

Design and Implementation of Multi Operand Binary Tree Adder Using Verilog HDL

¹N.C. CHANDU PRASANTH,
¹ASSISTANT PROFESSOR,
¹Department of ECE,
¹St. Peter's Engineering College, Hyderabad, Telangana
chanduprasanthkvl@gmail.com

Abstract— The critical path of ripple carry adder based binary tree adder is analysed to find the possibilities for delay minimization. Based on the findings of the analysis the new logic formulation and the corresponding design of RCA are proposed for the BTA. The comparison results show that the proposed RCA design offers better efficiency in terms of area, delay and energy than the existing RCA. Using this RCA design, the BTA structure is proposed. The synthesis results reveal that proposed 32-operand BTA provides the saving of 22.5% in area-delay product and in energy delay product over the recent Wallace tree adder which is the best among available multi-operand adders. The authors have also applied the proposed BTA in the recent multiplier designs to evaluate its performance. The synthesis result shows that the performance of multiplier design improved significantly due to use of proposed BTA. Therefore, the proposed BTA design can be a better choice to develop the area, delay and energy efficient digital systems for signal and image processing applications.

Key Words — Binary Tree Adder, Ripple Carry Adder, Verilog.

I. INTRODUCTION

Adders are the basic and the essential computational blocks in most digital logic and adiabatic logic-based circuits embedded in most of the real-time application systems in the electronics and communication field. Therefore, an area-efficient adder is designed here to design a low-power-consuming course. Nowadays, most of these systems are incorporating the VLSI integrated circuits. This is based on the consumer's choice to have miniaturized, low-power-consuming application circuits. We know that area reduction by reducing the number of transistors in a particular logic style while designing the full adder cells leads to a considerable decrease in overall power consumption. Concentrating on leakage power reduction is necessary to achieve a substantial reduction in static power consumption. We need to consider efficient power gating techniques to attain longer battery life on miniaturized, portable, and handheld systems. The decline in technology sizes during I.C. fabrication also leads to improved leakage current. Considering these concepts, in this research work, an efficient power gated adiabatic logic-based full adder cell is designed, which has less static and dynamic power consumption. This energy-efficient adder is incorporated inside the carry select adder. Thus, the designed carry select adder has very less static power.

II. LITERATURE SURVEY

Efstathiou, C., Owda, Z., Tsiatouhas, Y.: 'New high-speed multioutput carry look ahead adders', *IEEE Trans. Circuits Syst. II, Express Briefs*, 2013, 60, (10), pp. 667– 671

With the advancements in the semiconductor industry, designing a high-performance processor is a prime concern. Multiplier is one of the most crucial parts in almost every digital signal processing application. This paper addresses the implementation of an 8-bit multiplier design employing CMOS full adder, full adder using Double Pass Transistor (DPL) and multi output carry Look ahead logic (CLA). DPL adder avoids the noise margin problem and speed degradation at low value of supply

voltages associated with complementary pass transistor (CPL) logic circuits. Multi output carry look ahead adder leads to significant improvement in the speed of the overall circuitry. The investigation is carried out with simulation runs on HSPICE environment using 90 nm process technology at 25 °C. Finally, the design guidelines are derived to select the most suitable topology for the desired applications. Investigation reveals that multiplier design using multi output carry look ahead adder proves to be more speed efficient in comparison with the other two considered design strategies.

A static low-power, high performance 32-bit carry skip adder'. Euromicro Symp. on Digital System Design, 2004, DSD 2004, Rennes, France, 2004, pp. 615–619

The design of a 32-bit carry-skip adder to achieve minimum delay is presented in this paper. A fast carry look-ahead logic using group generate and group propagate functions is used to speed up the performance of multiple stages of ripple carry adders. The group generates and group propagate functions are generated in parallel with the carry generation for each block. The optimum block sizes are decided by considering the critical path into account.

Ramkumar, B., Kittur, H.M.: 'Low-power and area-efficient carry select adder', IEEE Trans. Very Large Scale Integr. Syst., 2012, 20, (2), pp. 371–375

The CSLA is used in many computational systems to alleviate the problem of carry propagation delay by independently generating multiple carries and then select a carry to generate the sum. However, the CSLA is not area efficient because it uses multiple pairs of Ripple Carry Adders (RCA) to generate partial sum and carry by considering carry input, then the final sum and carry are selected by the multiplexers (mux). The basic idea of this work is to use Binary to Excess-1 Converter (BEC) instead of RCA in the regular CSLA to achieve high speed and low power consumption.

Amin, A.: 'Area-efficient high-speed carry chain', Electron. Lett., 2007, 43, (23), pp. 1258–1260

An improved carry chain circuit with carry-skip capability is described. The carry-skip logic allows an arbitrarily long carry chain without the need for intermediate buffers for signal restoration, leading to an implementation that is both fast and area-efficient. The chain can flexibly accommodate technology-imposed maximum depth of NMOS transistor pull-down stack.

III. PROPOSED METHODOLOGY & IMPLEMENTATION

This is an interesting observation which can be used to minimize the critical path of the RCA design by performing the carry computation using complementary complex gates such as AND–OR–Invert (AOI) or OR–AND–Invert (OAI) gate in place of AO gate. the carry is expressed in the AOI form as

$$\overline{c_i} = \underbrace{\overline{g_i + p_i \cdot c_{i-1}}}_{\text{AOI}}$$

The i th carry signal computed is in complement form and cannot be directly used to compute the $(i + 1)$ th carry signal. Therefore, the expression of valid $(i + 1)$ th carry signal using the i th carry c_i is given as

$$c_{i+1} = \underbrace{\overline{\overline{g_{i+1} \cdot (p_{i+1} + \overline{c_i})}}}_{\text{OAI}}$$

The operation can be performed by using an OAI complementary gate which produces carry $c_{i + 1}$ signal in the normal form. However, it is clear that the OAI-based carry computation requires $p_{i + 1}$, $g_{i + 1}$ and c_i signals in their complemented form.

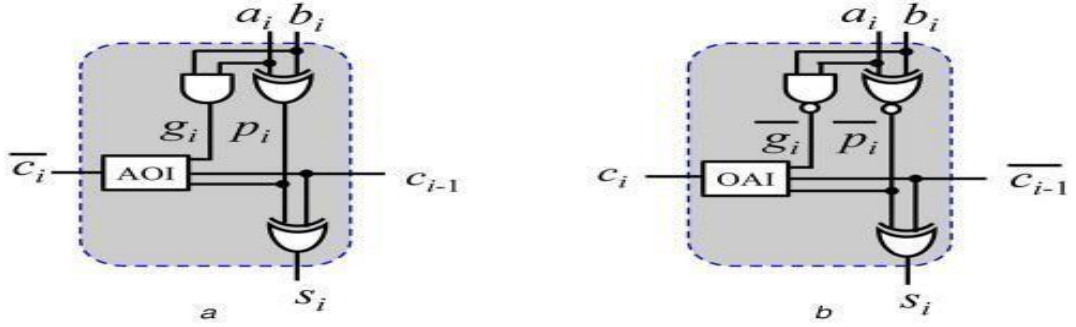


Fig 1: Complementary Gates

- Boolean expressions with AOI-based carry generation:

$$p_i = a_i \oplus b_i; \quad g_i = a_i \cdot b_i$$

$$\overline{c_i} = \overline{g_i + p_i \cdot c_{i-1}}$$

$$s_i = p_i \oplus c_{i-1}$$

- Boolean expressions with OAI-based carry generation:

$$\overline{p_i} = \overline{a_i \oplus b_i}; \quad \overline{g_i} = \overline{a_i \cdot b_i}$$

$$c_i = \overline{\overline{g_i} \cdot (\overline{p_i} + \overline{c_{i-1}})}$$

$$s_i = \overline{p_i} \oplus \overline{c_{i-1}}$$

The two 1-bit logic cells (AOI-LC and OAI-LC) are derived using the proposed logic formulation given the AOI-LC takes inputs a_i, b_i and c_{i-1} , and computes sum (s_i) and intermediate carry-out c_i signals while the OAI-LC computes sum (s_i) and carry-out (c_i) signals using 1-bit input signals a_i, b_i and c_{i-1} . Using AOI-LC and OAI-LC modules, an m -bit RCA design is proposed as (for even value of m). In the proposed RCA design, the AOI-LC and OAI-LC modules are connected alternatively because AOI-LC generates carry-out in the complement form while the OAI-LC generates normal carry-out.

It takes m -bit inputs (a and b) and initial carry-in (c_{in}) to compute the sum (s) and carry-out (c_{out}) signals, where the output carry is equal to c_{m-1} . If the value of m is odd then the proposed RCA includes AOI-LC in the MSB bit position which generates the carry-out signal in the complement form. Therefore, an actual output carry is obtained by complementing the carry-out signal of the AOI-LC placed in the MSB position.

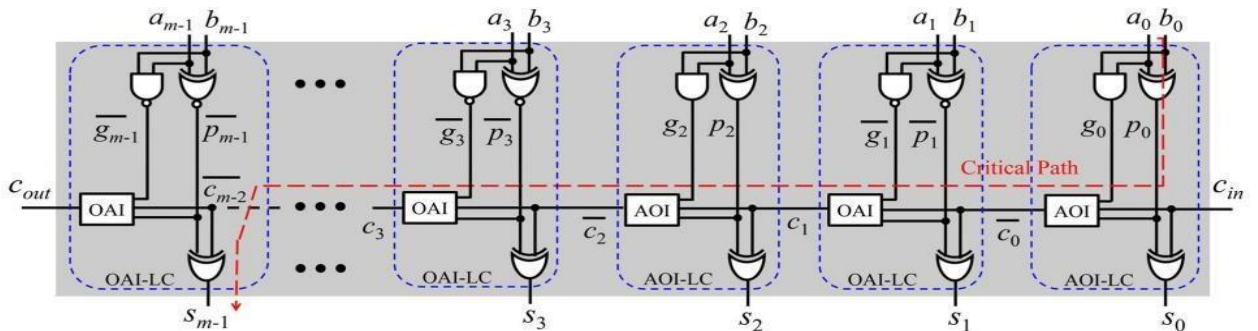


Fig 2: M-bit RCA

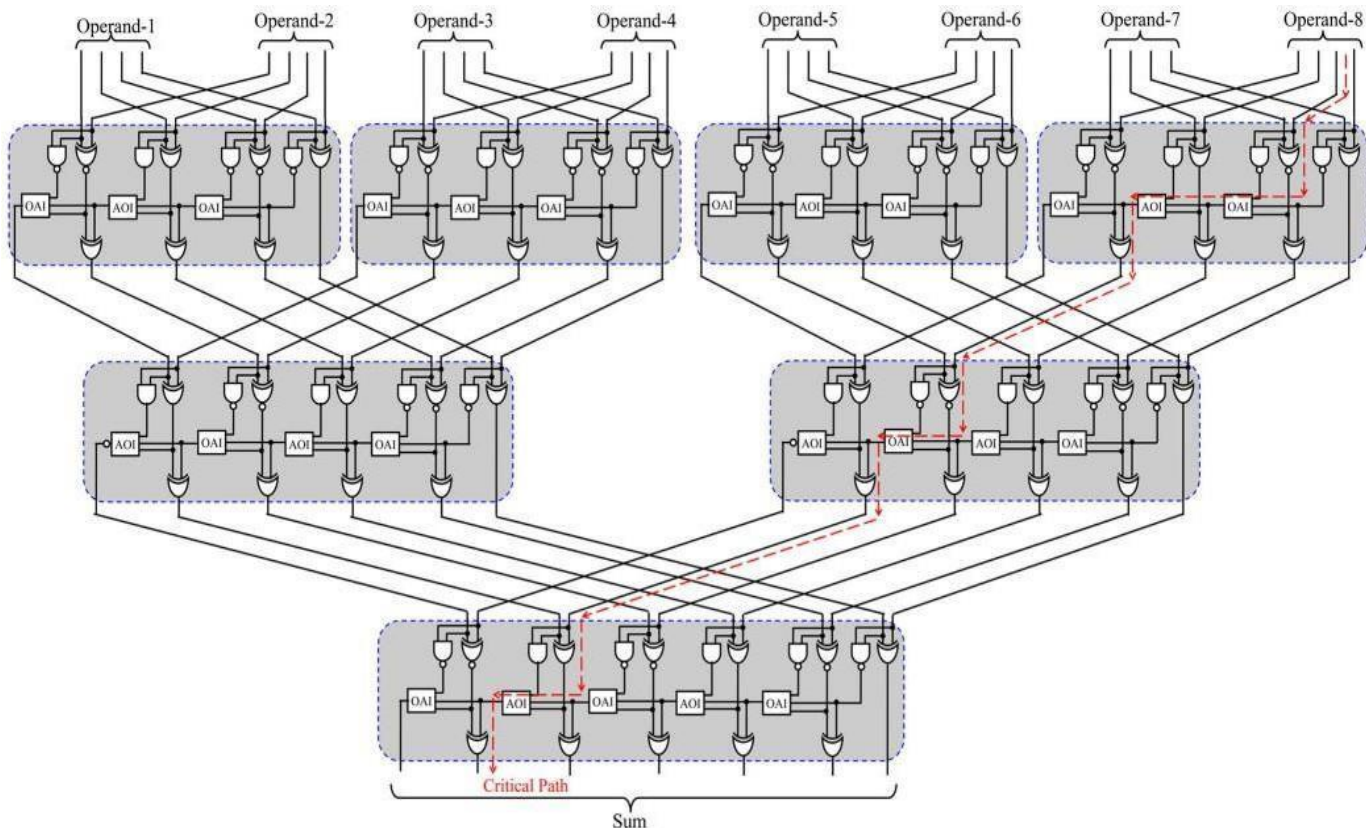


Fig 3: Proposed Architecture of Binary Tree Adder

Proposed binary tree adder The proposed BTA structure for $N = 8$ and $m = 4$ is It consists of seven RCAs, where the first stage uses four 4-bit RCAs, the second stage uses two 5-bit RCAs and the third stage uses one 6-bit RCA. These RCAs are made from AOI-LC and OAI-LC modules. The proposed BTA structure is scalable for any values of N and m .

It can be seen from Fig. 1 that the first, second and third addition stages of the proposed BTA, respectively, introduced four gates (one XNOR, one OAI, one AOI and one XOR), three gates (one XNOR, one OAI and one XOR) and three gates (one AOI and two XOR) in the critical path which is shown with red dotted line.

For simplicity in delay estimation, the equal delay is considered for AOI and OAI gates, similarly equal delay for XOR and XNOR gates. Based on this consideration, the generalized delay expression is derived for the proposed BTA is given as

$$T_{\text{Prop-BTA}} = 2T_{\text{XOR}} + (m - 2)T_{\text{AOI}} + (\log_2 N - 1)(T_{\text{AOI}} + 2T_{\text{XOR}})$$

Where $T_{\text{Prop-BTA}}$ Represents the delay of the proposed BTA takes AOI/OAI gates in the critical path. Therefore, it is expected that the delay of the proposed BTA will be significantly less over the existing RCA based BTA

IV. IMPLEMENTATION RESULTS

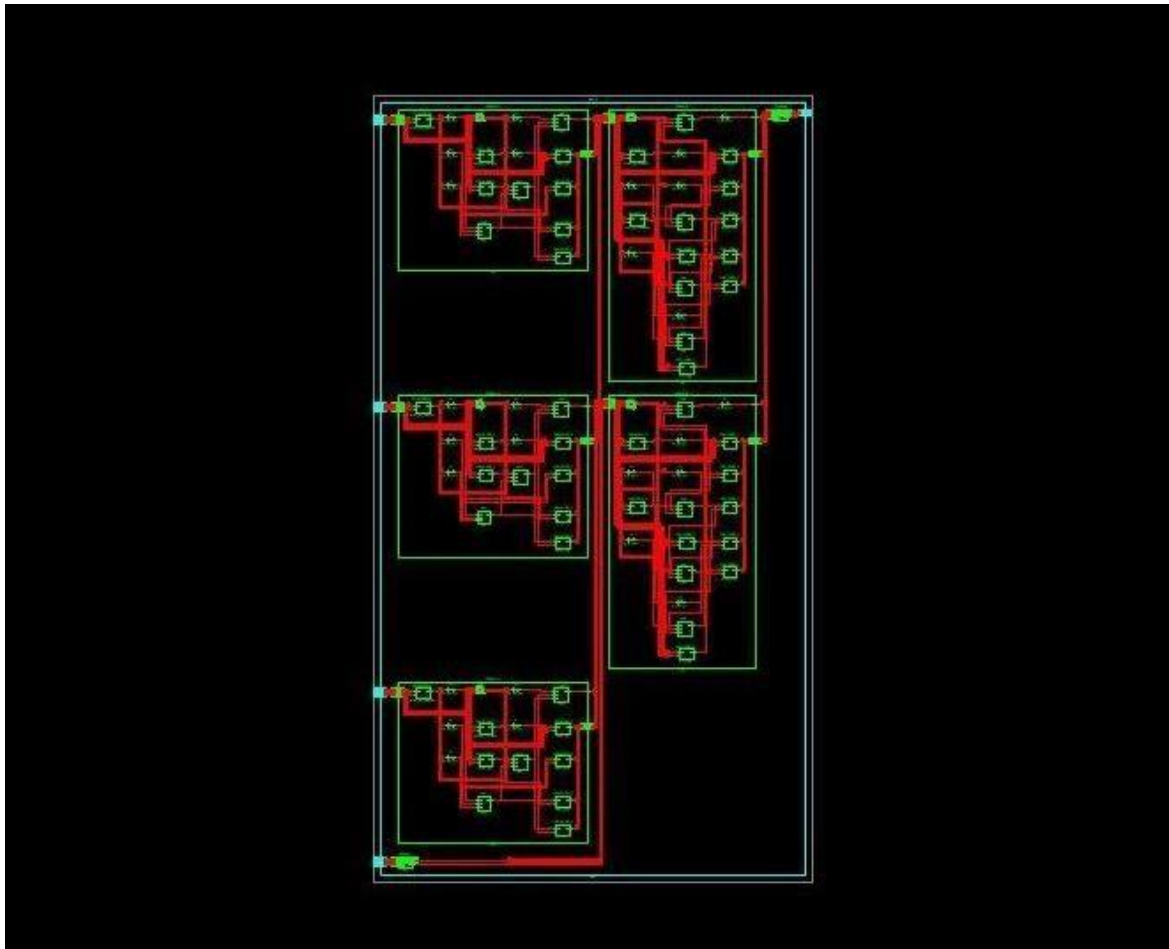


Fig 4: Proposed System RTL Schematic

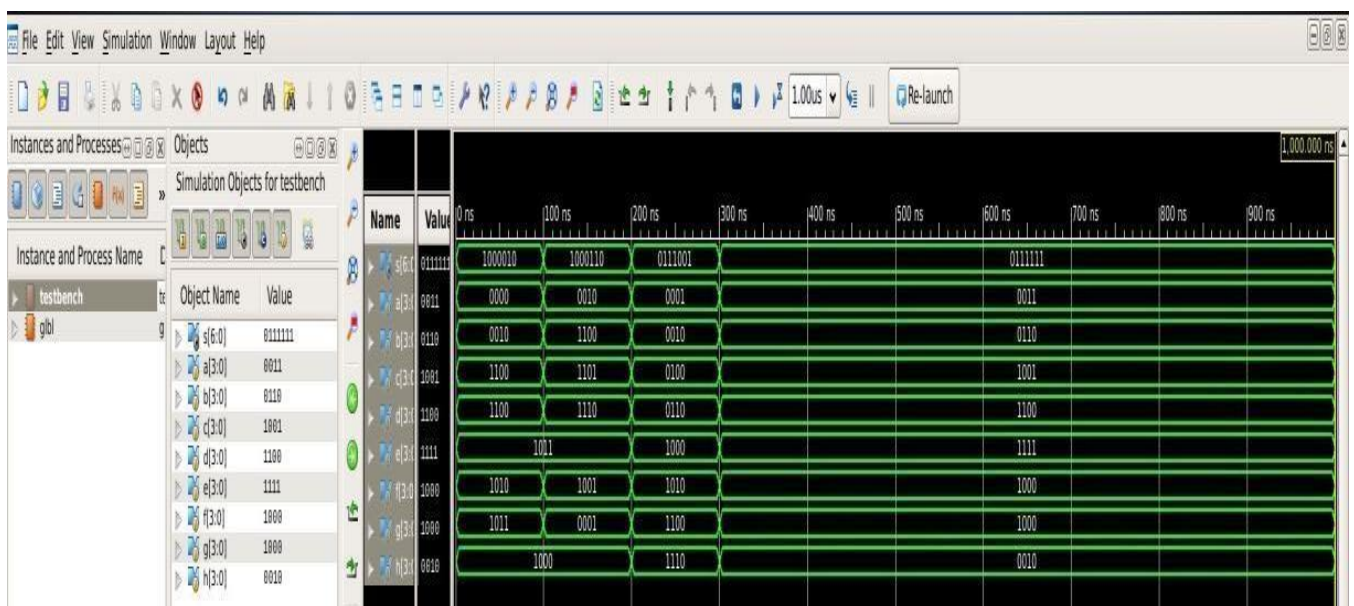


Fig 5: Simulation Result of Proposed Binary Tree Adder

| Adder | Bit width | Area | Delay | Less Area % | Less delay % |
|------------------|-----------|------|-------|-------------|--------------|
| RCA | 8 | 304 | 0.318 | - | - |
| | 16 | 608 | 0.586 | - | - |
| | 32 | 1216 | 1.121 | - | - |
| Prop. RCA | 8 | 280 | 0.211 | 7.89 | 33.6 |
| | 16 | 560 | 0.356 | 7.89 | 39.2 |
| | 32 | 1120 | 0.646 | 7.89 | 42.3 |

Table 1: Theoretical comparison for the proposed and existing RCAs

| Adder | Bit width | Area | Delay | Less Area % | Less delay % |
|--------------|-----------|------|-------|-------------|--------------|
| RCA | 4 | 6 | 4.614 | 34.5 | 65.2 |
| | 5 | 7 | 4.696 | 29.6 | 64.1 |
| | 6 | 9 | 5.273 | 23.04 | 58.7 |
| Prop. System | -- | 43 | 8.226 | 48.2 | 40.8 |
| Exe. System | -- | 33 | 7.395 | 62.8 | 45.4 |

Table 2: Comparison of Synthesis Results

V. CONCLUSION

The RCA-based BTA is widely used MOA due to its simplest structure that leads to area and energy efficient design. However, the long carry propagation path of RCA makes it poor in terms of delay performance. Therefore, in this paper, the delay analysis of RCA based BTA is presented. The comparison results show that the proposed RCA provides better efficiency in terms of area, delay and energy than the existing RCA. Using this RCA design, the BTA structure is proposed. We have also applied the proposed BTA in the recent multiplier designs to evaluate its performance. The synthesis result shows that the performance of multiplier designs improved significantly due to use of proposed BTA. Therefore, the proposed BTA design can be better choice to develop the area, delay and energy efficient digital systems for signal and image processing applications

VI. FUTURE SCOPE

Any VLSI application includes certain actions such as area, power, delay and performance. Proposed design has an immense future scope by the inclusion of complementary gates to the existing system for better results and further detailed analysis. With the help of complementary gates, the vast structures that will be work largely from industrial side can be useful.

REFERENCES

- [1] Amin, A.: 'Area-efficient high-speed carry chain', *Electron. Lett.*, 2007, 43, (23), pp. 1258–1260
- [2] Efstathiou, C., Owda, Z., Tsiatouhas, Y.: 'New high-speed multioutput carry look-ahead adders', *IEEE Trans. Circuits Syst. II, Express Briefs*, 2013, 60, (10), pp. 667–671
- [3] Kogge, P.M., Stone, H.S.: 'A parallel algorithm for the efficient solution of a general class of recurrence equations', *IEEE Trans. Comput.*, 1973, 100, (8), pp. 786–793
- [4] Efstathiou, C., Vergos, H.T., Nikolos, D.: 'Fast parallel-prefix modulo $2n + 1$ adders', *IEEE Trans. Comput.*, 2004, 53, (9), pp. 1211–1216
- [5] Hoe, D.H., Martinez, C., Vundavalli, S.J.: 'Design and characterization of parallel prefix adders using FPGAs'. 2011 IEEE 43rd Southeastern Symp. on System Theory, Auburn, AL, USA, 2011, pp. 168–172
- [6] Vergos, H.T., Dimitrakopoulos, G.: 'On modulo $2n + 1$ adder design', *IEEE Trans. Comput.*, 2012, 61, (2), pp. 173–186
- [7] Didier, L.S., Jaulmes, L.: 'Fast modulo $2n - 1$ and $2n + 1$ adder using carrychain on FPGA'. 2013 Asilomar Conf. on Signals, Systems and Computers, Pacific Grove, CA, USA, 2013, pp. 1155–1159
- [8] Daphni, S., Grace, K.V.: 'A review analysis of parallel prefix adders for better performance in VLSI applications. 2017 IEEE Int. Conf. on Circuits and Systems (ICCS), Thiruvananthapuram, India, 2017, pp. 103–106