Performance Comparision of Carry Select Adders

Nithin Bidare Puttaraju¹, Adithya²
¹,² Electronics and Communication, Global Academy of Technology, Karnataka, India-560098
Email address: ¹nithinb23@gmail.com

Abstract—Adder is a digital circuit that performs addition of
numbers. To perform fast arithmetic operations, carry select adder
(CSLA) is one of the fastest adders used in many data-processing
processors. The structure of CSLA is such that there is further scope
of reducing the area, delay and power consumption. Simple and
efficient gate – level modification is used in order to reduce the area,
delay and power of CSLA. Based on the modifications, 8-bit, 16-bit,
32-bit architectures of CSLA are designed and compared. In this
paper, conventional CSLA is compared with Modified Carry select
adder (MCSLA) and proposed CSLA, in terms of area, delay and
power consumption. The result analysis shows that the proposed
structure is better than the conventional CSLA.

Keywords— Adder, carry select adder (CSLA), modified CSLA
(MCSLA), proposed CSLA (CSLA), data processing processors.

I. INTRODUCTION
Adders are not only used in the arithmetic logic unit, but also
in other parts of the processor, where they are used to
calculate addresses, table indices and similar applications.
Some other applications of adders are in Multiply –
Accumulate (MAC) structures. Adders are also used in
multipliers, in high speed integrated circuits and in digital
signal processing to execute various algorithms like FFT, IIR
and FIR. Now a days, design of low power, area efficient high
speed data path logic systems are the most substantial areas in
the research of VLSI design. On the basis of requirements
such as area, delay and power consumption some of the
complex adders are Ripple Carry Adder, Carry look-Ahead
Adder and Carry Select Adder. Ripple Carry Adder (RCA)
shows the compact design but their computation time is
longer. Time critical applications make use of Carry Look-
Ahead Adder (CLA) to derive fast results but it leads to
increase in area. But the carry select adder provides a
compromise between the small areas but longer delay of RCA
and large area with small delay of Carry Look Ahead adder [1].

This paper presents a comparative analysis of various adders
and proposed design of CSLA by sharing Common Boolean
Logic and modified CSLA using Binary to Excess-1
Converter (BEC). Both these adders show less area, delay and
power than other adders.

This paper is organized as follows: In section I Types of
adders is shown, section A. deals with Ripple Carry Adder
section B. deals with modified CSLA, section C. explains
about Proposed CSLA using common Boolean logic. II
Results are analyzed in section III and Conclusion in section
VI.

II. ADDERS
A. Ripple Carry Adder

Ripple Carry Adder consists of cascaded — N\text{bit single bit full
adders. Output carry of previous adder becomes the input
carry of next full adder. Therefore, the carry of this adder
traverses longest path called worst case delay path through N
stages. Figure 1 shows the block diagram of ripple carry
adder. Now as the value of N increases, delay of adder will
also increase in a linear way. Therefore, RCA has the lowest
speed amongst all the adders because of large propagation
delay but it occupies the least area.

The basic idea of this work is to use Binary to Excess-1
converter (BEC) instead of RCA with Cin=1 in conventional
CSLA in order to reduce the area and power. [2], [3] BEC
uses less number of logic gates than N-bit full adder structure.
To replace N-bit RCA, an N+1 bit BEC is required. Therefore,
Modified CSLA has low power and less area than conventional
CSLA. CSLA has been chosen for comparison with modified design using BEC as it has more balanced
delay, less area and low power [4]. Regular CSLA also uses
dual RCAs. In order to reduce the area and power, the design
is modified by using BEC instead of RCA with Cin=1.
Therefore, the modified CSLA occupies less area and low
power. Further also, the parameters like delay, area and power
can be reduced.

B. Modified CSLA

Fig. 1. 4-bit ripple carry adder

Fig. 2. Conventional carry select adder

By sharing Common Boolean Logic (CBL), a circuit of CSLA is proposed. This proposed design is better than all the other adders in respect of area, delay and power consumption.

B. Modified CSLA

The main idea of this work is to use BEC instead of RC with carry Cin=1 in order to reduce the area and power of conventional CSLA. BEC [3] is a circuit used to add 1 to the input numbers. Circuit of BEC as shown in figure 3. Goal of addition is achieved using BEC together with the multiplexer as shown in figure 4. One of the input of 8:4 MUX gets as its inputs(B3,B2,B1andB0) and another input of MUX is BEC output. Boolean expressions of 4-bit BEC are listed below

\[
\begin{align*}
X0 &= \neg B0 \\
X1 &= B0 \oplus B1 \\
X2 &= B2 \oplus (B0 \& B1) \\
X3 &= B3 \oplus (B0 \& B1 \& B2)
\end{align*}
\]

The main idea of this work is to use BEC instead of RCA with Cin=1 in order to get the reduced area and power consumption of the conventional CSLA. To replace the N-bit RCA, N+1 bit BEC is required

C. Proposed CSLA Using Common Boolean Logic

To remove the duplicate adder cells in the conventional CSLA, an area efficient CSLA is proposed by sharing Common Boolean Logic (CBL) term as shown in figure 5. The truth table shown in table I of a single-bit full-adder indicates that output sum (S0) is Ex-OR of inputs A and B when carry initial is logic —01 while output S0 is Ex-NOR of inputs A and B when carry initial is logic —11 as illustrate as two red circles in truth table. The improved CSA can be implemented by using this technique of sharing the common Boolean logic term in summation generation.

<table>
<thead>
<tr>
<th>Cin</th>
<th>A</th>
<th>B</th>
<th>S0</th>
<th>C0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Fig. 3.** 4-bit binary to excess-1 converter

**Fig. 4.** 4-bit binary to excess-1 logic with 8:4 multiplexer

Thus, modified CSLA is designed such that it occupies less area and has low power than conventional CSLA.

**Fig. 5.** Internal structure of the proposed area-efficient carry select adder is constructed by sharing the common Boolean logic term

Hence we need to use Ex-OR gate and INV gate to generate the output sum signal pair. Sum output either the Ex-OR or the Ex-NOR could be selected using the multiplexer with select line as previous carry signal. The truth table also reveals that output carry (C0) is AND of A, B inputs when initial carry is logic —01 while C0 is OR of A, B when initial carry is logic —11. Same previous carry as select line to second multiplexer is used to select the carry output of the first stage which would act as select line of the multiplexers in the second stage. As both sum generation and carry generation is carried out in parallel therefore there exist some competitiveness in speed also the power consumption reduces as duplication of the hardware doesn’t exist in improved CSA as in case of the conventional This method replaces the Binary
to Excess-1 converter add one circuit by common Boolean logic. As compared with modified CSLA, the proposed structure is little bit faster. Internal structure of proposed CSLA is shown in figure 5.

### III. RESULT

<table>
<thead>
<tr>
<th>Model</th>
<th>Area</th>
<th>Power</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSLA</td>
<td>129.8</td>
<td>48289.875nw</td>
<td>43.311ns</td>
</tr>
<tr>
<td>Modified CSLA</td>
<td>909</td>
<td>79.81MILIW</td>
<td>42.65 NS</td>
</tr>
<tr>
<td>Proposed CSLA</td>
<td>835</td>
<td>18887.119nw</td>
<td>44.36 NS</td>
</tr>
</tbody>
</table>

### IV. CONCLUSION

The radix-8 multiplier is designed by using Carry Select Adder, Modified Carry Select Adder and Proposed CSLA Using Common Boolean Logic, and area occupied is compared with Carry Select Adder, Modified Carry Select Adder and Proposed CSLA Using Common Boolean Logic. Among these three designs adders using CSLA Using Common Boolean Logic is having less area occupied than compared to adders using Carry Select Adder, Modified Carry Select Adder. The propagation delay can be further reduced by using different combination of adders and also work can be done to reduce the area occupied and power dissipation.

### REFERENCES