Efficient Method for Area-Efficient 32bit CSLA

Malti Aryayan\(^1\), Jaikaran Singh Chauhan\(^2\)
\(^1\)M.Tech Scholar, \(^2\)Asso.Professor, SSSUT & MS Sehore, India

Abstract— Modern applications demand extremely less area budgets and enhanced speed in computer architectures for battery-operated devices like Laptop and others. In this thesis, the main focus is on the area and provides high speed to the processors. Less area and high speed circuits are becoming more desirable due to growing portable device markets and they are also becoming more applicable today in processors. The main focus in this work is to improve the speed of the 32-bit processor and in this case the carry select adder is the better choice. The second concern in the design of this carry select adder is the reduction of area that is achieved by implementing the internal structure of the ripple carry adder in the branching architecture. The approach used here is to implement these pipelines in a manner that only one pipeline will be activated through which the carry is propagating. Furthermore, RTL level optimizations have also been done to ensure less area during intermediate computations. In this thesis, the entire adder architecture has been implemented using Verilog and simulated using ISE tool suite.

Keywords—32bit CSLA, Carry Select Adder, High Speed Adder

I. INTRODUCTION

The design of high speed, low power and as well as minimum area adder architecture has been the main concern of many Very Large Scale Integration (VLSI) researchers and this resulted in a large number of adder architectures. These various available architectures provide the capacity to obtain the gain more accurate and deep instinctive understanding of adder and thus suggest various implementations. Here we are going to discuss about the requirement of adder in processors and will also discuss about different adder architectures. This Design uses a simple and efficient gate-level modification to meaning fully shrink the area and power of Carry Select Adder (CSLA). Based on this modification 16-, 32 Square-Root Carry Select Adder (SQR T CSLA) architecture have been developed and compared with the regular SQR T CSLA architecture.

The proposed Work has reduced area, and delay as compared with the regular SQRT CSLA. The CSLA is used in computational systems to shrink the badly behaved delay by autonomously producing multiple carries and then select a carry to generate the sum of carry propagation. In this thesis, the main focus is on the low power consumption and provides high speed to the processors. In processors the main concern is an adder which is required in ALU to perform arithmetic operations as well as to decode and fetch addresses in the memory and from the memory. A pipelining technique is using here to reduce the area and provide high speed of data processing. The carry select adder belongs to the grouping of conditional sum adder as it works over conditional statements. Conditional sum adder means which works on some condition. In this sum and carry are calculated by assuming input carry as 1 and 0 coming before the input carry. When actual carry input arrives in system, the actual calculated values of sum and carry are selected using multiplexer.

II. METHODOLOGY

The actual modification which has been made in this proposed CSLA is to use a method which gives more optimum results is to apportion the adder non-linearly. Here, as the computation of 16-bits has been implementing, Thus, to minimize the number of stages require for the computation can reduced only and only if the variable size ripple carry adders are implement. Fig.1. Shows 16-bit carry select adder design.

This concept can be understood as; if linear carry propagation style is used then it will require the 4-blocks of 4-bit RCA. But if the non-linear style of carry propagation has been used then the implementation of 16-bits can be performed in only 4-stages and in this way delay will be reduced and also the area.
For example to design a 16 bit Carry-Select Adder one can use 6 stages of adders with sizes: 4, 4, 4, 4 = 16 bits. Each stage computes a partial sum; Ripple adders can be used for stage adders.

III. RESULT ANALYSIS
This is our proposed result section that is show design architecture which is design with the help Xilinx. In a synthesis results, there are shown the top level block diagram, RTL view and the technological view of the implemented CSLA. In the top level block diagram a green block shows the block diagram of 32-bit carry select adder which consist of two buses A<31:0> and B<31:0> and clock pulse and sum output is obtain of 32-bit.
Fig 5: Technology Design 32 Bit CSLA Internal Structure

Fig 6: RTL Design 4 Bit CSLA

Fig 7: RTL Design 4 Bit CSLA internal structure

Fig 8: RTL Design 5 Bit BCD internal structure

Fig 28: RTL Design 5 Bit BCD

TABLE 1. Comparative Analysis with Base Paper

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Performed work</th>
<th>Base Paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delay(ns)</td>
<td>3.494</td>
<td>5.482</td>
</tr>
<tr>
<td>Area(Total Gate count)</td>
<td>272</td>
<td>408</td>
</tr>
</tbody>
</table>
In this paper, the entire adder architecture has been implemented using VHDL and simulated using Mentor Graphics tool. This paper has deal with fundamental concepts of addition and optimization. Here we are presented a number of interesting results. Here the conclusion is obtained that this proposed structure of CSLA is better than all the architectures that have been implemented previous, since every constraints are optimized by using this proposed CSLA. In this paper show differ result.

REFERENCES

