International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869 (O) 2454-4698 (P), Volume-3, Issue-10, October 2015

High Speed Pipelined 4 Input Decimal Adder

Rishabh Panday, Himanshu Joshi

***Abstract*— In the present days after increasing the complexity in the computation, internet based applications we need a fast and compact decimal adder which work with less delay and same power consumptions. So we can design a pipelined four input decimal adder using the DG, DP signals and Correction digits. By using the CSA, CLA, PG generator and with a register we reduce the delay of the adder with 46.22% compare to conventional decimal adders by synthesize the simulation in Xilinx software. In our proposed work decimal adder is divided in two parts and a register is used in between that for reduce the delay of critical path. So the pipelined decimal adder can work fast addition of the decimal numbers.**

***Index Terms*—Adder, CSA, CLA, DG & DP generator, Register.**

I. INTRODUCTION

In the past time all the computers and processers and many digital devices used the binary arithmetic such that addition but that type of addition have some approximate errors. For an example take $(0.9)_{10}$= $(0.1110)_2$ this is not show the exact result of the decimal fraction number. It requires the infinite bits for represent the decimal number so for overcome by these types of problems BCD binary coded decimal numbers are used for take the exact result. In the BCD numbers every decimal digit represent in the four bits. By using the BCD, decimal numbers are exactly represented like $(0.9)_{10}$=$(0.1001)_2$. So the results are taken in finite bits.

So we proposed a work on high speed pipelined four input decimal adder. In the BCD adder sums of two digits are generated but when the sums are greater than 9 than in every digit a special correction value $(0110)_2$ is added to the sum of every digit [2]. But by using these methods the delay of addition in the adders is so much.

So our proposed pipelined decimal adder can reduce the delay of the decimal adder with using same area constraints. So we provide the fast addition of the decimal numbers compare to convention decimal adders.

Now with following this in section [2] a brief idea on the convention decimal adders is given. In section [3] our proposed pipelined work is explained with a numerical example. And in section [4] our implementation results and comparison between the conventional adders and pipelined decimal adder is given. In section [5] the summery of our proposed work is given.

Rishabh Panday, Electronics & Communication, Jagannath University, Jaipur, India, 09460344714.

Himanshu Joshi, Assistant Professor, Jagannath University, Jaipur, India, Mobile No. 9414618668.

II. PREVIOUS WORK

A. BCD adder

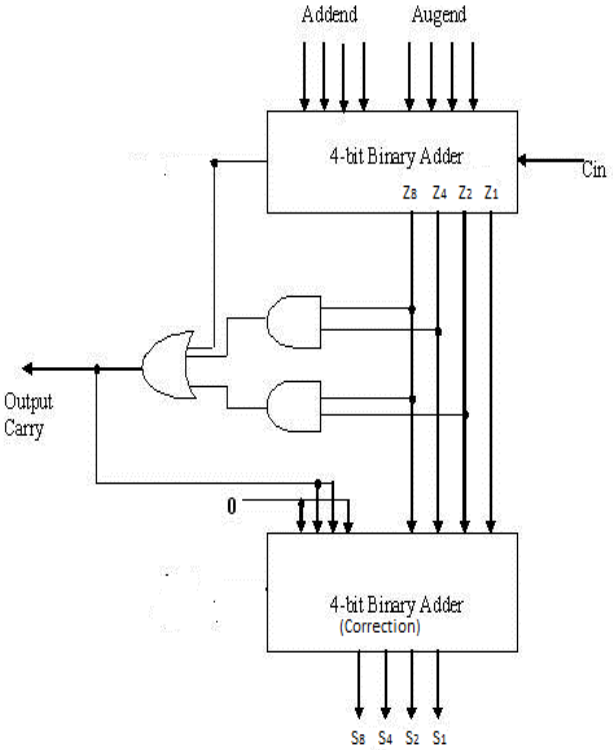

Fig 1: One digit BCD adder

Previously three input decimal adders are proposed. In the addition of the two digits (each 4 bits) one digit BCD adders are used for add the decimal numbers. In that DG and DP signals [1] are used for generate the real carries and then add the sums with the correction logic digits for take the valid results of the decimal numbers.

In a BCD adder we take two decimal digits A and B and take the sums of that with the one digit BCD adder with composed with the four full adders. Now if the sum of the digit is greater than 9 than a correction value 6 $(0110)_2$ is added to the each sum of the digit. For that a carry network is used in the addition. It is used for generate the carries of the decimal digits.

Digit 1	A_3	A_2	A_1	A_0
Digit 2	B_3	B_2	B_1	B_0
Carry	S_3	S_2	S_1	S_0

And CC= carry + $(Z_8.Z_4)$ + $(Z_8.Z_2)$
+ denotes logical OR
. denotes logical AND

So if we increase the number of input in the decimal adder than full adders are also increase in the decimal number and the delay of the addition is also increased by that. So a new type of BCD adder is designed for reduce the delay of the adder.

B. Reduced delay BCD adder

In the second type of adder there are three stages which are used in the reduced delay BCS adders. In the first stage of the adder a block Adder+ Analyzer [2] is used. Two inputs are given to that block. This block is used for sum the digits and generate the DG (Digit Generation) and DP (Digit Propagation) signals. The sums which are generate by this block if more than 9 than DG identify the condition of the sums in the decimal adder. And when the sum is equal to 9 than condition is identified by the DP. Bothe these signals are sent to the second stage which is a carry network. In the carry network real carries are generate by using DG and DP signals.

Carry out = $DG+DP.C_{in}$

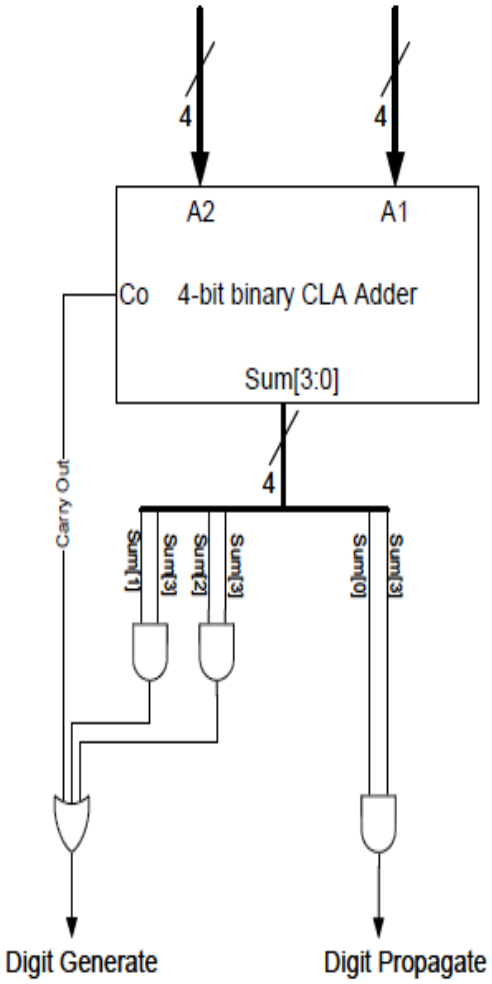

Fig2: Adder + Analyzer

In the third stage the correction values are added to the digit sums for take the valid results.

III. PROPOSED HIGH SPEED PIPELINED FOUR INPUT DECIMAL ADDER

In our proposed pipelined decimal adder three stages are used for add the decimal numbers. each decimal number has four bits. In the first stage of our proposed decimal adder first block consists CSA and DG, DP generator. This block generate the sums of each decimal digit and also DG (digit generate) and DP (digit propagate) signals. These DG and DP signals are sent to the second stage which is carry network. Carry network generate the real carries of the decimal digits.

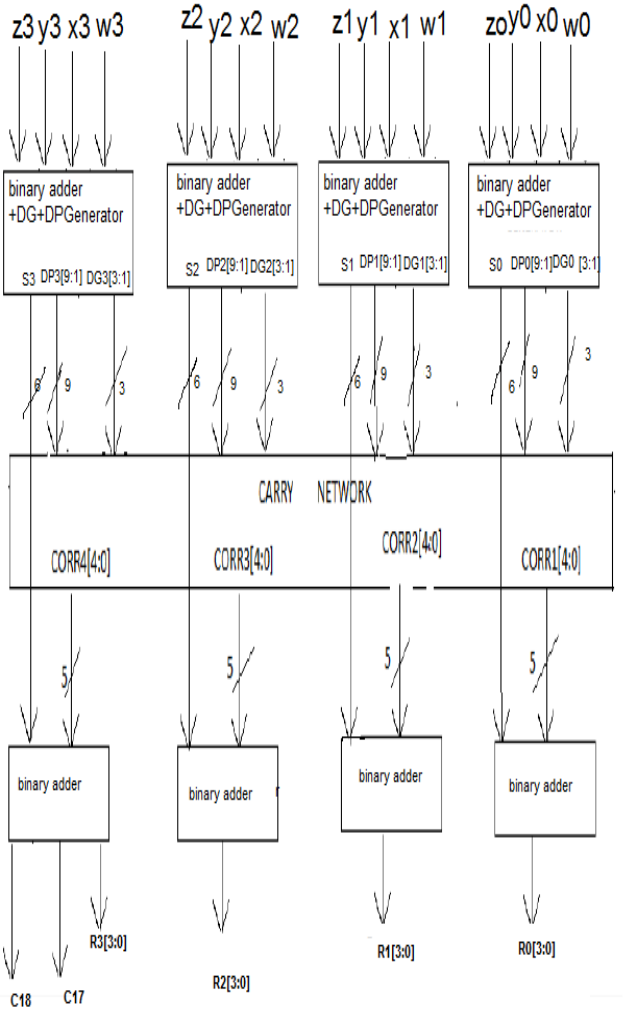

Fig 3: Architecture of four input decimal adder

In the second stage DG signals are identify the conditions for the sums that if sums are not more than 9 or 19 or 29. And DP signals are identify the condition that if sums are equal to 7,8,9,17,18,19,27,28 or 29 and generate these signals. And after that in the third stage a adder is used for add the correction values with the sums of each digit which is generate in the last stage and produce the valid results.

In our proposed pipelined decimal adder also a stage is used before the carry network. In that stage a register is used for store the values of the carries in the decimal adder. Register divides the decimal adder in two parts. By divide the adder in two parts the critical path of the adder is reduced so the delay of the adder is also reduced. A clock signal is given to the register or flip-flop.

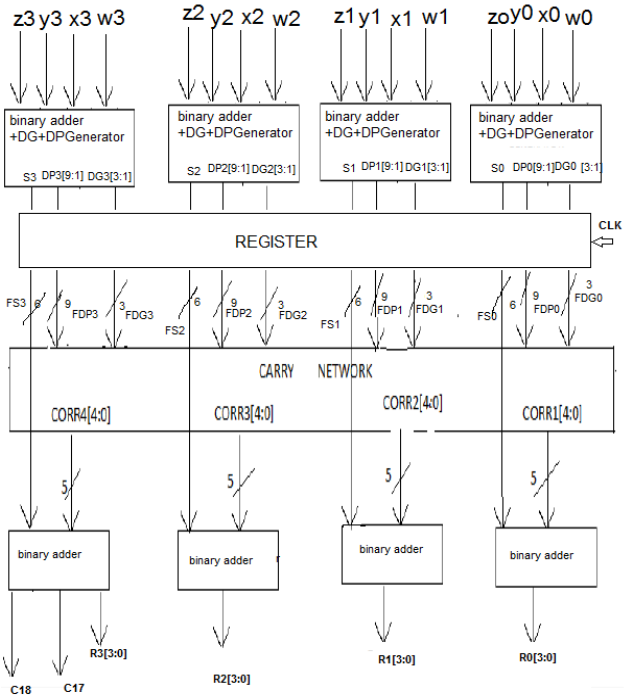

Fig4: Proposed pipelined four input decimal adder

Table1: For DG and DP identifying conditions

Signals	Conditions of the sum in each digit
DG[1]	>9
DG[2]	>19
DG[3]	>29
DP[1]	=7
DP[2]	=8
DP[3]	=9
DP[4]	=17
DP[5]	=18
DP[6]	=19
DP[7]	=27
DP[8]	=28
DP[9]	=29

So by using the register the delay of the path is reduced. And the total delay of the decimal adder is also decreased.

We take a numerical example of four input decimal numbers

9820	W
8974	X
5899	Y
+7476	Z
32169	Result

Digit 4	Digit 3	Digit 2	Digit 1	
1001	1000	0010	0000	W
1000	1001	0111	0100	X
0101	1000	1001	1001	Y
0111	0100	0111	0110	Z
11101	11101	11001	10011	Sum[4]
1	1	1	1	DG[1]
1	1	1	0	DG[2]
0	0	0	0	DG[3]
0	0	0	0	DP[1]
0	0	0	0	DP[2]
0	0	0	0	DP[3]
0	0	0	0	DP[4]
0	0	0	0	DP[5]
0	0	0	1	DP[6]
0	0	0	0	DP[7]
0	0	0	0	DP[8]
1	1	0	0	DP[9]
1	1	1	1	C_{10}
1	1	1	0	C_{20}
1	1	0	0	C_{30}
10101	10100	01101	00110	Corr
11101	11101	11001	10011	Sum[4]
10101	10100	01101	00110	Corr[4]
(110010)	(110001)	(100110)	(11001)	Result

IV. IMPLEMENTATION RESULTS AND COMPARISON

In the designing of pipelined four input decimal adder we use the Xilinx software. In that verilog module is used for the designing. After implementation and simulation of the codes we get the delay of the decimal adder.

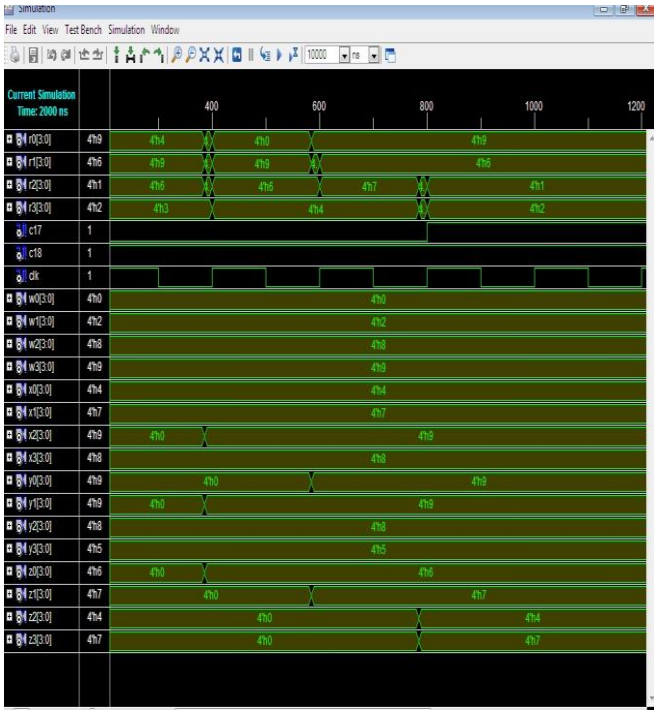

Fig5: Simulation result

By comparing our results with the previously developed decimal adders we can conclude that the delay of our decimal adder is so much reduced. Comparision of our proposed decimal adder with among previously decimal adders are shown below:

Table2: Comparision between adders

Type of adder	Delay(ns)	I/O Buffers
Pipelined 4-input(proposed work)	27.377	82
4-input (type3)	50.908	82
4-input (type2)	63.994	83
4-input (type1)	80.889	83
3-input (type3 previous work)[1]	49.921	66

In the comparison we can see that our proposed pipelined decimal adder reduce the delay in compare to other decimal adders.

V. Conclusion

Our proposed pipelined decimal adder is implemented using the verilog code. Our proposed decimal adder has four inputs with each digit have four bits. Decimal adder is designed using the CSA, CLA adder, DG, DP generator and register. Our designed pipelined four input decimal adder presents the less delay and same power with increase number of inputs in compare to other decimal adders.

Pipelined decimal adder has reduced the delay about 46.22% compare to conventional adders. So this is adding of the decimal numbers in very high speed. And in the future our proposed decimal adder can implemented with more inputs.

Acknowledgment

The completion of that project gives us so much pleasure. I would like to express my sincere gratitude to my project guide **"Mr Himanshu Joshi"** for giving me the opportunity and guideline to work on that. It would never be possible for us to take this project to this level without his innovative ideas and his relentless support and encouragement.

References

[1] Tso-Bing Juang, Hsin-Hao Peng, Chao-Tsung Kuo "Area-Efficient 3-Input Decimal Adders Using Simplified Carry and Sum Vectors" 2011 IEEE/IFIP 19th International Conference on VLSI and System-on-Chip.

[2] A. Bayrakci and A. Akkas, "Reduced delay BCD adder," Proc. IEEE18th International Conference on Application-specific Systems, Architectures and Processors, (ASAP), pp. 266-271, July 2007.

[3] R.D. Kenney and M.J. Schulte, ''High-speed multioperand decimal adders,'' IEEE Transactions on Computers, pp. 953-963, Vol. 54, No.8, Aug. 2005.

[4] G. Bioul, M. Vazquez, J. P. Deschamps, and G. Sutter, "Decimal addition in FPGA," Proc. SPL. 5th Southern Conference on Programmable Logic, pp. 101-108, 2009.[11] A. Vazquez and E. Antelo, "A High-Performance Significant BCD Adder with IEEE 754-2008 Decimal Rounding," Proc. 19th IEEE Symposium on Computer Arithmetic (ARITH-19), pp. 135-144,2009.

[5] A. Vazquez, E. Antelo and P. Montuschi, "A New Family of High Performance Parallel Decimal Multipliers," Proc. of the 18th IEEE Symposium on Computer Arithmetic (ARITH-18), pp. 195-204, 2007.

[6] G. Jaberipur and A. Kaivani, "Improving the Speed of Parallel Decimal Multiplication," IEEE Transactions on Computers, Vol. 58,No. 11, pp. 1539-1552, Nov. 2009.

[7] M. S. Schmookler and A.W.Weinberger. ''High Speed Decimal Addition,'' IEEE transactions on Computers, Vol. 20, No. 8, pp.862--867, Aug. 1971.

[8] Song Yu-yun, Hu Qing-sheng "A O.18J.lffi Pipelined 8BIOB Encoder for a High speed SerDes" New Star Research Institute of Applied Technology, No.451 , Huangshan Road, Hefei, Anhui.

[9] Rahul Jain, Khushboo Singh, Ghanshyam Jangid "A 64 Bit Pipeline Based Decimal Adder Using a New High Speed BCD Adder" International Journal of Science and Research (IJSR).

[10] R. K. James, T. K. Shahana, K. P. Jacob, and S. Sasi, "Decimal multiplication using compact BCD multiplier," Proc. International Conference on Electronic Design (ICED), pp. 1-6, 2008.

[11] I. S. Hwang. High Speed Binary and Decimal Arithmetic Unit. United StatesPatent, (4,866,656), September 1989.

[12] A. Vazquez, E. Antelo and P. Montuschi, "Improved Design of High-Performance Parallel Decimal Multipliers," IEEE Transactions on Computers, Vol. 59, No. 5, pp. 679-693, May 2010.

[13] 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.

[14] T. Lang and A. Nannarelli, "A Radix-10 Digit-Recurrence Division Unit: Algorithm and Architecture," IEEE Transactions on Computers, Vol. 56, No. 6, pp. 727-739, June 2007.

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

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

[17] 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.

[18] L. -K. Wang, M. A. Erle, C. Tsen, E. M. Schwarz and M. J. Schulte, "A survey of hardware designs for decimal arithmetic," IBM Journal of Research and Development, Vol. 54, Issue 2, pp. 8:1-8:15, 2010.

[19] E. Cornea, J. Harrison, J. C. Anderson, P. Tang, E. Schneider, and E. Gvozdev, "A Software Implementation of the IEEE 754R Decimal Floating-Point Arithmetic Using the Binary Encoding Format," IEEE Transactions on Computers, Vol. 58, No. 2, pp. 148-162, Feb. 2009.

[20] M. F. Cowlishaw, "Decimal Floating-Point: Algorism for Computers," Proc. of 16th IEEE Symposium on Computer Arithmetic (ARITH-16), pp. 104–111, June 2003.

[21] Draft IEEE Standard for Floating-Point Arithmetic. New York: IEEE, Inc., 2004.

[22] R.D. Kenney and M.J. Schulte, ''Multioperand Decimal Addition,''Proc. IEEE Computer Society Ann. Symp. VLSI, pp. 251-253, Feb.2004.

Rishabh Panday, Student of M.Tech (Embedded System) in jagannath university, Jaipur, India. I am completed B.Tech (Electronics & Communication) degree in 2012 from Rajasthan Technical University.

Himanshu Joshi, Assistant Professor Department of ECE in Jagannath University, Jaipur, India. He has completed his M.Tech (VLSI and Embedded system) in 2011 from Gyan Vihar University, Jaipur, and B.E degree in 2007 from Rajasthan University. He is currently working in the VLSI and Communication Field.