OPTIMIZED DESIGN OF AM OPERATOR USING MODIFIED KOGGESTONE ADDER

Smitha U

ME student, Department of Electronics and communication, SVCE, Madhya Pradesh, India

ABSTRACT

Multipliers are the inevitable component in most of the Digital Signal Processing applications. Multipliers with High speed and low area are of substantial importance from the view point of research. To obtain high speed, the multiplier utilizing modified koggestone adders are implemented in this work. The design and implementation of Add-Multiply (AM) unit constituting modified koggestone block improves the overall performance of the multiplier. The conventional AM unit using Modified Booth (MB) multiplier make use of CSA and CLA adders which limits its performance due to delay involved and hardware complexity. Fast adder introduced in the design speeds up the entire process of multiplication. It further reduces the area occupancy and considerable power improvements are achieved. The speed of conventional multiplier design can be enhanced by employing modified koggestone adder in the AM design and it attempts to reduce the delay of conventional design using CSA and CLA adder trees. Therefore an area effective and speed optimized AM unit is proposed in order to overcome the drawbacks of conventional design. The analysis of both existing and proposed techniques are clearly shown. AM units are simulated using Xilinx ISE 14.7 for performance analysis.

Keyword: - *Modified koggestone adder, Modified Booth, CSA, CLA, Add Multiply*.

1. INTRODUCTION

Digital Signal Processing involves manipulation of the signals and data numerically in the digital form. The basic operations involved are addition, subtraction, multiplication using which variety of signals can be produced. Area of the adders and multipliers affect the performance of the Digital Signal Processing systems. So in order to achieve efficient performance of the DSP system improving the speed of the multipliers and adders is one of the effective measure. In digital signal processing applications, operations mostly depends on additions, multiplications. Increased area consumption and delay of multiplier typically hinders its performance, therefore a good design of multiplier requires that it is a designed with smaller area and reduced delay. The multiplication operation entails three major steps: (i) Generation of Partial products (ii) Reduction of partial products and (iii) Computation of final product. The partial products reduction stage is the one which is responsible for the overall speed of the multiplier. This stage mainly contributes to the overall delay, area and the power. Generally adders are used in the reduction process to minimize the long delay in the path.

Earlier the multiplication was generally implemented as a sequence of addition, subtraction, and shift operations. Multiplication can be considered to be series of repeated additions. The number to be added is the multiplicand, the number of times it is added to the multiplicand is the multiplier, and the result is the product. This process of generates partial product. Different adders like carry select adder, carry look ahead adder, ripple carry adder structures are commonly used in the design of multipliers. But the necessity to reduce the area, delay, and the power consumed to optimize the design causes use of new adder architectures which can be used as a substitute for former adders.

2. LITERATURE REVIEW

Kostas Tsoumanis theorized that the straightforward design of the AM unit, by first allocating an adder and then driving its output to the input of a multiplier, increases significantly both area and critical path delay of the circuit. Targeting an optimized design of AM operators, fusion techniques are employed based on the direct recoding of the sum of two numbers equivalently a number in carry-save representation in its Modified Booth (MB)

form. He focused on the efficient design of FAM operators, targeting the optimization of the recoding scheme for direct shaping of the MB form of the sum of two numbers.

Riboy Cherian described that the three bits are needed to form the modified booth digit and also the most significant bit among them is negatively weighted and the other two bits have positive weights. Hence a signed bit arithmetic is needed in order to transform the above mentioned pairs of bits to modified booth form. Hence by considering the inputs to be signed a bit level signed half adder and full adder was developed. He used carry save adder to accumulate the partial products resulting in the final sum and carry. It improved the speed of accumulation of the partial product since it saves the carry and give it to the next level of carry select adder. He explained that the adders in the same layer becomes independent of each other and can be executed simultaneously. Hence the time required for the addition operation is reduced. To improve the speed of the accumulation process a Wallace carry save adder is used. In Wallace carry select adder first three partial products are given to the first carry select adder. The next three partial products are given to the next carry select adder and so on. After accumulation of partial products the result will be a final sum and carry.

Vadithe Madhu Bai explained that after pre-processing step the recoder is needed in order to handle operands in carry save representation the technique i.e. direct recoding of sum of two numbers in its MB form leads to more efficient implementation of fused Add-multiply as compared to the conventional one. She proposed some new techniques which decreases critical delay, power consumption and area. She explored three alternative schemes of the proposed S-MB algorithm approaches using conventional and signed bit Full Adders (FAs) and Half Adders (HAs) as building blocks. She showed that the adoption of the proposed recoding technique gave the optimized solutions for the FAM design.

Bipin and Sakshi in 2013 compared the performance of the multiplier using different adders like ripple carry adder, carry look ahead adder and carry select adder also he employed an efficient technique to find 2's complement and to generate regular partial products. It has been demonstrated that the significant improvement is achieved in terms of power consumption, area and delay using proposed MBE with regular partial product arrays as compared with those of conventional multiplier.

Laxman Shanigarapu in 2012 proposed that the problem of carry propagation delay is overcome by independently generating multiple radix carries and using these carries to select between simultaneously generated sums and introduced a scheme to generate carry bits with block carry in 1 from the carries of a block with block carry in 0. The design of carry select adder was implemented by using D-Latch and compared with regular CSA and modified CSA (BEC and without using Multiplexer).

Veena V Nair in 2013 proposed to reduce the area and power of SQRT CSLA architecture. The reduced number of gates of her work offered the great advantage in the reduction of area and also the total power. The modified CSLA reduced the area and power when compared to regular CSLA with increase in delay by the use of Binary to Excess-1 converter. The proposed scheme reduced the delay, area and power than regular and modified CSLA by the use of D-latches.

Minu Thomas in 2014 proposed work for the design and simulation of Radix-8 Booth Encoder multiplier for signed-unsigned numbers. The Radix-8 Booth Encoder circuit generated n/3 the partial products in parallel. By extending sign bit of the operands and generating an additional partial product of the Radix-8 Booth Encoder multiplier was obtained. The Carry Save Adder (CSA) tree and the final Carry Look ahead (CLA) adder used to speed up the multiplier operation. Since signed and unsigned multiplication operation was performed by the same multiplier unit the required hardware and the chip area reduces and this in turn reduces power dissipation and cost of a system.

3. PROPOSED AM (ADD-MULTIPLY) UNIT DESIGN

The multiplication process is simplified in 3 steps:

1. First step: Multiplier bits are divided in groups then these groups are fed to encoder.
2. Second step: Here operations are applied on the multiplicand and it will generate the partial products.
3. Third step: Now generated partial products are added with adders.

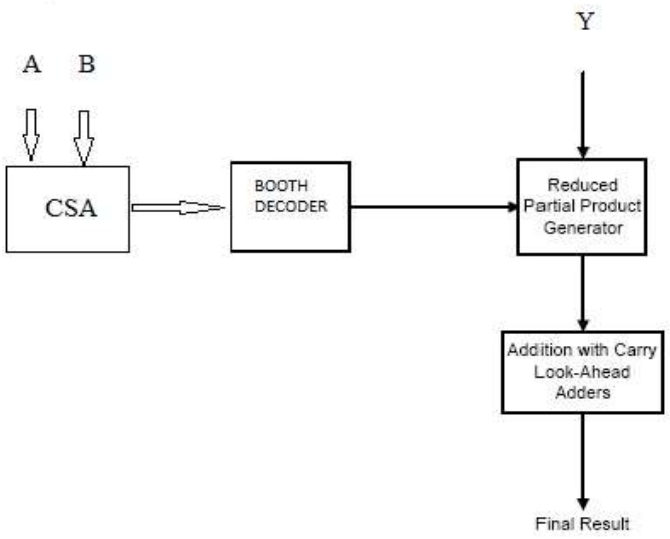

Fig-1: Conventional Add-Multiply unit using CLA and CSA adder

The multiplier is considered for grouping in blocks of four. One bit in each block overlaps the other block. From the LSB, the grouping starts, and the first block only uses three bits of the multiplicand. Each block is then decoded to generate the correct partial products. The encoding of the multiplier Y, based on modified booth, then generates the following five signed digits -2, -1, 0, +1, +2 respectively.

After generation of these partial products, for adding them, adders are used. In this paper we have used modified kogge-stone adder. It reduces the propagation path delay and increases the performance of the system, which is mostly required in digital systems. Fig-1 illustrates the conventional method of multiplication process which makes use of CLA and CSA

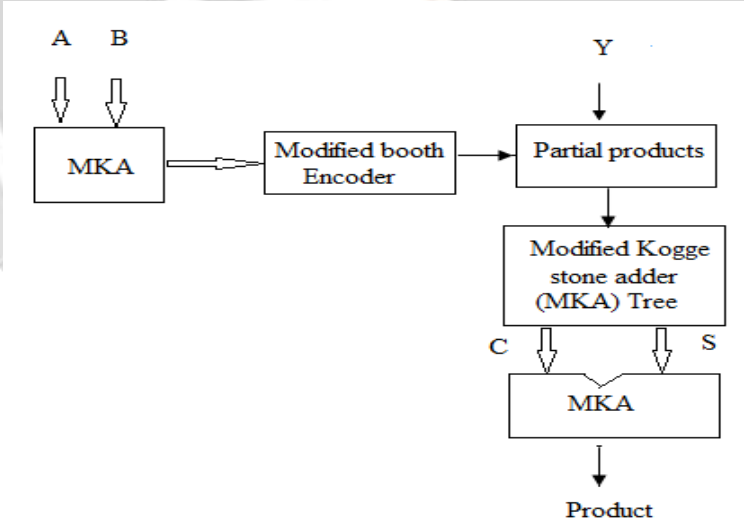

Fig-2: Proposed Add-Multiply unit using Modified koggestone adder

The proposed Add-Multiply unit is shown above in Fig-2. Modified koggestone adders are used initially to obtain X=A+B. To implement P=X.Y,then encoding of Y is done based on the encoding scheme in the Table-1. Then the partial products are generated using the logic circuit shown in Fig-4. Finally partial products are added using Modified koggestone adder.

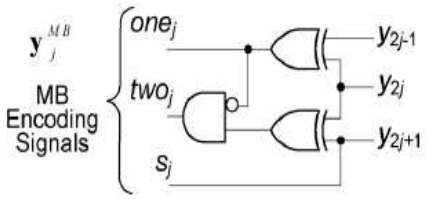

Fig-3: Gate-level schematic for the implementation of the Modified Booth encoding signals

Table-1: Encoding scheme

Binary			y_j^{MB}	*MB Encoding*			Input carry $C_{in,j}$
y_{2j+1}	y_{2j}	y_{2j-1}		Sign=s_j	x1=one_j	x2=two_j	
0	0	0	0	0	0	0	0
0	0	1	+1	0	0	0	0
0	1	0	+1	0	1	0	0
0	1	1	+2	0	1	1	0
1	0	0	-2	1	0	1	1
1	0	1	-1	1	0	0	1
1	1	0	-1	1	1	0	1
1	1	1	0	1	1	0	0

Table-1 shows the encoding scheme used and Fig-3 shows the gate level schematic for the implementation of modified booth encoding signals.Fig-4 shows the gate level schematic for the generation of partial products.

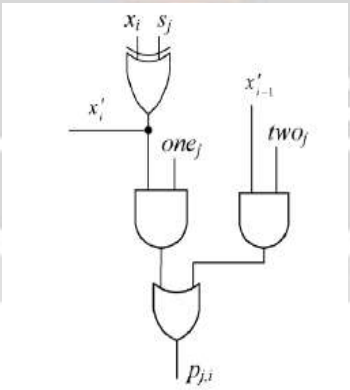

Fig-4: Generation of the i^{th} bit of the partial product

3.1 MODIFIED KOGGESTONE ADDER

The Kogge-Stone adder is faster than other well-known adders and it has a fan-out of 2 in all stages. By eliminating the redundant cells the computation can be minimized. The Kogge-Stone adder can be further modified to compensate the functionality by rerouting and reducing the Black cells. Propagate-Generate (PG) network of modified 8-bit Kogge-stone adder is shown in Fig-5. Since the area do not change if we reroute the wires in

modified kogge-stone adder it's not very efficient way of improving the delay. By eliminating the black cells further we can achieve increased speed and also reduced area.

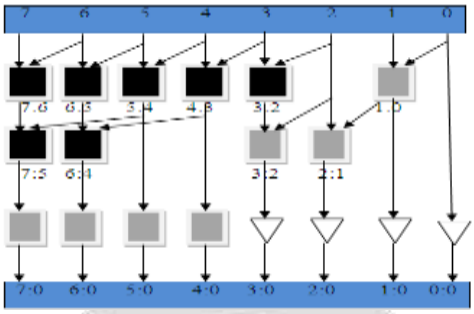

Fig-5: Modified Kogge-stone adder

4. RESULTS

The Fig-6 and 7 shown are the RTL schematic of the proposed MB encoder and modified koggestone adder and Add Multiply unit using modified koggestone adder.

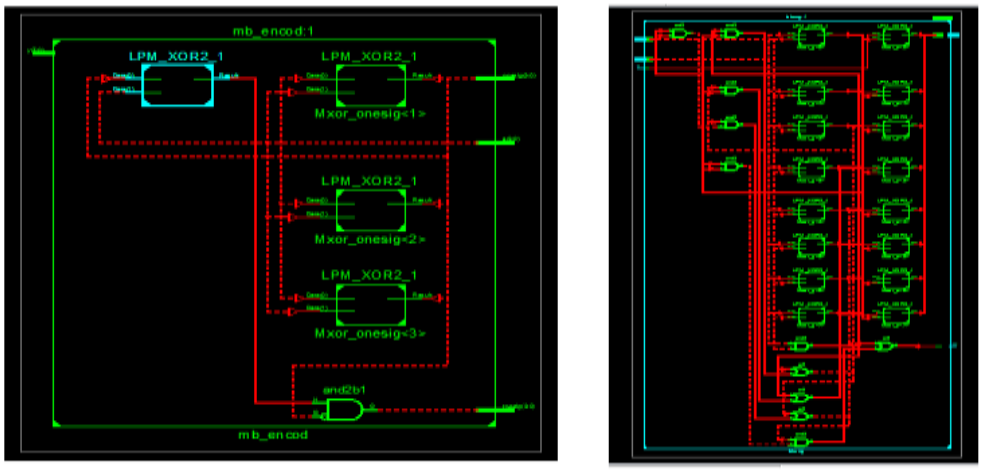

Fig-6:RTL schematic of MB encoder and Modified koggestone adder

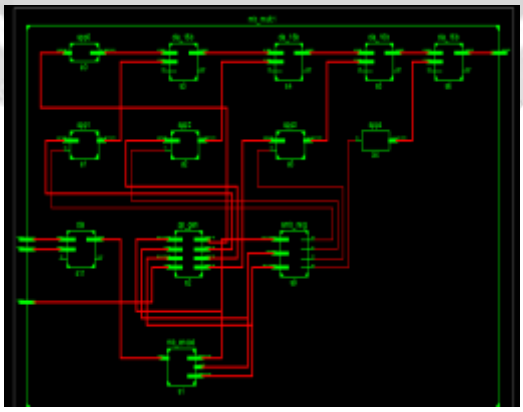

Fig-7:RTL of proposed Add-Multiply unit

The design is implemented in Verilog HDL and synthesized in Xilinx ISE 14.7.The results are verified and studied to check the performance of the design in terms of area and delay.

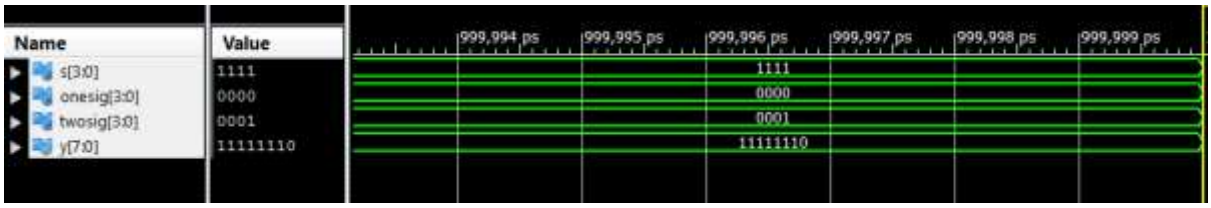

Fig-8:Waveforms of MB encoder

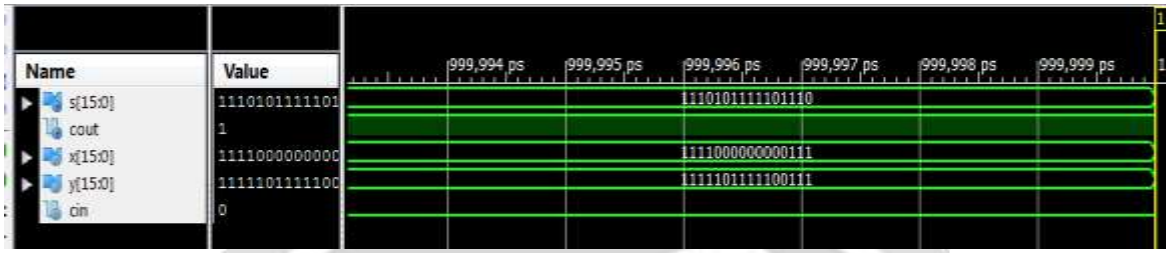

Fig-9: Waveforms of Modified koggestone adder

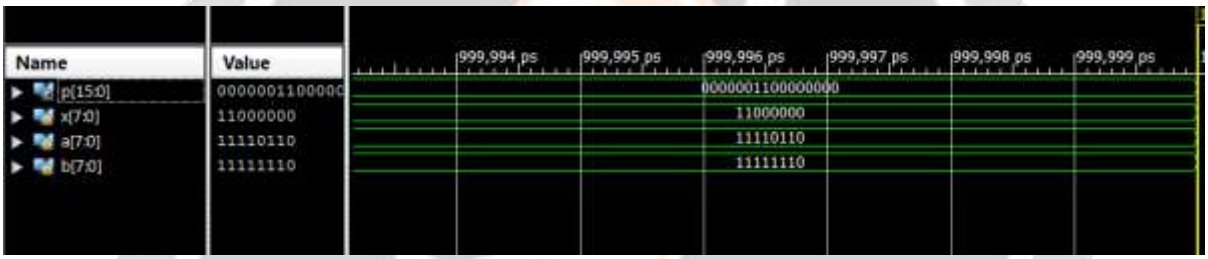

Fig-10 : Waveforms of Modified Booth multiplier using Modified koggestone adder

Comparison has been carried out by synthesizing the designs of Add-Multiply unit employing Carry select adder, Carry look ahead adder and Modified kogge-stone adder and results are tabulated below in Table-2. It clearly shows the area, delay, power-delay product of the design using conventional carry select adder, CLA and proposed Modified kogge-stone adder. The tabular result verifies that by using modified kogge-stone adder the design has speeded up compared to the CSA and CLA. Also the area occupied is reduced.

Table-2: Comparison of power, delay, area of Add-Multiply unit using conventional and proposed adders

Adder Used in multiplier	CSA	CLA	Modified Koggestone
AM operator Area(LUT)	177	143	132
AM operator Delay(ns)	32.244	33.03	28.972
Power*Delay(J)	1.9346	1.98	1.738

5. CONCLUSION

Rapid changes are taking place in Digital Signal Processing systems and the major changes reflect upon optimization concerning the speed, power and delay improvements leading to an efficient design.

Conventional design consisting of CLA and CSA adder that cause delay and area consumption are modified to overcome drawbacks with the help of using modified kogge-stone instead CLA and CSA adders. Overall performance of add multiply unit is improved. Comparison has been made between power, delay, area of the design using CSA, CLA and Modified kogge-stone and results are verified.

6. REFERENCES

[1] An Optimized Modified Booth Recoder for Efficient design of the Add-Multiply Operator, Kostas Tsoumanis, Student Member, IEEE, Sotiris Xydis, 2014.

[2] High speed adder multiplier unit with S-MB recoding, Riboy Cherian, International journal of advanced research in Electrical and Instrumentation Engineering, 2015.

[3] An Efficient Modified Booth Recoder for Different Applications, Vadithe Madhu Bai, IEEE, Kakinadu, 2016.

[4] Design and Comparison of Regularize Modified Booth Multiplier Using Different Adders, Bipin Likhar, Sakshi, IEEE, 2013.

[5] Low power high speed carry select adder, Laxman Shanigarapu, Bhavana P, Shrivastava, International Journal of Scientific and Research Publications, Volume 3, Issue 8, August 2013.

[6] Modified Low-Power and Area-Efficient Carry Select Adder using D-Latch by Veena V Nair, International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 2, Issue 4, July 2013.

[7] Design and Simulation of Radix-8 Booth Encoder Multiplier for Signed and Unsigned Numbers, Minu Thomas, IJIRST–International Journal for Innovative Research in Science and Technology Vol. 1, Issue 1, June 2014.