

Novel 16-Bit and 32-Bit Group High Speed BCD Adders

Sundaresan C
Manipal University

Chaitanya CVS
Manipal University

Mohan Kumar J
Manipal University

P R Venkateswaran
Bharat Heavy Electricals Limited

Somashekara Bhat
Manipal University

ABSTRACT

The VLSI binary adder is the basic building block in any computation unit. It is widely used in the arithmetic logic unit, memory addressing computation and in many other places. In this paper the binary adder is presented with keeping in mind speed, power and finally area. In this paper the BCD adder is designed using the mixed approach such as hierarchical, muxing and variable grouping techniques. The design and result are presented in this paper.

Keywords

BCD Adder, Hierarchy, Variable grouping, adder.

1. INTRODUCTION

Arithmetic unit is the most important component of modern embedded computer systems. In computing, Binary and decimal arithmetic operations are performed by arithmetic unit.

The arithmetic unit is a basic building block which play vital role in performing operations of a computer. Very powerful and complex arithmetic units[1, 2] are used in the processors of current generations Central Processing Units (CPUs) and Graphics Processing Units (GPUs).

Arithmetic unit generally includes floating point and fixed-point arithmetic operations and trigonometric functions. The arithmetic unit which is used to perform complex operations will have long latencies and high power consumption.

In electronics system, each digit in the decimal number can be represented in binary format using Binary Coded Decimal (BCD) encoding method. Decimal fractions cannot be represented by binary fractions, as they are pervasive in human activities.

Extensive work has been done on building adders for BCD arithmetic and different adders have been proposed [9, 10, 11, 12]. Enhancing the speed of operation is still the major consideration while implementing BCD arithmetic which is being addressed in this paper.

In arithmetic operations such as multi-operand addition [3, 4], multiplication [5] and division [6], adders form the core. This paper introduces and analyses various techniques for high speed addition of higher order BCD numbers. A new architecture for the fast decimal addition is proposed, based on which architectures for higher order adders are derived.

The rest of paper is organized as follows: In section 2 details on the proposed algorithm for fast BCD addition. Section 3 discusses about the High speed BCD adder which has been developed using proposed algorithm for fast BCD addition. In section 4 area, timing and power results for reduced delay BCD adder and suggested adder has been presented. Last section presents our conclusion.

2. BCD ADDITION ALGORITHM

The existing algorithm [13] for addition of two BCD digits uses the carry network for the generation of carry for each digit. As the number of digits increase in both BCD digits, the number of

levels in carry network will increase which in turn increases the delay in calculating the final carry out. In the both proposed 16-bit and 32-bit group adders, three signals named C_G , C_P , and P will be computed using input operands. C_G is the carry generate signal, C_P is the carry propagate signal and P shows that the sum of two corresponding digits is greater than or equal to 9. The above three signals are used to generate the Carry output without using the carry network. Signal P to next decimal digit and signal P from previous digit in a decimal number can be used to determine whether carry will be suppressed in corresponding digit location or it will be propagated to the next digit. The output of the carry network and the output of the carry suppression or propagation logic is ORed to generate the value which will be used in the correction step. The equation for value (V) used in the correction step is as follow.

$$V = C_n + (P_n \cdot P_p \cdot C_p) \quad (1)$$

C_n : Output carries to next digit, P_n : Signal P to next digit, P_p : Signal P from previous digit, C_p : Input carry from previous digit.

The digital logic which implements the above algorithm is discussed in the following section.

3. HIGHER VALENCE BCD ADDITION

The existing conventional BCD adder [9] is simple in operation, but very slow due to the ripple carry effect. In BCD additions following cases are considered:

- Case 1: The sum of two BCD digits is smaller than 9.
- Case 2: The sum of two BCD digits is greater than 9.
- Case 3: The sum of two BCD digitals is exactly 9.

For the first two cases, the incoming carry has no effect on determining the output carry. Therefore, the carry output will be independent of carry input. On the other hand for case 3, the output carry will depend on the input carry. Case 2 and Case 3 can be represented by a Carry Generate (CG), Carry propagates (CP) and P signals, respectively.

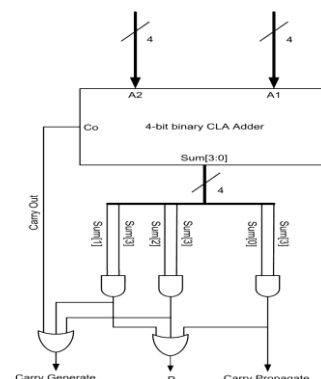


Fig. 1 Carry generate and carry propagate

The computation of the CG, CP and P signals of a digit are shown in Fig 1. The signals CG, CP and P are calculated using

the Sum and Carry outputs of the CLA adder in first stage using following equations.

$$C_p = Sum(3).Sum(0) \quad (2)$$

$$C_g = C_{out} + Sum(3).(Sum(2) + Sum(1)) \quad (3)$$

$$P = Sum(3).(Sum(2) + Sum(1) + Sum(0)) \quad (4)$$

In High speed BCD adder, the carry input is no more connected to the adder block as in reduced delay BCD adder [13], instead it is directly connected to the carry look ahead block which will generate carry to next stage. Because of the above modification the first stage adders will not depend on the carry input. Also the output carry will be independent of the PG logic. Output carry from CLA block is calculated from the following equation.

$$C(i + 4) = CG(i + 4) + P(i + 4) . (CG(i + 3) + P(i + 3) . (CG(i + 2) + P(i + 2) . (CG(i + 1) + P(i + 1) . C_i))) \quad (5)$$

Where, i = 0,4,8,etc

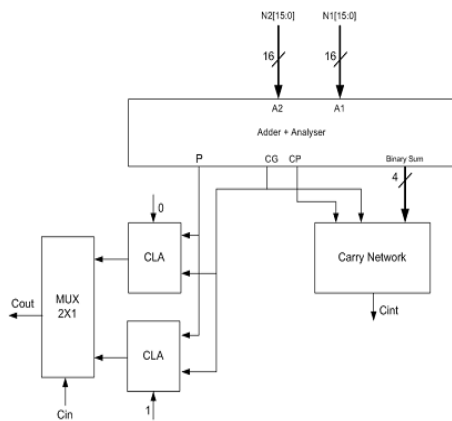


Fig. 2 16-bit Group Adder and analyzer block.

Carry network can be a parallel prefix network which performs their operations in a constant time irrespective of the length of inputs. In this paper Kogge-Stone [8] prefix network is used as carry network. The output of the carry network is used for correction of the output.

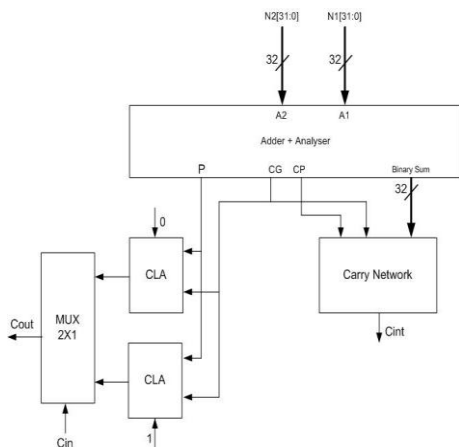


Fig. 3 32-bit Group Adder and analyzer block.

Carry suppressor will take in P value from the present digit, P value from the previous digit and the carry input and determine whether to suppress the carry or to propagate to the next digit. Carry suppressor also generates the correction value which is to be added to the sum value to convert it back to the BCD equivalent.

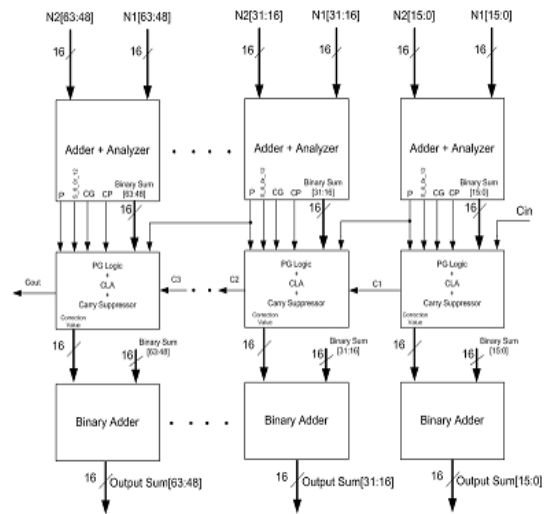


Fig. 4 16-bit Group Higher valence BCD adder

The integrated block diagram of Adder, Analyzer, CLA and Carry network is for both 16-bit and 32-bit group are shown in Figure 2 and Figure 3. The correction to the sum output of first stage adder is done by adding 0, 1, 6 or 7 to it. As in [13], the generation of correction value will depend on the present digit carry output and previous digit carry output. Figure 3a and Figure 3b shows the block diagram for 16-bit and 32-bit group High Speed BCD adders which includes CLA adder in the first stage, combination of PG logic, CLA to generate next stage carry and Carry suppressor in the second stage and Correction logic in third stage.

The number of level in the critical path of 32-bit group High Speed BCD adder is half of that of the 16-bit group High Speed BCD Adder, because of which there will be decrease in the delay.

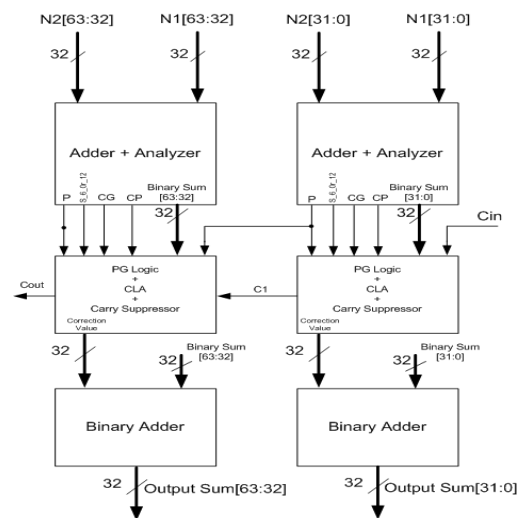


Fig. 5 32-bit Group Higher valence BCD adder

4. RESULTS & DISCUSSION

The reduced delay BCD adder [13], 16-bit High speed BCD adder and 32-bit High Speed BCD adder are designed to support 64-bit and 128-bit decimal addition with BCD operands. They are synthesized using the TSMC 65nm library. Both the adders are implemented in Verilog HDL. They are synthesized using Synopsys Design Compiler synthesis tool and TSMC 65nm library. The synthesis results are presented in Table 1.

According to the Table 1, the proposed High speed BCD adder will take less delay when compared to reduced Delay BCD adder. When the area is under consideration, then High speed BCD adder will occupy more area when compared to reduced Delay BCD adder. It can also be observed that 32-bit group High Speed BCD adder takes less delay.

When compared to both 16-bit Group High Speed Delay and MRDBCD adders. However Delay of MRDBCD is higher than the suggested adders. The high speed addition can be compromised with little area and power overheads.

Table 1 Synthesis result of High speed BCD adder and Reduced Delay BCD adder

Parameters	Group Size for 64-bit		Modified Reduced Delay(64)	Group Size for 128-bit		Modified Reduced Delay(128)
	16-bit	32-bit		16-bit	32-bit	
	High Speed	High Speed		High Speed	High Speed	
Gate Count	552	549	452	1104	1101	950
Area	1352.16	1350.71	1216	2704.31	2701.44	2517.11
Dynamic Power	360.15 μ W	356.50 μ W	365.83 μ W	721.38 μ W	714.88 μ W	748.65 μ W
Cell leakage Power	37.54 μ W	36.86 μ W	35.9 μ W	75.11 μ W	73.76 μ W	73.3 μ W
Delay	0.20 ns	0.13 ns	0.88ns	0.42 ns	0.30 ns	1.14ns

5. CONCLUSION

This paper describes the design of 16-bit and 32-bit high speed BCD adders to perform decimal addition. The new proposed adder has the shortest delay among the decimal adders examined in this paper. The new decimal adder improves the delay of BCD addition by increasing parallelism. The adder has been designed using the Verilog HDL and verified for different corner case inputs. Also the adder has been synthesized with 65nm CMOS library.

6. ACKNOWLEDGMENT

We would like to thank Vidyanand, system administrator for their assistance in VLSI Lab.

7. REFERENCES

[1] W. Buchholz. Fingers or Fists? (The Choice of Decimal or Binary Representation). Communications of the ACM, 2(12):3–11, December 1959.

[2] M. F. Cowlishaw. Decimal Floating-Point: Algorithm for Computers. In Proceedings of 16th IEEE Symposium on Computer Arithmetic, pages 104–111, June 2003.

[3] R.D. Kenney and M.J. Schulte, “Multioperand Decimal Addition,” Proc. IEEE CS Ann. Symp. VLSI, pp. 251–253, Feb. 2004.

[4] R.D. Kenney and M.J. Schulte, “High-speed multioperand decimal adders,” IEEE Transactions on Computers, Page(s):953 – 963, Volume 54, Issue 8, Aug. 2005.

[5] M.A. Erle and M.J. Schulte, “Decimal Multiplication via Carry-Save Addition,” Proc. IEEE 14th

International Conference. Application-Specific Systems, Architectures, and Processors, pp. 348-358, June 2003.

[6] P. Parhami, Computer Arithmetic: Algorithms and Hardware Designs. New York: Oxford Univ. Press, 2000.

[7] S. Hwang. High Speed Binary and Decimal Arithmetic Unit. United States Patent, (4,866,656), September 1989.

[8] P. M. Kogge and H. S. Stone. A Parallel Algorithm for The Efficient Solution of a General Class of Recurrence Equations. IEEE Trans. on Computers, C-22(8), Aug. 1973.

[9] M. M. Mano. Digital Design, pages 129–131. Prentice Hall, third edition, 2002.

[10] M. S. Schmookler and A.W. Weinberger. High Speed Decimal Addition. IEEE Transactions on Computers, C-20:862–867, Aug. 1971.

[11] B. Shirazi, D. Y. Y. Young, and C. N. Zhang. RBCD: Redundant Binary Coded Decimal Adder. In IEEE Proceedings, Part E, No. 2, volume 136, pages 156–160, March 1989.

[12] J. D. Thompson, N. Karra, and M. J. Schulte. A 64-Bit Decimal Floating-Point Adder. In Proceedings of the IEEE Computer Society Annual Symposium on VLSI, pages 297–298, February 2004.

[13] Alp Arslan Bayrakci and Ahmet Akkas. Reduced Delay BCD Adder. IEEE, 2007.

[14] Sundaresan C, Chaitanya CVS. Modified Reduced Delay BCD Adder. IEEE, 2011