MULTI DOMINO DOUBLE MANCHESTER CARRY CHAIN ADDERS FOR HIGH SPEED CIRCUITS

S. Alagubalakrishnan

PG Scholar, Department of VLSI Design,
Theni Kammavar Sangam College of Technology, Tamilnadu, (India)

ABSTRACT

The carry look-ahead adders are designed till now by using standard 4 bit Manchester carry chain. Due to its limited carry chain length, the carries of the adders are computed using 4 bit carry chain. This leads to slow down the operation. A high speed 8 bit (MCC) adder in multi output domino CMOS logic is designed in this work. Due to its limited carry chain length this high speed MCC uses 2 separate 4-bit MCC. The 2 MCC namely odd carry chain and even carry chain are computed in parallel to increase the speed of the operation. This technique has been applied for the design of 8 bit adders in multi output domino logic and the simulation results are verified. Results prove that 8bit MCC produces less delay compared to conventional 4 bit delay. The reduced delay realizes better speed compared to the conventional designs. The existing design and the previous designs are designed and simulated using TANNER EDA. The delay of these designs is compared with 8 bit with 130 nm technology file. Implementation results reveal that the high speed comparator has delay of 41.64% less compared to the conventional designs.

Keywords: Carry Look-Ahead (CLA) Adders, Manchester Carry Chain, Multioutput Domino Logic.

I. INTRODUCTION

Addition is the most commonly used arithmetic operation and also the speed-limiting element to make faster VLSI processors. As the demand for higher performance processors grows, there is a continuing need to improve the performance of arithmetic units and to increase their functionality. High-speed adder architectures include the carry look-ahead (CLA) adders, carry-skip adders, carry-select adders, conditional sum adders, and combinations of these structures. High-speed adders based on the CLA principle remain dominant, since the carry delay can be improved by calculating each stage in parallel.

II. FOUR BIT MCC

The 4-bit MCC is mainly used to reduce to computation time. The 4-bit MCC can perform the operation of 16 bit CLA. The MCC is mainly used to reduce the number of transistor count by using shared logic.

2.1 Domino Implementation for the Generate
The generate signal implemented in domino logic is shown in Figure 1. It consists of two inputs namely $a_i$ and $b_i$ and has one output $g_i$. The two inputs are connected in series thus perform AND operation. The operation of the circuit is controlled by clock signal. If the clock signal goes to value ‘0’, then the circuit will enter into precharge state and pmos will get connected to ground and output will maintain the value of 0. If the clock makes the transition from 0 to 1 then the circuit will enter into evaluation state and the output depends on the input value. Since generate signal possess AND operation.

![Fig. 1: Domino Implementation For the Generate](image)

### 2.2 Domino Implementation for XOR Propogate

The propagate signal implemented in domino logic is shown in Figure 2. Here the propagate signal is implemented in XOR operation. The propagate circuit is controlled by clock signal. If clk goes to ‘0’, then the circuit will enter into precharge state and the output remains in 0 value. If clk value is 1, then the output value depends on input value. Since this propagate signal is XOR operation based if both the inputs are different then output $p_i$ will maintain the value 1 else $p_i$ will have value 0.

![Fig. 2: Domino Implementation for XOR Propogate](image)
2.3 Domino Implementation For OR Propagate

The propagate signal implemented in domino logic is shown in Figure 3. It consists of two inputs \( a_i \) and \( b_i \) and consists of one output signal \( t_i \). Here the propagate signal is implemented in OR operation. The propagate circuit is controlled by clock signal. If clk goes to ‘0’, then the circuit will enter into precharge state and the output remains in 0 value. If clk value is 1, then the output value depends on input value. Since this propagate signal is OR operation based if any one of the inputs is 1, then output \( p_i \) will maintain the value 1 else \( p_i \) will have value 0.

![Fig. 3: Domino Implementation for OR Propagate](image)

2.4 Conventional Four Bit MCC

Let \( A = a_{n-1}a_{n-2} \cdots a_1a_0 \) and \( B = b_{n-1}b_{n-2} \cdots b_1b_0 \) represent two binary numbers to be added and \( S = s_{n-1}s_{n-2} \cdots \). The computation of the carry signals is based on

\[
c_i = g_i + z_i \cdot c_{i-1}
\]  

(A)

where \( g_i, b_i, \) and \( z_i \) are the carry generate and the carry propagate terms, respectively

\[
c_i = g_i + z_i \cdot c_{i-1} + z_i \cdot c_{i-2} + \cdots + z_i \cdot c_i + 1 \quad \text{...}\quad c_{i-1}.
\]  

(B)

This conventional circuit consists of 4 bit two inputs namely \( p_0, p_1, p_2, p_3 \) and \( g_0, g_1, g_2, g_3 \). The operation of the circuit is controlled by clock signal. The input values are get from \( p_i \) and \( g_i \) values of the domino propagate and generate output values. If clock equals to ‘0’, the circuit will enter into precharge state and no output will be obtained. If clock value is ‘1’, then the output will depend on the input values. The inputs of propagate and generate signals from \( p_i \) and \( g_i \) will possess and the corresponding output carry signals namely \( c_0, c_1, c_2, c_3 \).
III. EIGHT BIT MCC

The 8-bit MCC is mainly used to reduce the delay by increasing the speed. Here two 4-bit MCC is used and the carries are generate in parallel simultaneously. The use of the 8 bit adder as a basic block, instead of 4 bit MCC adder, can lead to high speed adder implementations. The derived here carry equations are similar to those for Lin's carries equation. The derived carry equations allow the even carries separately of the odd ones. Implementation of the carries by two independent 4bit carries chains one chain computes the even carries, while the other chain computes the odd carries.

3.1 Implementation of Carry

Domino Logic implementation of Carry Signals consists of two signals namely carry generate signal and carry propagate signals respectively. The Implementation of generate and propagate signals using domino logic
3.2 Even Carry Computation

This carry chain gets computed when input value has even values. Say i= 0,2,4,6. For the even input values say p0, p2, p4, p6 and g0, g2, g4, g6 the corresponding intermediate even carries say h0, h2, h4, h6 is obtained. The input values of propagate and generate signals are obtained from pi and gi respectively.

The even carries can be analytically given by

\[ H_2 = g_2 + p_2g_0 \]  
\[ H_4 = g_4 + p_4g_2 + p_4p_2g_0 \]  
\[ H_6 = g_6 + p_6g_4 + p_6p_4g_2 + p_6p_4p_2g_0. \]

![Fig. 6: Even Carry Computation](image)

3.3 Odd Carry Computation

This carry chain gets computed when input value has odd values. Say i= 1,3,5,7. For the odd input values say p1, p3, p5, p7 and g1, g3, g5, g7 the corresponding intermediate odd carries say h1, h3, h5, h7 is obtained. The input values of propagate and generate signals are obtained from pi and gi respectively.

The odd carries can be analytically given by

\[ H_1 = g_1 + p_i .c_{i-1}(4) \]  
\[ H_3 = g_3 + p_3g_1 + p_3p_1c_{-1}(5) \]  
\[ H_5 = g_5 + p_5g_3 + p_5p_3g_1 + p_5p_3p_1g_0. (6) \]  
\[ H_7 = g_7 + p_7g_5 + p_7p_5g_3 + p_7p_5p_3g_1 + p_7p_5p_3p_1c_{-1}. (7) \]
Fig. 7: Odd Carry Computation

3.4 Sum Bit Implementation

The sum has mux to selects one of several analog or digital input signals and forwards the selected input into a single line. A multiplexer of $2^n$ inputs has $n$ select lines, which are used to select which input line to send to the output. A multiplexer is also called a data selector. When $hi-1=0$ it selects $pi$ then $hi-1=1$ it selects $pi@ti-1$.

Fig. 8: Sum Bit Implementation

Fig. 9: Schematic Diagram for Eight Bit MCC
IV. SIMULATION RESULTS

Fig. 10: OUTPUT FOR 4-BIT MCC

Fig. 11: OUTPUT FOR 8-BIT MCC

Fig. 12: DELAY IN 4-BIT MCC

Fig. 13: DELAY IN 8-BIT MCC

4.1 Delay Reduction

<table>
<thead>
<tr>
<th></th>
<th>Existing (4-bit MCC)</th>
<th>Proposed(8-bit MCC)</th>
<th>Delay percentage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>9.74(ns)</td>
<td>4.03(ns)</td>
<td>41.64%</td>
</tr>
</tbody>
</table>

Table 4.1 DELAY ANALYSIS
Delay reduction % = (9.74 - 4.03)/ 13.71 * 100 = 41.64%

V. CONCLUSION

An 8-bit adder is designed using Manchester carry chain. This circuit is designed and simulated using TANNER TOOLS software. This design realizes better improvement in reducing the delay by introducing parallelism concept in carry chains. To increase the speed of the operation by using two independent carry chain in parallel and thus reduces the time delay of the operation its performance is analysed by using 130nm with the supply voltage 1.3v, 1.2v respectively. Thus, the proposed 8-bit Manchester carry chain is superior compared to 4-bit Manchester carry chain circuit. As a further work reducing the area of this chain and further reducing the delay by analyzing this design in submicron technology and implementing it in a variable bits like 16 bit, 32 bit Manchester Carry Chain in multi output domino CMOS logic can be considered.

VI. NOMENCLATURE

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLA</td>
<td>CARRY LOOK AHED ADDER</td>
</tr>
<tr>
<td>MCC</td>
<td>MANCHESTER CARRY CHAIN</td>
</tr>
<tr>
<td>CMOS</td>
<td>COMPLEMENTARY METAL OXIDE SEMICONDUCTOR</td>
</tr>
<tr>
<td>MOSFET</td>
<td>METAL OXIDE SEMICONDUCTOR FIELD EFFECT TRANSISTOR</td>
</tr>
<tr>
<td>VLSI</td>
<td>VERY LARGE SCALE INTEGRATED CIRCUIT</td>
</tr>
<tr>
<td>NM</td>
<td>NANOMETER</td>
</tr>
</tbody>
</table>

REFERENCES


Biographical Notes:

MR. S. ALAGUBALAKRISHNAN is presently pursuing M.E final year in Electronics and Communication Engineering Department (specialization in VLSI design) from TheniKammavarSangam College of Technology, Theni, Tamilnadu, India.