Design and Analysis of 32-bit Parallel Prefix Adders for Low Power VLSI Applications

Samraj Daphni*, Kasinadar Sundari Vijula Grace

Electronics and Communication Engineering, Noorul Islam Centre for Higher Education, 629180, India

ARTICLE INFO
Article history:
Received: 26 January, 2019
Accepted: 27 February, 2019
Online: 15 March, 2019

Keywords:
Parallel Prefix Adder
Ladner Fischer Adder
Power Gating Technique
Carry Generation Stage
Carry Look ahead Adder

ABSTRACT
The basic processes like addition, subtraction can be done using various types of binary adders with dissimilar addition times (delay), area and power consumption in any digital processing applications. To minimize the Power Delay Product (PDP) of Digital Signal Processing (DSP) processors is necessary for high performance in Very Large Scale Integration (VLSI) applications. In this paper, a 32-bit various Parallel Prefix adders design is proposed and compared the performance results on the aspects of area, delay and power. Implementation (Simulation and Synthesis) results really achieve significant improvement in power and power-delay product when compared with the previous bit adders which is used in processors. To reduce the power, here apply the energy recovery logic like power gating technique for all three adders. All the simulations and synthesis results can be noted using Xilinx ISE 14.2i tool.

1. Introduction

Generally, the basic processes such as addition, subtraction, division that can be done by using different types of binary adders in any digital based processors and control systems [1]. The adder performance which is used in the device is only measured the high speed and accuracy of a processor or system. Previously the processors are used 32 bit carry adders like Ripple Carry Adder (RCA), Carry Propagate Adder (CPA), and Carry Look ahead Adders (CLA) with different addition times (delay), area and power consumption [2]. How fast the carry reaches for every single bit position, from which the delay of any binary adder is calculated. Henceforth, the carry chain which generates the carry bit is the major challenge in binary adder design. But the above existing 32 bit basic carry adders having high delay value in higher order bits because each level of adder has to wait for the previous carry result [3]. Due to the above problem of 32 bit basic existing carry adders, in today’s world of technology, PPA is well suitable designed adder for high speed addition process with less delay in VLSI technology [4]. Also the PPA is one of the most popular designs and provides good negotiation amongst area, speed and power [5]. The low order PPA is designed at earlier like 8 bit and 16 bit. This paper is designed as follows, in second section: describes briefly about PPA, in third section: explains the design of 32 bit proposed Parallel Prefix (Kogge Stone Adder, Brent kung Adder, Ladner Fischer Adder), in fourth section: it expresses the simulation results (waveform and reports) of 32 bit PPA which we have designed in the previous section with the performance aspects (delay, area and power). The last section is concluded that Kogge Stone Adder performance is best among other adders with low power and less delay from the above analyzed results of 32 bit PPA.

2. Parallel Prefix Adder

Now a days, to avoid the higher delay problem of existing carry adders the PPA is used which is simply the modified design form of CLA. The Prefix adders can be designed in many different ways based on the different requirements and the production of carries [6]. Recently, use the tree structure form of adders to raise the speed of addition function in any kind of processors. PPA are fastest adders with tree structure based and used for high performance arithmetic processes in successive industries and DSP laboratories [7].

The PPA’s are also called as logarithmic delay adders because the delay value is established using logarithmic functions [8]. Addition in PPA can be processed using three main actions such as Pre-computation (P and G signal generation), Prefix-computation (carry signals group generation), Post-computation (Sum signal generation) [9].

2.1. Pre – Computation

In the pre-processing stage, propagate functions and generate functions are calculated depends upon the given input signals [10]. The propagate functions are carried out by the equation (1).

*Samraj Daphni, W/O. Mr. D. Thavasumony, North Midalakkadu, Palappallam Post, Kanyakumari District – 629159, Tamil Nadu, India, +91 8675857541 & daphnihavasumony@gmail.com

www.astesj.com
https://dx.doi.org/10.25046/aj040213
functions are carried out by the –

It is stated that X and Y are the input signals that composed by XOR logic gate. The generate functions are carried out by the equation (2).

\[ G[j] = X[j] \cdot B[j] \]

(2)

It is stated that X and Y are the input signals that composed by AND logic gate. Since the above equations (1) and (2) are done in parallel, it does not increasing a significant calculation of area consumption and delay fully depends upon the bit size which is desired at the input [11].

2.2. Prefix – Computation

This prefix computation stage, calculates the carry signal groups directly, which uses the input and values which measured from the first stage. Carry signals generation uses the more than two inputs for which the delay is automatically increased in this process [12]. The carry propagation function and carry generation function [13] is measured by the equations (3) and (4).

\[ \text{Carry } P = P[j] \cdot P[j + 1] \]
\[ \text{Carry } G = (G[j] \cdot P[j + 1]) + G[j + 1] \]

(3)

(4)

2.3. Post – Computation

In this stage, the sum result is generated by an Ex-OR operation that uses the values of carry generation stage (prefix-computation). The last sum operation is calculated by the equation (5)

\[ S[j + 1] = P[j + 1] \oplus C[j] \]

Where C is the last carry signal and P is the propagate function [14].

3. Design and analysis of proposed 32 bit various PPA

To overcome high delay problem of existing carry adders this work proposed the design of 32-bit various PPA for less delay and low power VLSI applications. This proposed system consists of two modules: The first module is to design of 32 bit PPA like KSA, BKA, and LFA. The second module is to analyses the performance comparison of PPA on the basis of area, delay and power.

In this section, analyse the different technologies of adders to design in the form of parallel prefix, apart from the RCA topology, such as Kogge-Stone, Brent-Kung, and Ladner-Fisher PPA. The important aim is to examine the trade-off between area consumption delay and power consumption in the particular PPA depends upon the design performance. All the designs are using a power gating technique to reduce the power consumption [15].

3.1. Kogge Stone Adder (KSA)

Normally, the KSA attains the key role with fast addition operation and it reflects like prefix form of Carry Look ahead Adder (CLA). Also this type of PPA entirely decreases the delay time in design to generate the carry signals [16]. Henceforth this KSA is popularly used in DSP (Digital Signal Processing) laboratory and Control system industries for fast arithmetic function.

The structure of 32-bit KSA design is exposed in Figure 1. This design can be divided into 5 stages [17]. The calculation of Propagate and Generate signals using full adders with carry input that process included in first and second stage. The generation of carry signals which used the values of Propagate and Generate that process included in third and fourth stage. The calculation of sum bits based on the P and carry generation values that is included in the fifth stage [18]. This 32 bit design of KSA is coded by VHDL and viewed the test bench waveform and analyzed the performance and noted the results.

3.2. Brent Kung Adder (BKA)

The BKA calculates the prefixes based on the bit groups. Initially calculate the prefixes values for 2 bit groups. These 2 bit prefix values are used to find the prefix values for the 4 bit groups, that are used to calculate the prefix values for 8 bit groups and so on [19]. Then these prefixes values are used to measure the carry out of the particular bit stage. These carries will be used along with the Group Propagate of the next stage to calculate the Sum bit of that stage [20]. Brent Kung Tree will be using (2\log2N-1) stages for any bit design.

The structure of 32-bit BKA design is given in Figure 2. Hence the designing of 32- bit adder takes the number of stages will be 9. The fan-out for every bit stage is limited to 2. The above diagram shows the fan out being reduced and the loading on the advance stages being reduced [21]. This 32 bit design of BKA is coded by VHDL and viewed the test bench waveform and analysed the performance and noted the results.

3.3. Ladner Fischer Adder (LFA)

The LFA tree structures are a family of tree networks between Brent Kung and Sklansky tree. It is very close like to Sklansky PPA, but it calculates the prefix values for odd number bits after that again uses another stage which ripple into
the even locations [22]. At higher order bits, to get improved in speeds the cells must still be properly sized or grouped.

Figure 3: 32 bit Ladner Fischer Adder design

The structure of 32-bit LFA design is exposed in Figure 3. The Ladner Fischer adder is used for high performance arithmetic operation with complicate designs. The LFA consists of black cells and gray cells with 5 stages for 32 bit design. Each black cell encloses only one OR logic gate and two AND logic gates. Each gray cell contains only one AND logic gate [23]. This 32 bit design of LFA is coded by VHDL and viewed the test bench waveform and analysed the performance and noted the results.

4. Simulation results of proposed 32 bit PPA

In this simulation section, took all three types of 32 bit parallel prefix adders (KS, BK, LF) that are discussed above. All the PPA’s are designed on VHDL (Very high speed Hardware Description Language) / Verilog project navigator 14.2i is used for synthesis (Xilinx version) [24]. Simulation results are verified on the basis of area, power and delay. In addition to that the waveforms and the comparison results for all three parallel prefix adders are given.

Figure 4: Area Consumption report for KSA

Figure 5: Power Analyzer report for KSA

Figure 6: Test bench waveform results of KSA

Figure 7: Area Consumption report for BKA

Figure 8: Power Analyzer report for KSA

Figure 9: Test bench waveform results of BKA

Figure 10: Area Consumption report for LFA
techniques can be achieved on the performance parameters that will be the future work of the paper.

References


Table 1: Comparison results of area, delay and power for various types of PPA

<table>
<thead>
<tr>
<th>Adder Types</th>
<th>LUT’s Used</th>
<th>Delay (ns)</th>
<th>Power (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kogge Stone Adder</td>
<td>189</td>
<td>6.483</td>
<td>21</td>
</tr>
<tr>
<td>Brent kung Adder</td>
<td>156</td>
<td>6.489</td>
<td>24</td>
</tr>
<tr>
<td>Ladner Fischer Adder</td>
<td>51</td>
<td>21.879</td>
<td>24</td>
</tr>
</tbody>
</table>

Conclusion

In this paper, an efficient 32 bit Parallel Prefix adders like KSA, BKA, LFA is designed. This proposed 32 bit adder addition operation offers a great advantage in reducing delay. For low power VLSI applications, also the designed adders are compared on the basis of power, area consumption, and delay. The synthesis results reveal that among the proposed adders, KSA is achieved some saving of power-delay product due to less power utilization. But the area delay product is little increased, compared to other adders due to high area consumption. For decreasing the complexity at all performance aspects, further optimization


