is designed using transmission gates such that the static power dissipation is reduced far better [1]. But whenever we consider the total power dissipation the static power dissipation accounts for very less amount that is it is in the order of micro watts whereas the dynamic power dissipation will be in the order of milli watts so decreasing the static power does not account much for the circuit instead it makes the design difficult and irregular as all the four sets are designed using gate level design and only the set 5 elements are designed using transistor level. It is not possible to design the entire circuit in gate level. Therefore, instead of using transmission gate as set-5 elements we go for a gate level design

\[ S_{5,m,A_i} = S_{4,m,A_i} \]  
\[ S_{5,m,B_i} = S_{4,m,B_i} \]  

4. Simulation results

Fig. 4.1 Result showing all possible conditions and states

When two input operand bits are equal then the comparison must be carried out for all the bits which results in a worst case activating all the cells. Fig 4.2 shows the output for the condition when all the bits for two operands are equal.

Fig. 3.2 Design of set-5 element [1]
When two input operand bits are not equal then the comparison must be carried out only for those bits until the decision can be made.

Here we have verified for the worst cases by making most significant bits of two input operands to be equal and only the least significant bit of two input operands is equal. This helps us to calculate the worst-case delay and power consumption by activating all the cells as mentioned in table 4.1. Fig 4.3, 4.4, 4.5 shows the output for all possible conditions

### 4.1 RTL Schematics

![Fig. 4.6 RTL for single stage](image)

### 4.2 Layout

![Fig. 4.7 Layout for 128-bit](image)

### 4.3 Power and Delay Results

<table>
<thead>
<tr>
<th>No of bits</th>
<th>Dynamic Power (mw)</th>
<th>Power Leakage (μw)</th>
<th>Delay (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>0.03</td>
<td>0.05</td>
<td>0.020</td>
</tr>
<tr>
<td>32</td>
<td>0.07</td>
<td>0.1</td>
<td>0.033</td>
</tr>
<tr>
<td>64</td>
<td>0.14</td>
<td>0.22</td>
<td>0.052</td>
</tr>
<tr>
<td>128</td>
<td>0.28</td>
<td>0.41</td>
<td>0.087</td>
</tr>
</tbody>
</table>
To evaluate the performance of comparator, we simulated the complete design with various inputs using the cadence with 0.18 μm-TSMC digital CMOS technology. The worst-case delay occurs when all the most significant bits are logic low and the least significant bit is logic high. Because in this condition almost all the cells are activated which does not uses the advantage of parallel tree structure. It helps in replicating the design that supports VLSI reconfigurable applications for binary comparators.

We have designed 128-bit digital cmos comparator using digital CMOS cells. This architecture consists of parallel structure which helps in replicating the design that supports VLSI reconfigurable topology. It yields a power efficient comparator structure when compared with the previous comparator structures that make use of fast adders and multiplexers. The comparator dissipates a power of 0.28 mw and has a delay of 0.087ms.

![Power Dissipation and Delay Results](image)

**Fig.4.7 Power dissipation and delay results**

Type 3 cells have a extreme fan-in of five and extreme fan-out of four. We analyzed the performance of comparator with different number of inputs and evaluated the leakage power and delay. As the number of bits upsurges the delay and power dissipation rise linearly as discussed in [1], [2], [3]. The results for our 8-b comparator, 16-b comparator, 32-b comparator, 64-b comparator, 128-b comparator and reported results and tabulated in table 4.1

**5. Conclusion**

We have designed 128-bit digital cmos comparator using digital cmos cells. This architecture consists of parallel structure which helps in replicating the design that supports VLSI reconfigurable topology. It yields a power efficient comparator structure when compared with the previous comparator structures that make use of fast adders and multiplexers. The comparator dissipates a power of 0.28 mw and has a delay of 0.087ms.

**References**

[10] TEXAS INSTRUMENTS SN7485 4-bit Magnitude Comparators.