ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Design and Implementation of Carry SelectAdder without Using Multiplexers

S.Karishma Sri
M.Tech Student,
Department of ECE,
KITS for Women's, kodad, T.S, India.

Mrs.D.Jyothi
Associate Professor,
Department of ECE,
KITS for Women's, kodad, T.S, India.

ABSTRACT:

Design of high performance digital adder is an importantrequirement in advanced digital processors for faster computation. In digital adder circuits, the speed of addition is limited by the time required for a carry to propagatethrough the adder. Many different approaches had already been suggested to improve the performance of the adder. Carry Select Adder (CSA) is one among them and is used to solve the problem of carry propagation delay by independently generating multiple carries and then select a carry to generate the final sum In CSA, requirement of producing two adders and final selection multiplexers make it consuming more area, even though carry propagation delay is reduced much. Parallel prefix adders can also be used to reduce the delay.Several examples of such adders have been published and there are many efficient implementations. Kogge and Stone scheme limit the lateral logical fan-out at each node to unity. This paper discusses about the implementation of Carry Select Adder without using MUX for final selection. Our approach uses first, the implementation of cin=0 adder and then Excess 1 adder. Excess 1 adder is designed in such a way that it becomes a first zero finding logic and replaces the final MUX stage used in traditional approach. Parallel adder configuration is also used to reduce the delay between stages. Removing the MUX stage will reduce the area as well as propagation delay to give much higher performance for the adder. The Kogge Stone parallel approach will give option to generate fast carry for intermediate stages. The adder is implemented on Spartan 3E FPGA and is compared with CSA with MUX, Kogge Stone adder. Results show that the new addergives reduced area and better speed compared to other adders.

Keywords: CSA,MUX.

1. INTRODUCTION:

This approach will reduce the problem of existing scheme and CSA [2] is one among them which will reduce the carry propagation delay problem. In CSA, requirement of producing two adders and final selection multiplexers make it consuming more area, even though carry propagation delay is reduced much. Buffering inverters are to be added appropriately to support these large loads and there is a corresponding increase in the delay. Brent & Kung [3] proposed the fan-out trees such that the lateral fan-out of each node is restricted to unity, as for the Kogge Stone graph, but without the explosion of wires. Although looks attractive it increases the logical depth. This illustrates the approach of carry select adder implementation to achieve minimum delay and reduced area without increasing the fanout In CSA, requirement of producing two adders and final selection multiplexers make it consuming more area, even though carry propagation delay is reduced much. Parallel prefix adders can also be used to reduce the delay. Several examples of such adders have been published and there are many efficient implementations. Kogge and Stone scheme limit the lateral logical fan-out at each node to unity, but at the cost of a dramatic increase in the number of lateral wire at each level. Ladner and Fischer introduced the minimum depth prefix graph.

The longest lateral fanning wires go from a node to n/2other nodes. Capacitive fan-out loads become large for later levels in the graph as increasing logical fan-out combines with increasing span of the wires. Buffering inverters are to be added appropriately to support these large loads and there is a corresponding increase in the delay. Brent & Kung proposed the fan-out trees such that the lateral fan-out of each node is restricted to unity, as for the Kogge Stone graph, but without the explosion of wires.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Although looks attractive it increases the logical depth. Han and Carlson give a good overview of prefix addition formulations, and present their own hybrid synthesis of the Ladner- Fischer and Kogge-Stone graphs. Again this trades an increase in logical depth for a reduction in fan-out. Kowalczuk, Tudor & Mlynek achieve a similar compromise by serializing the prefix computation occurring at the higher fan-out nodes and Beaumont-Smith & Burgess combined this idea with the Han-Carlson scheme. All these latter papers allow the logical depth, and hence the delay increase in exchange for reductions in fan-out or wire flux. Knowles showed that fan-out and wire flux gains are available without increasing logical depth from the minimum used in the Ladner-Fischer and Kogge-Stone structures. There are many publications available that compares between different parallel tree adders and shows the advantage of these adders in terms of area, fanout and wire tracks. There are many carry select adder approaches available but most of them use ripple carry adders to implement the adder.

Behnam Amelifard et.al, suggested a new adder called carry select adder with sharing (CSAS) which is area efficient but the delay is more. M. Alioto et.al suggested using variable size block sizing depending on the MUX delay. Some papers suggested using add one circuit to eliminate the second adder required for the CSA with cin=1 condition. compares different parallel prefix adders in IBMs EAC adder. Authors showed that Han Carlson and Knowles configurations are best compromise between speed and area. We have shown that CSA implemented with parallel prefix adder approach can give better delay and area performance This paper illustrates the two different approaches of carry select adder implementation to achieve minimum delay and reduced area without increasing the fan-out or lateral wires.

2. METHODOLOGY:

RIPPLE CARRY ADDER There are many carry select adder approaches available but most of them use ripple carry adders [1] to implement the adder. Disadvantages of existing system Delay is more. It requires more area. Power consumption is more In electronics, an adder or summer is a digital circuit that performs addition of numbers. In many computers and other kinds of processors, adders are used not only in the arithmetic logic unit(s), but also in other parts of the processor, where they are used to calculate, table indices, addresses and similar operations.

Although adders can be constructed for many numerical representations, such as binary-coded decimal or excess-3 the most widely recognized adders work on binary numbers. In cases where two's complement or ones' complement is being used to represent negative numbers, it is trivial to modify an adder into an added. Other signed number representations require a more complex adder. The adder we are using here is a ripple carry adder. It is possible to create a logical circuit using multiple full adders to add N-bit numbers. Each full adder inputs a Cin, which is the Cout of the previous adder. This kind of adder is called a ripple-carry adder [5], since each carry bit "ripples" to the next full adder. Note that the first (and only the first) full adder may be replaced by a half adder. The ripple carry adder is implemented using full adder as the basic building block.

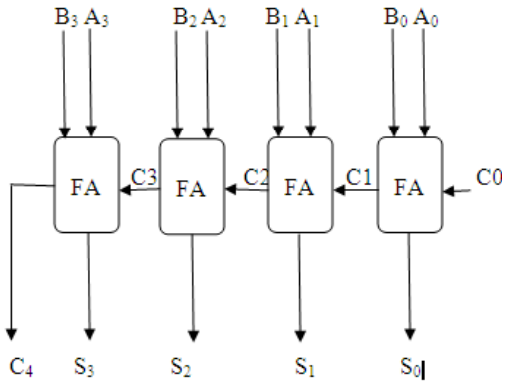


Fig.1. 4 Bit Ripple Carry Adders

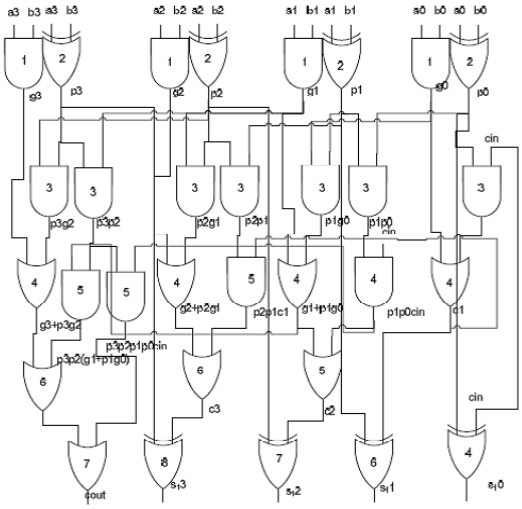
The Fig.1 shows 4 bit ripple carry adder which is constructed with using 4 full adders. The input carry in the least significant position is 0. Each full adder receives the corresponding bits of A and B and the input carry and generate the sum bit for S and the output carry. The output carry in each position is the input carry of the next-higher-order position. It can be inferred, that for N bit addition, the proposed ripple carry adder architecture uses only N reversible gates and produces only 3N garbage outputs. But, the ripple carry adder [4] using our proposed gate (PG) is the most optimized one.

3. IMPLEMENTATION:

To generate the fast carry for intermediate stages by using kogge-stone adder. The carry propagation delay of adder is proportional to log2 (n) and the number of logic elements used is proportional to nlog2 (n), where n is the number of bits used in addition.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

In this the carry select adder is implemented It mainly focus on to improve the performance of adder with reduced area, high speed and low power consumption. In digital adder circuits, the speed of addition is limited by the time required for a carry to propagate through the adder. Carry Select Adder (CSA)[6] is one among them and is used to solve the problem of carry propagation delay by independently generating multiple carries and then select a carry to generate the final sum.



The above analysis is the result of the 4bit with KS CSA adder in which the inputs are 'a','b' and Cin and the output is 's' and carry out is Cout . The input is taken as a=Fh'4 and b=6h'4 and the output we get is s=5h'4 and the carry out Cout=1. The above analysis is the result of the 16bit KS CSA adder in Fig.5 which the inputs are 'a' and 'b' and the output is Sout and carry out is Cout. The input is taken as a=FFFF h'4 and b=0000 h'4 and the output we get is s=FFFF h'4 and the carry out Cout=0. ii) CSA Zero finder From Fig.6 the input is taken as a=1111 and b=0000 and the output we get is S=0000 and the carry out Cout. Kogge-Stone (KS) adder is a parallel prefix form Look ahead Adder [2]. Kogge-Stone adder can be represented as a parallel prefix graph consisting of carry operator nodes. The time required to generate carry signals in this prefix adder is proportional to log n. It is the fastest adder with focus on design time and is the common choice for high performance adders in industry.The better performance of KoggeStone adder is because of its minimum logic depth and bounded fan-out. On the other side it occupies large silicon area. The carry equations of KS adder are shown below. The carry propagation delay of the adder is proportional to log2 (n) and the number of logic elements used is proportional to nlog2 (n), where n is the number of bits used in addition and has 8 units delay. For delay calculation XOR is considered to have two units delay and all the other elements have single unit delay. Carry input for the intermediate stages can be realized using Kogge Stone approach and by doing so we can eliminate the carry propagation through multiplexers as in CSA. CSA has lesser area utilization compared to KS adder. Area utilization of CSA can still be reduced by realizing the Cin=1 adder from Cin=0 adder, instead of going for a separate adder. Two different such realizations are discussed in the next section.

4. DISCUSSION:

Fig. 3 shows the proposed method of implementation of CSA. Here instead of using simple Excess ladder, first zero finding logic is used.

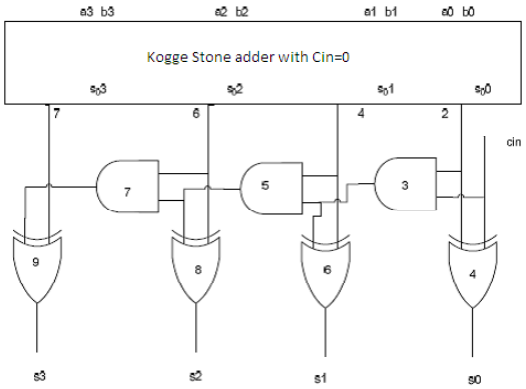
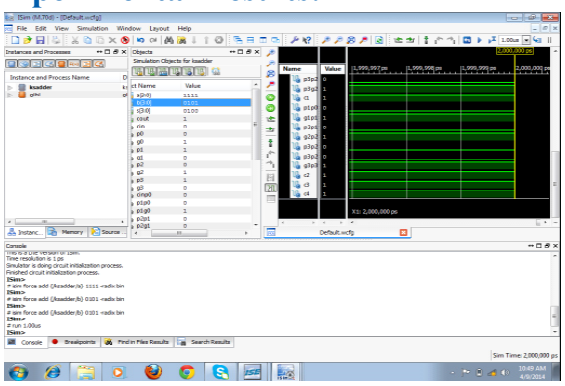


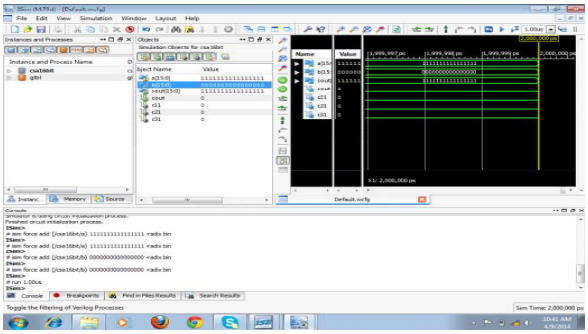
Fig.3. Carry Select Adder with KS adder (c in =0). If Cin =0, logic will select the KS Adder output as the final output from LSB to MSB until it finds the first occurrence of a zero. The proposed adder uses lesser logic elements compared to CSA with MUX. A) KS Adder The below analysis is the result of the KS ADDER in Fig.4 which the inputs are a, b and Cin. The output is s and Cout. The input is taken as a=1111 and b=0101 and the output we get is s= 0100 and carry out Cout =1.

5. Experimental Results:

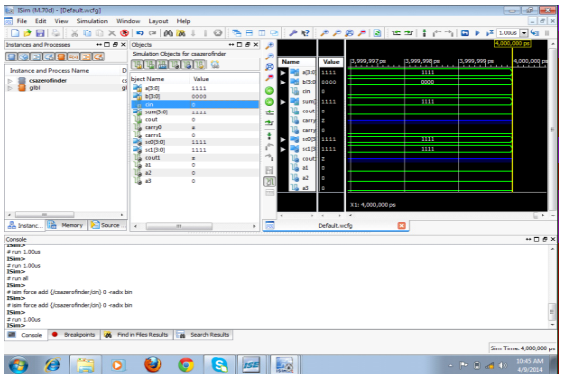


ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

KS Adder.



CSA Zero finder.



6.CONCLUSION:

In this paper we have shown the design of carry select adder implemented with Kogge Stone tree using two different approaches. One approach uses Excess1 adder and the other uses first zero finding logic to realize the carry select adder. Both the adders are implemented on Spartan XC3-S500EFPGA device and the performance is compared. CSA with MUX performs better in terms of delay and CSA without MUX performs better in terms of area. CSA without MUX performs slightly better compared to CSA with MUX when the area-delay product is taken. FPGA inbuilt adder is alsoused for comparison. This is to show the closeness of the performance of the optimized adders with FPGA adder.

7.ACKNOWLEDGMENTS:

I am S.Karishma Sri and would like to thank the publishers, researchers for making their resources material available. I am greatly thankful to Assistant Prof: Mrs.D.Jyothi for their guidance. We also thank the college authorities, PG coordinator and Principal for providing the required infrastructure and support. Finally, we would like to extend a heartfelt gratitude to friends and family members.

8.REFERENCES:

[1] O. J. Bedrij, “Carry-select adder,” IRE Trans. Electron. Computer.,pp.340–344, 1962.

[2] J. Sklansky, “Conditional-Sum Addition Logic” IRE. Transactions onElectronic Computers, vol. EC-9, pp. 226-231, 1960.

[3] P.M. Kogge, H.S. Stone; “A Parallel Algorithm for the Efficient Solutionof a General Class of Recurrence Equations”IEEE Trans., C-22(8):786-793, Aug. 73.

[4] R.E. Ladner, M.J. Fischer; “Parallel Prefix Computation”JACM,27(4):831-838, Oct. 80.

[5] R.P. Brent, H.T. Kung; “A Regular Layout for Parallel Adders” IEEETrans., C-31(3):260-264, March 82.

[6] T. Han, D.A. Carlson; “Fast Area-Efficient VLSI Adders”8th IEEESymp. Computer Arithmetic, Como Italy, pp. 49-56, May 87.

[7] J. Kowalczuk, S. Tudor, D. Mlynek; “A New Architecture for AutomaticGeneration of Fast Pipelined Adders”ESSCIRC, Milano Italy, pp. 101-104, Sept 91.

[8] A. Beaumont-Smith, N. Burgess; “A GaAs 32-bit Adder”13th Symp.Computer Arithmetic, hilomar California, pp. 10-17, June 97.

[9] S. Knowles, “A family of adders,” Proceedings of the 14th IEEE Symposiumon Computer Arithmetic, April 14-16 1999, adelaide, Australia.

Author’s Details:

Ms.S.Karishma Sri. MTech student, in M.Tech Student, Dept of ECE in KITS for women’s,kodad, T.S, India.

Mrs.D.Jyothi. working as a Assistant at ECE in KITS for women’s,kodad, T.S, IndiaJNTUH Hyderabad. she has 6 years of UG/PG Teaching Experience.