Reduced Complexity Hybrid Ripple Carry Lookahead Adder

Ravish Aradhya H.V
Department of E and C,
R V college of Engineering,
Bangalore 560059, India.

Lakshmesha J
Department of E and C,
R V college of Engineering
Bangalore 560059, India.

Muralidhara K.N
Department of E and C,
P E S college of Engineering
Mandya 571 401, India.

ABSTRACT

In this paper we discuss Hybrid Ripple Carry Lookahead Adder (HRCLA), which is a hybrid between Carry Lookahead Adder (CLA) and ripple adder (RA). In HRCLA time is traded off for area and power. HRCLA has been designed by rippling the last carry bit of a 4-bit CLA. HRCLA extracts the traits of Carry Lookahead Adders (CLA) speed and ripple adders (RA), area. A four bit proposed HRCLA has been implemented in Cadence using 45nm technology; the implementation results showed 12.2 %Area, 4.6 % power improvement and 14.01 % critical path delay overhead over CLA.

General Terms

Hybrid adders, Tradeoff adders.

Keywords

Ripple Adder, Carry Lookahead Adder (CLA), Hybrid Ripple Carry Lookahead Adder (HRCLA).

1. INTRODUCTION

The need for low area and power design is the impact of many design issues, such as the exponential increase of portable device market, heat sinking cost, device reliability and environmental effects. Shear processing power is not the only factor for designing a chip, but also with a low power and area efficient design, can make it the market winner and acquire a major part of the market.

In any processing element, an adder is an indispensible component. Adders speed places a limitation on the operating frequency and it consumes high power due to high transition density [8]. This leads to creation of hot spots and hence device reliability issues, reducing the shell life. To reduce this problem tradeoff has to be performed. Optimization of speed, area or power of adders is crucial for the design. Many hybrid designs of adders have been proposed, which discusses how to tradeoff area, power or speed by combining two or more types of adder [1], [5]. When tradeoff is performed on one parameter, might degrade the other parameters [6].

The ripple carry adder (RCA) consumes very less area, but as the number of bits increases the critical path delay (carry in to carry out) increases linearly [6], [7]. Hence for high speed designs RCA is not preferred. On the other hand the CLA is designed by separating the carry chain and the sum logic[2]-[4], there by the critical path delay is substantially reduced but the complexity, area and power increases dramatically with the number of bits[2], [7].

The proposed HRCLA is a trade off between the CLA and RCA which reduces the complexity at the cost of critical path delay increase, in the HRCLA the critical path contains the full adder, the critical path delay cam be greatly reduced by using faster full adder [9].

2. DESIGN

2.1 Ripple Carry Adder Architecture

The ripple carry adder is constructed by cascading full adders (FA) blocks in series. One full adder is responsible for the addition of two binary digits at any stage of the ripple carry. The carryout of one stage is feed directly to the carry-in of the next stage. A number of full adders may be added to the ripple carry adder or ripple carry adders of different sizes may be cascaded in order to accommodate binary vector strings of larger sizes. For an n-bit parallel adder, it requires n computational elements (FA). Figure 1 shows the structure of an N-bit ripple carry adder. It is composed of four full adders. The input bits of 'A 'are added to the bits of 'B ',respectfully of their binary position. Each bit addition creates a sum and a carry out. The carry out is then transmitted to the carry in of the next higher-order bit. The final result creates a sum of N bits plus a carry out.

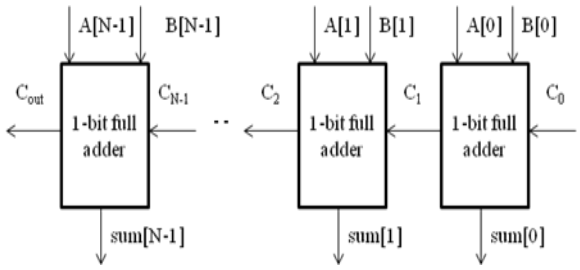

Fig 1: N-bit Ripple Carry Adder

As in Figure 1, RCA is cascaded structure of full adders. If A_{FA} is the area of a full adder then the total approximate area will be NA_{FA} and the critical path delay is given by

$$Tcri = Tc(N-1) + Ts + Tw \quad (1)$$

Where Tc is the delay to compute the carry out, Ts is the delay to compute the sum and Tw indicates the total wiring delay. Equation (1) is generic; more accurate results for Tcri can be obtained after layout generation.

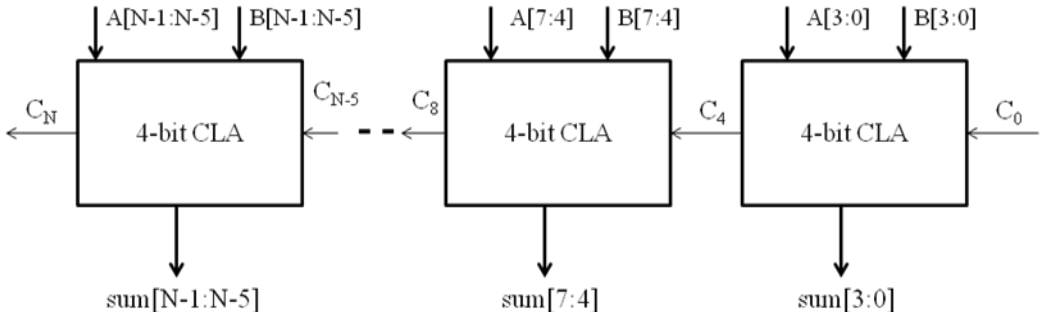

Fig 2: N-bit cascaded CLA structure

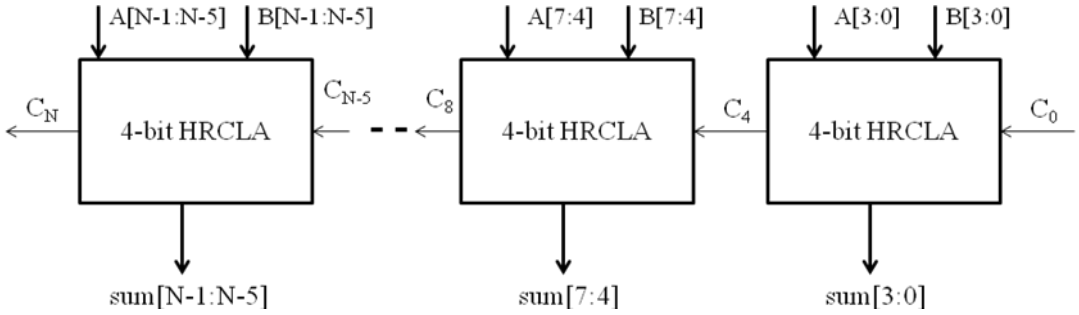

Fig 3: N-bit HRCLA structure

2.2 Carry Lookahead Adder Architecture

The delay in RA increases with number of stages, as seen in equation (1). CLA resolves this problem by separating sum and carries logic (see figure 2). The carry lookahead logic produces carry out even before the sum is computed. Hence CLA is faster than RA.

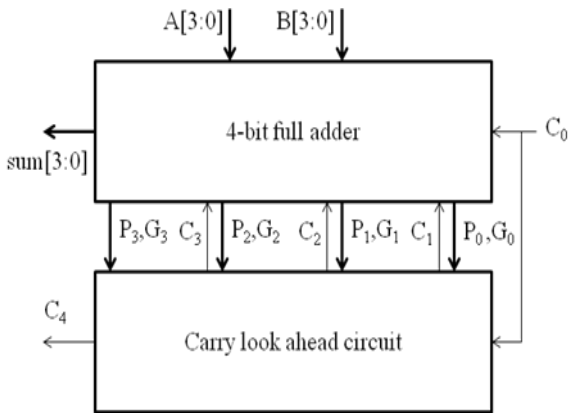

Fig 4: 4-bit Carry Lookahead Adder

To get more insight of operation, let us consider the following Boolean equations of the carry lookahead logic.

Pi = Ai XOR Bi Carry propagate

Gi = Ai AND Bi Carry generate

Both carry propagate and generate signals depend only on the input bits and thus will be Valid after respective gate delays (XOR gate area and delay is far more than AND gate).

For a four bit CLA, the governing equations for the carry are:

C1 = G0 + P0C0

C2 = G1 + P1G0 + P1P0C0

C3 = G2 + P2G1 + P2P1G0 + P2P1P0C0

C4 = G3 + P3G2 + P3P2G1 + P3P2P1G0 + P3P2P1P0C0

The above expressions show that C2, C3 and C4 do not depend on its previous carry-in. Therefore C4 need not to wait for C3 to propagate. As soon as C0 is computed, C4 can reach steady state. Another observation is that, as the number of bits increases the fan out, fan in and complexity increases rapidly, results in more area and delay. To generate C4 4 input OR gate and 5 input AND gate is needed. The carry propagate signal has to drive 3 AND gates, which is quiet high. Hence the CLA is limited to 4-bit block.

In RA the critical path is through adders but in CLA the critical path is in the carry lookahead circuit (see figure 2); as the number of bits increases RA's critical path delay and area increases linearly but for CLA it increases at a faster rate, so CLA is grouped into smaller parts usually of 4-bits each which reduces the fan-out need to generate carry out. An N bit CLA can be realized by cascading 4-bit CLA blocks (see figure 2).

2.3 The proposed Hybrid Ripple Carry Lookahead Adder Architecture

A 4-bit HRCLA is designed by rippling the 3rd carry of the CLA through a full adder to produce the fourth carry and sum (see Figure 4), by cascading this four bit block N-bit adder is realized as shown in Figure 3. In general by rippling the last but one carry through a full adder the HRCLA can be designed. The direct consequence intuitively is reduction of area and power, but an increase in delay. Another important

observation is the critical path is through the last full adder, when compared with CLA there is an increase in critical path delay because the last full adder has to wait for its prior carry out to be computed.

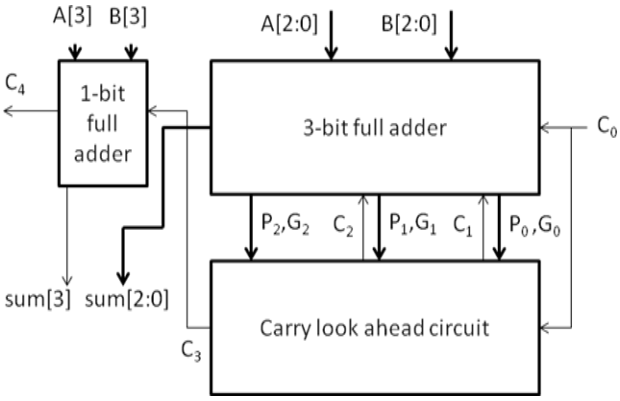

Fig 5: 4-bit HRCLA

For the 4-bit HRCLA shown in figure 4, critical path delay is through the 3-bit carry lookahead logic and the final full adder. This inclusion of the full adder has resulted in the timing overhead with respect to 4-bit CLA but achieves area and power saving. An N bit CLA can be realized by cascading 4-bit CLA blocks (see figure 3).

3. IMPLEMENTATION AND RESULT

The front end designs have been developed using Verilog HDL and synthesized in Cadence Encounter(R) RTL Compiler v08.10-s121_1 with the following attributes:

- **Technology library:** gscl45nm
- **Operating conditions:** typical (balanced_tree)
- **Wireload mode:** enclosed

The synthesized blocks of conventional CLA and the proposed HRCLA adder are shown in Figure 6 and 7 respectively. The lower half is the sum output logic and the upper is carry out logic but both logic share the XOR gates output because it is used for carry generate signal and where as carry propagate uses AND gates

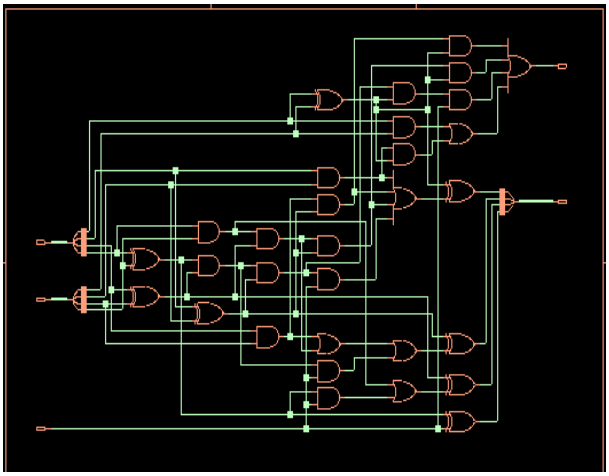

Fig 6: synthesized block of CLA

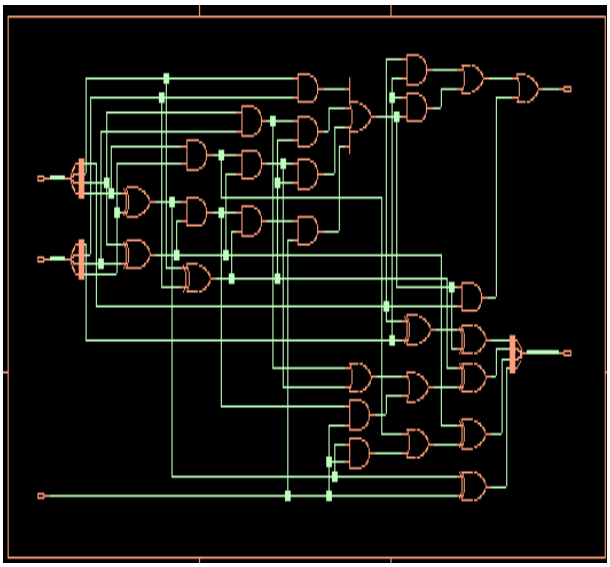

Fig 7: synthesized block of HRCLA

The number of gates and the routing complexity for the HRCLA substantially reduced, this is evident from figure 7 where carry out logic (upper half) is quiet reduced than CLA.

Table 1: Performance contrasts for four bit adders

	RCA	CLA	HRCLA
Average Power(µW)	14520.68	18002.76	17162.85
Area (µm^2)	76.24	125.79	110.54
Worst case delay(pS)	734.1	387.6	441.9

From Table 1, it is clear that the critical path delay has substantially increased over CLA. In HRCLA area saving is more than the power, as the number of bits increases saving is also more as in seen in the Figures 8 and 9, but the critical path delay of HRCLA also increases, as observed from the Figure 10.

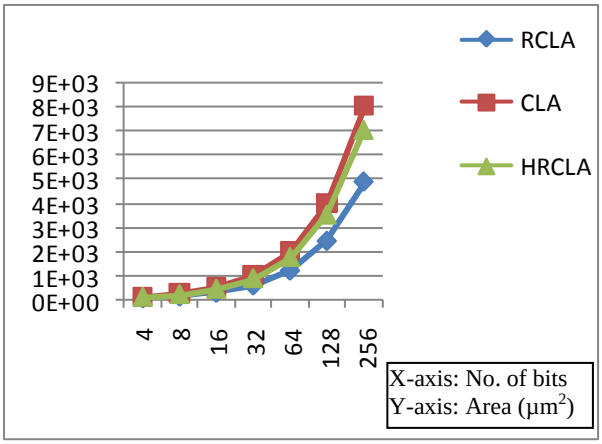

Fig 8: Area contrast of adders

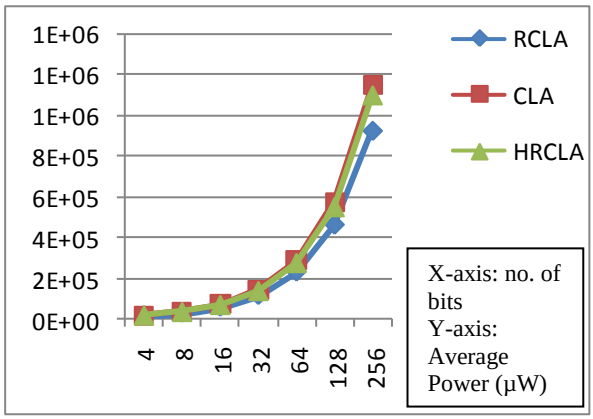

Fig 9: Power contrast of adders

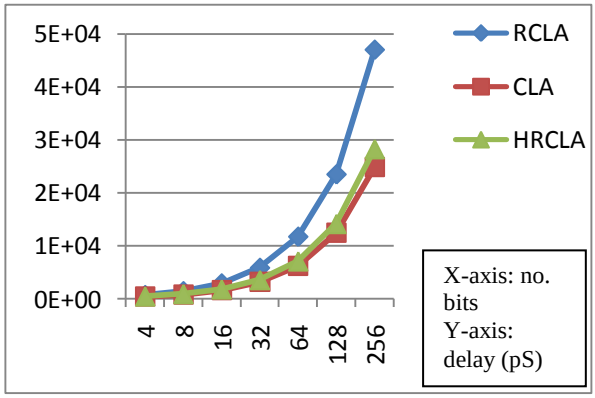

Fig 10: Critical path delay contrast of adders

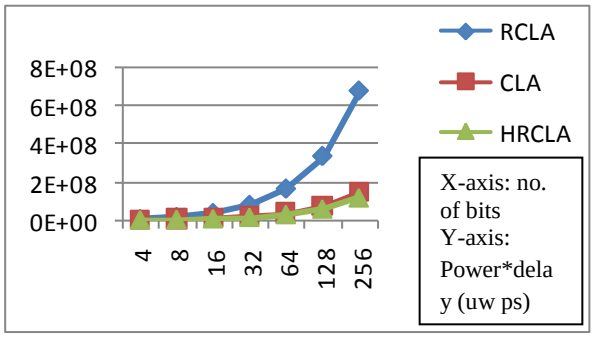

Fig 11: Variation of power delay product of adders

Summarizing, the HRCLA has achieved substantial power and area reduction but at the cost of increase of delay. The area, power, critical path delay curves Figure 8, 9, 10 shows that HRCLA curves are in the middle of the CLA and RCA curves this is the direct consequence of hybridization, hence its parameters fall in between them. But in the power delay product curve (see Figure 11), the HRCLA shows the least value and hence has better power product delay characteristics.

4. CONCLUSION

The intent the work was to trade off time for area and power optimization and has been achieved with marginal increase of critical path delay Figure 10. The power delay product curve (see Figure 11) shows that the HRCLA is the better choice. The comparative study of proposed and existing adder designs have been verified for its functionality and synthesized in cadence digital design flow (using gscl45nm technology library). The full adder falls in the critical path, which has increased the delay. The delay can be reduced by choosing minimum delay full adder [9], which can used to reduce the delay, perhaps even match the delay of the CLA.

5. REFERENCES

[1] Wang Y, Pai C, Song X, "The design of hybrid carry look-ahead/carry-select adders, Circuits and Systems II," Analog and Digital Signal Processing, IEEE Transactions, Volume 49, 2002, pp: 16-24.

[2] A. Weinberger and J. L. Smith, "A Logic for High-Speed Addition," National Bureau of Standards, Circ. 591, 1958, pp: 3-12.

[3] Chen P, Zhao J, Xie G, Li Y, "An improved 32-bit carry-lookahead adder with Conditional Carry-Selection," 4th International Conference on Computer Science & Education, 2009, pp: 1911-1913.

[4] G. A. Ruiz, M. Granda, "An area-efficient static CMOS carry-select adder based on a compact carry look-ahead unit," Microelectronics Journal 35. 2004, pp: 939–944.

[5] V. Kantabruta, "A recursive carry-lookahead/carry-select hybrid adder," IEEE Transactions on Computers, 1993, pp: 1495–1499.

[6] C. Nagendra, M. J. Irwin, and R. M. Owens, "Area-time power tradeoffs in parallel adders," IEEE Transactions on Circuits and Systems II, 2002, vol. 43, pp: 689-702.

[7] N. Weste and K. Eshragian, "Principles of CMOS VLSI Designs: A System Perspective," 2nd ed., Addison-Wesley, 2003.

[8] Kaushik Roy, Sharat Prasad, "Low-power CMOS VLSI Circuit Design," John Wiley, 2000.

[9] Phuong Thi Yen, Noor Faizah Zainul Abidin, Azrul Bin Ghazali, "Performance Analysis of Full Adder (Fa) Cells," IEEE Symposium on Computers & Informatics, 2011.