International Journal of
Engineering Research and Science & Technology

IJERST

www.ijerst.com

Research Paper

COMPARATIVE ANALYSIS OF MODIFIED BOOTH RECODER WITH MODIFIED WALLACE MAC FOR POWER CRITICAL APPLICATIONS

A Venkatesh[1]* and A Sirisha[2]

*Corresponding Author: A Venkatesh ✉ akisettivenkatesh@yahoo.in

Many Digital Signal Processing (DSP) applications carry out a large number of complex arithmetic operations. Multiplier and adder take important role in high performance of the system, which is responsible for the power and area reports. This paper is focus on optimizing the design of Fused Add Multiply (FAM) operator. This implements a technique by direct recoding of sum two numbers in Modified Booth (MB) form. An efficient multiplier with reduce partial product by N/2 where N is the number of multiplicand is introduced. The proposed FAM unit is coded in VERILOG HDL and the simulation, synthesis processes is carried out using Xilinx ISE tool. The performance of FAM unit is compared with other existing technique in terms of power consumption and area. The proposed FAM unit yields considerable reduction in terms of power consumption and area utilization.

Keywords: Add multiply operator, Modified booth, Partial product, Xilinx ISE

INTRODUCTION

Fast multipliers are essential parts of digital signal processing systems. The speed of multiply operation is of great importance in digital signal processing as well as in the general purpose processors today, especially since the media processing took off. In the past multiplication was generally implemented via a sequence of addition, Subtraction, and shift operations. Multiplication can be considered as a series of repeated additions. The number to be added is the multiplicand, the number of times that it is added

is the multiplier, and the result is the product. Each step of addition generates a partial product. In most computers, the operand usually contains the same number of bits. When the operands are interpreted as integers, the product is generally twice the length of operands in order to preserve the information content.

Recent research activities in the field of arithmetic optimization, have shown that the design of arithmetic components combining operations which share data, can lead to significant performance improvements. Based

---

[1] M.Tech Student, Department of E.C.E., Chirala Engineering College, Ramapuram Beach Rd, Chirala, Andhra Pradesh 523157, India.

[2] Assistant Professor, Department of E.C.E., Chirala Engineering College, Ramapuram Beach Rd, Chirala, Andhra Pradesh 523157, India.

on the observation that an addition can often be subsequent to a multiplication (e.g., insymmetric FIR filters), the Multiply-Accumulator (MAC) and Multiply -Add (MAD) units were introduced leading to more efficient implementations of DSP algorithms compared to the conventional ones, which use only primitive resources . Several architectures have been proposed to optimize the performance of the MAC operation in terms of area occupation, critical path delay or power consumption, MAC components increase the flexibility of DSP data path synthesis as a large set of arithmetic operations can be efficiently mapped onto them. Except the MAC/MAD operations, many DSP applications are based on Add-Multiply (AM) operations (e.g., FFT algorithm). 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 de-sign 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. Thus, the carry-propagate (or carry-look-ahead) adder of the conventional AM design is eliminated resulting in considerable gains of performance. Lyu and Matulapresented a signed -bit MB recorder which trans-forms redundant binary inputs to their MB recoding form.

In this paper, we focus on AM units which implement the op-eration $Z = X(A+B)$. The conventional design of the AM operator (Figure 1a) requires that its inputs *A* and *B* are first driven to an adder and then the input and the sum are driven to a multiplier in order to get *Z*. The

drawback of using an adder is that it inserts a significant delay in the critical path of the AM. As there are carry signals to be propagated inside the adder, the critical path depends on the bit-width of the inputs.

An optimized design of the AM operator is based on the fusion of the adder and the MB encoding unit into a single data path block (Figure 1b) by direct recoding of the sum $Y = A+B$ to its MB representation. The Fused Add-Multiply (FAM) component contains only one adder at the end (final adder of the parallel multiplier). As a result, significant area savings are observed and the critical path delay of the recoding process is reduced and decoupled from the bit- width of its inputs. In this work, we present a new technique for direct recoding of two numbers in the MB representation of their sum.

In the majority of Digital Signal Processing (DSP) applications the critical operations usually involve many multiplications and/or accumulations. For real-time signal processing, a high speed and high throughput Multiplier-Adder is always a key to achieve a high performance digital signal processing system and versatile Multimedia functional units.

In the last few years, the main consideration of MAD design is to enhance its speed. This is because; speed and throughput rate is always the concern of block. But for the epoch of personal communication, low power design also becomes another main design consideration. This is because; battery energy available for these portable products limits the power consumption of the system. Therefore, the main motivation of this work is to investigate various Pipelined multiplier/accumulator architectures and circuit design techniques which are suitable for implementing high throughput signal

Figure 1: Conventional Multiplier

$$Z = X \cdot Y = X \cdot (A + B)$$

(a)
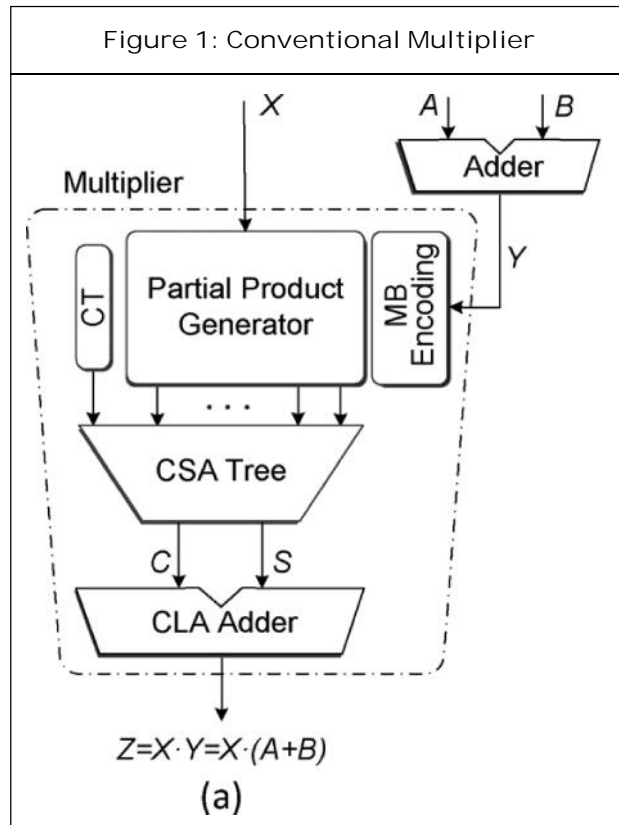
processing algorithms and at the same time achieve low power consumption. A conventional VMFU unit consists of (fast multiplier) multiplier and an accumulator that contains the sum of the previous consecutive products. The function of the VMFU unit is given by the following equation:

F = Σ Ai Bi

F = F*X

The main goal of a block design is to enhance the speed of the MAD unit, and at the same time limit the power consumption. In a pipelined MAD circuit, the delay of pipeline stage is the delay of a 1-bit full adder. Estimating this delay will assist in identifying the overall delay of the pipelined MAD. In this work, 1-bit full adder is designed. Area, power and delay are calculated for the full adder, based on which the pipelined MAD unit is designed for low power.

CIRCUIT DESIGN FLOW

One of the most advanced types of MAC for general-purpose digital signal processing has been proposed by Elguibaly. It is an architecture in which accumulation has been combined with the Carry Save Adder (CSA) tree that compresses partial products. In the architecture proposed in, the critical path was reduced by eliminating the adder for accumulation and decreasing the number of input bits in the final adder.

While it has a better performance because of the reduced critical path compared to the previous VMFU architectures, there is a need to improve the output rate due to the use of the final adder results for accumulation. The architecture to merge the adder block to the accumulator register in the VMFU operator was proposed to provide the possibility of using two separate N/2-bit adders instead of one-bit adder to accumulate the MAC results. Recently, Zicari proposed an architecture that took a merging technique to fully utilize the 4-2 compressor .It also took this compressor as the basic building blocks for the multiplication circuit.


Figure 2: Circuit Design Flow
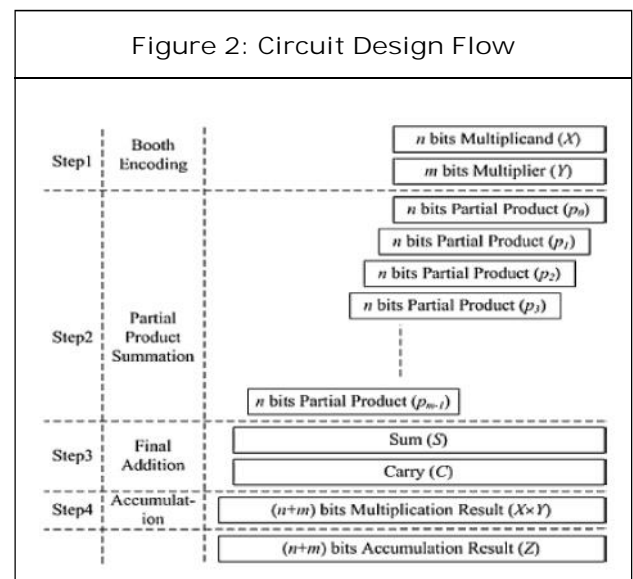
A new architecture for a high-speed MAC is proposed. In this MAC, the computations of multiplication and accumulation are combined and a hybrid-type CSA structure is proposed to reduce the critical path and improve the output rate. It uses MBA algorithm based on 1's complement number system. A modified array structure for the sign bits is used to increase the density of the operands. A Carry Look-Ahead Adder (CLA) is inserted in the CSA tree to reduce the number of bits in the final adder. In addition, in order to increase the output rate by optimizing the pipeline efficiency, intermediate calculation results are accumulated in the form of sum and carry instead of the final adder outputs.

A multiplier can be divided into three operational steps. The first is radix-2 Booth encoding in which a partial product is generated from the multiplicand X and the multiplier Y. The second is adder array or partial product compression to add all partial products and convert them into the form of sum and carry. The last is the final addition in which the final multiplication result is produced by adding the sum and the carry. If the process to accumulate the multiplied results is included, a MAC consists of four steps, as shown in Figure 3 which shows the operational steps explicitly.


Figure 3: Block Diagram of Mac
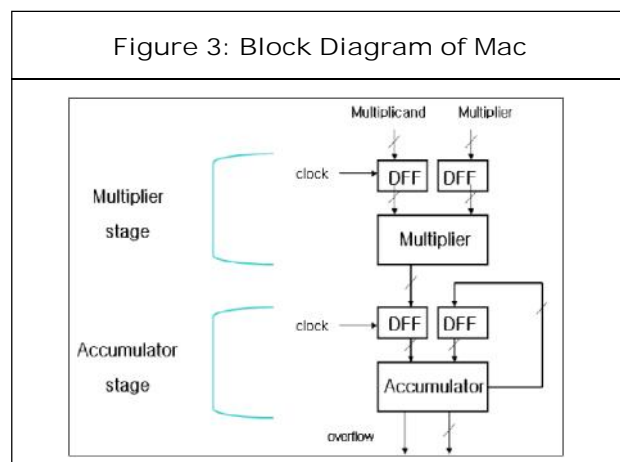
Modified Booth Encoder

In order to achieve high-speed multiplication, multiplication algorithms using parallel counters, such as the modified Booth algorithm has been proposed, and some multipliers based on the algorithms have been implemented for practical use. This type of multiplier operates much faster than an array multiplier for longer operands because its computation time is proportional to the logarithm of the word length of operands.

Figure 4: Modified Booth Encoder
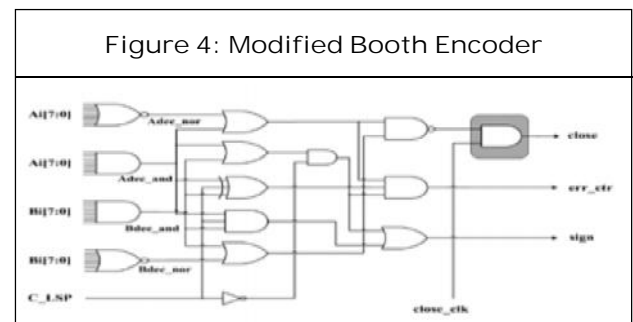
Booth multiplication is a technique that allows for smaller, faster multiplication circuits, by recoding the numbers that are multiplied. It is possible to reduce the number of partial products by half, by using the technique of radix-4 Booth recoding. The basic idea is that, instead of shifting and adding for every column of the multiplier term and multiplying by 1 or 0, only takes every second column, and multiply by ±1, ±2, or 0, to obtain the same results.

The advantage of this method is the having of the number of partial products. To Booth recode the multiplier term and consider the bits in blocks of three, such that each block overlaps the previous block by one bit. Grouping starts from the LSB, and the first block only uses two bits of

Figure 5: Grouping of Bits from the Multiplier Term
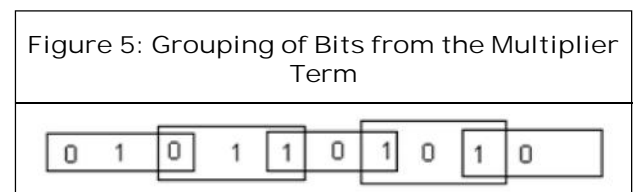
the multiplier. Shows the grouping of bits from the multiplier term for use in modified booth encoding.

Each block is decoded to generate the correct partial product. The encoding of the multiplier Y, using the modified booth algorithm, generates the following five signed digits, –2, –1, 0, +1, +2. Each encoded digit in the multiplier performs a certain operation on the multiplicand, X, as illustrated in Table 1.

Table 1: Modified Booth Encoder		
Block	Re - coded digit	Operation on X
000	0	0 X
001	+1	+1 X
010	+1	+1 X
011	+2	+2 X
100	-2	-2 X
101	-1	-1 X
110	-1	-1 X
111	0	0 X

For the partial product generation and adopt Radix-4 Modified Booth algorithm to reduce the number of partial products for roughly one half. For multiplication of 2's complement numbers, the two-bit encoding using this algorithm scans a triplet of bits. When the multiplier B is divided into groups of two bits, the algorithm is applied to this group of divided bits.

The PP generator generates five candidates of the partial products, i.e., {-2A,-A, 0, A, 2A}. These are then selected according to the Booth
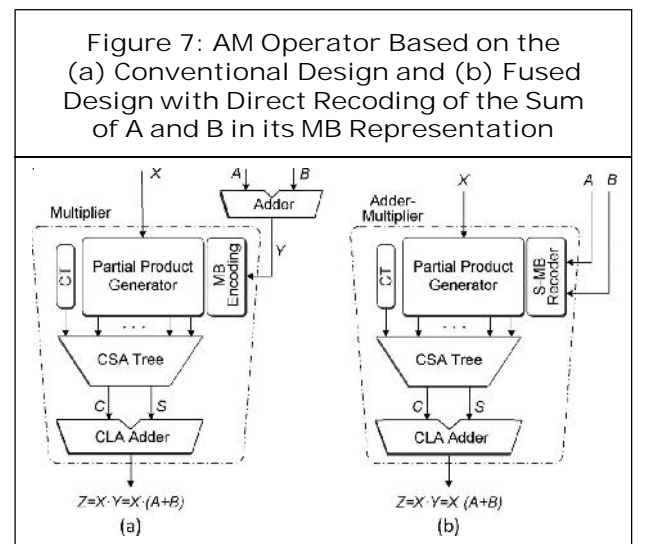

Figure 6: Illustration of Multiplication Using Modified Booth Encoding
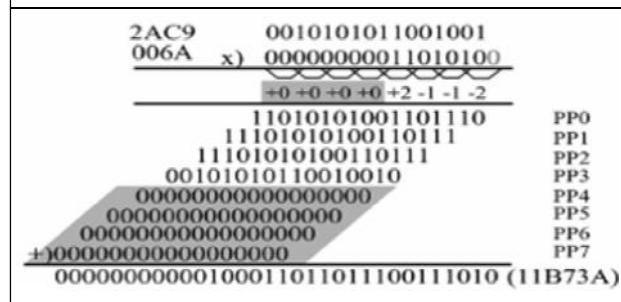
encoding results of the operand B. When the operand besides the Booth encoded one has a small absolute value, there are opportunities to reduce the spurious power dissipated in the compression tree.

Modified Booth (MB) is a prevalent form used in multiplication. It is a redundant signed-digit radix-4 en-coding technique. Its main advantage is that it reduces by half the number of partial products in multiplication comparing to any other radix-2 representation.

The multiplier is a basic parallel multiplier based on the MB algorithm. The terms CT, CSA Tree and CLA Adder are referred to the Correction Term, the Carry-Save Adder Tree and the final Carry-Look-Ahead Adder of the multiplier.

Figure 7: AM Operator Based on the (a) Conventional Design and (b) Fused Design with Direct Recoding of the Sum of A and B in its MB Representation

PARTIAL PRODUCT GENERATOR

The multiplication first step generates from A and X a set of bits whose weights sum is the product P. For unsigned multiplication, P most significant bit weight is positive, while in 2's complement it is negative.

The partial product is generated by doing AND between 'a' and 'b' which are a 4 bit vectors and

Figure 8: Booth Partial Product Selector Logic
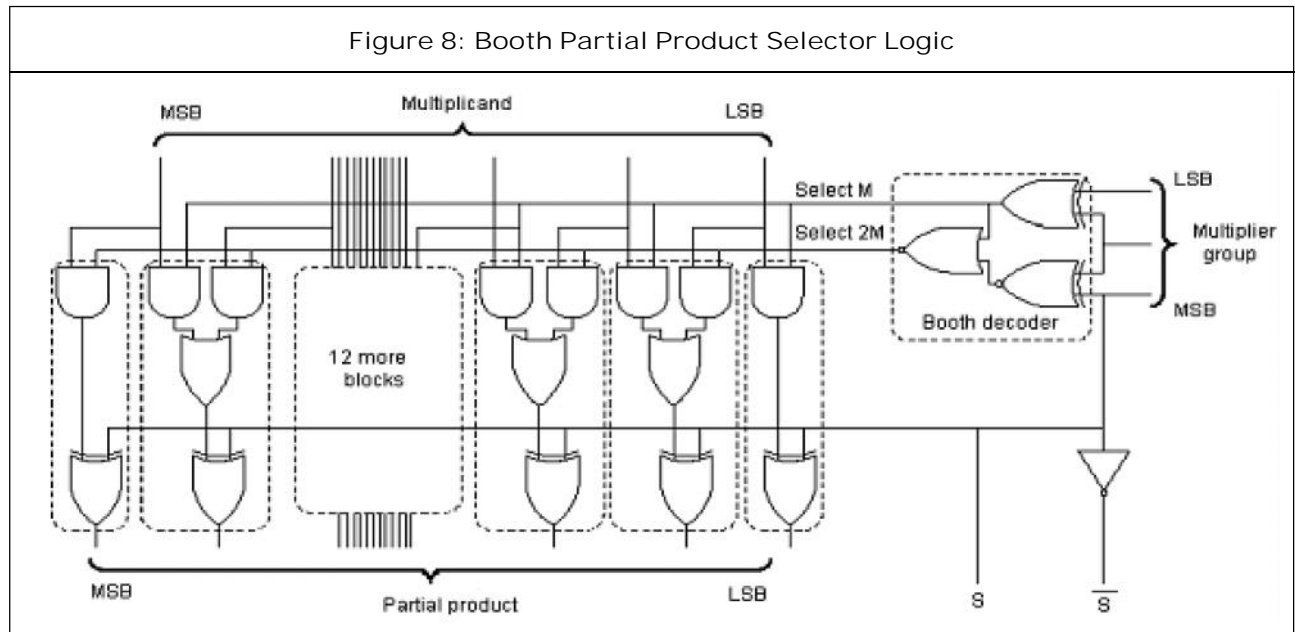
take four bit multiplier and 4-bit multiplicand get sixteen partial products in which the first partial product is stored in 'q'. Similarly, the second, third and fourth partial products are stored in 4-bit vector n, x, y.

Multiplication consists of three steps: 1) the first step to generate the partial products; 2) the second step to add the generated partial products until the last two rows are remained; 3) the third step to compute the final multiplication results by adding the last two rows.

The modified Booth algorithm reduces the number of partial products by half in the first step and used the Modified Booth Encoding (MBE) scheme proposed in. It is known as the most efficient Booth encoding and decoding scheme. To multiply X by Y using the modified Booth algorithm starts from grouping Y by three bits and encoding into one of {−2, −1, 0, 1, 2}. Table shows the rules to generate the encoded signals by MBE scheme.

Figure 9: Booth Partial Products Generation
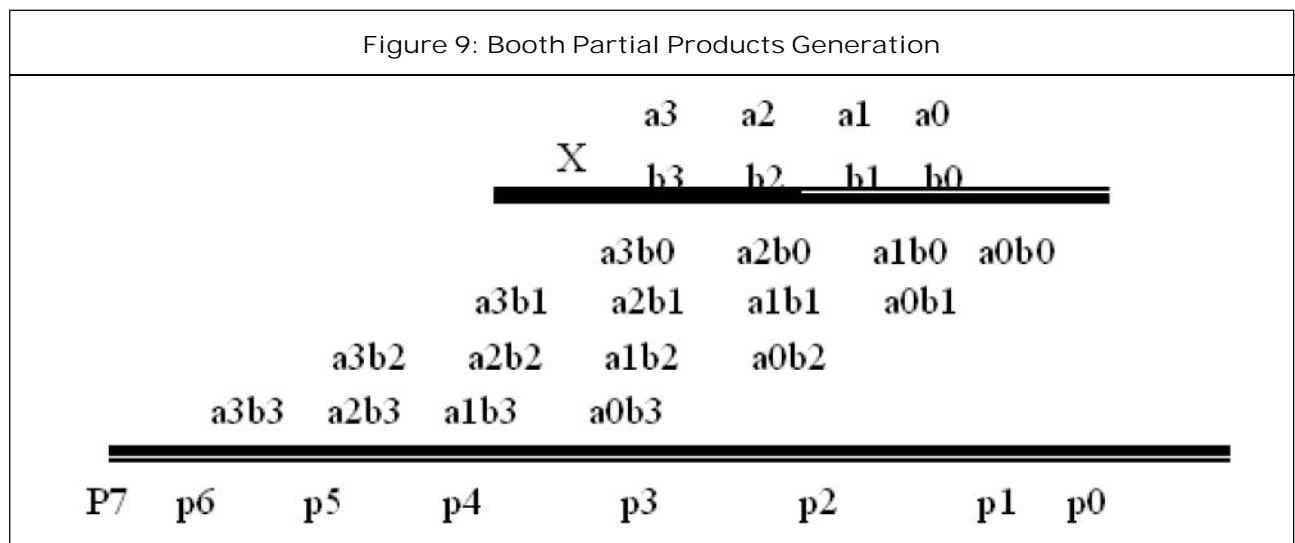
RESULTS

In the waveform, *X*, *A*, *B* represents the inputs which we are applying to the design. Similarly *Z* is the output signal for the design. To obtain the required outputs force the inputs logic with the required values. Here the addition operation is performed between the inputs *A* and *B* and the obtained result is multiplied with the input X which is nothing but add multiply operation. The intermediate signals which are called as wires those information also available in the waveform.

RTL Schematic

The RTL Schematic gives the information about the user view of the design. The internal blocks
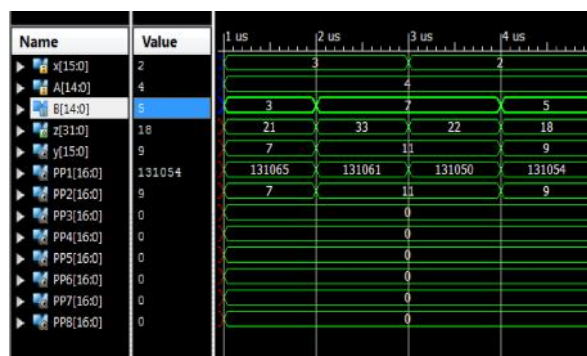
Figure 10: Waveform for Add Multiply Operator
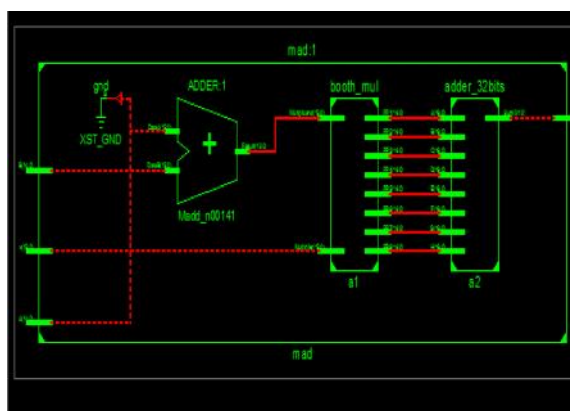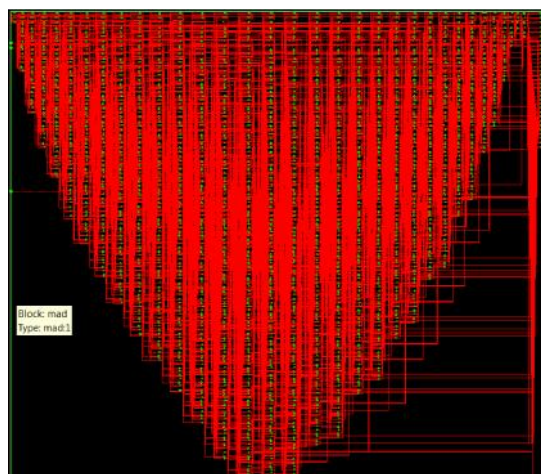

Figure 11: RTL Schematic

Figure 12: Technology Schematic

contains the basic gate representation of the logic. These basic gate realization is purely depend upon the corresponding FPGA selection and the internal database information.

Technology Schematic

The Technology Schematic gives the information about the chip view of the design. This mainly consists of LUTs, input buffers , output buffers, D-Flipflop components. Internally Look Up Tables (LUTs) contains the corresponding logic boolean equations, its schematic representation, k-map representation and its truth table representation.

CONCLUSION

This paper focuses on optimizing the design of the Fused-Add Multiply (FAM) operator. This work presents a functional unit which is designed with multiplier-accumulator (MAC), addition, subtraction. The basic building blocks for the unit are identified and each of the blocks is analyzed for its performance. We propose a structured technique for the direct recoding of the sum of two numbers to its MB form. The proposed recoding scheme consume 0.076 mw power and 956 LUTs only by using XC3S500e-5FG320

device, when they are incorporated in FAM designs, yield considerable performance improvements in comparison with the most efficient recoding schemes found in literature. The presented technique explores its applications in multimedia/DSP computations, where the theoretical analysis and the realization issues are fully discussed. In this project Xilinx-ISE tool is used for logical verification, synthesizing performing placing and routing operation for system verification.

REFERENCES

1. Aswathy Sudhakar and Gokila D (2010), "Run-Time Configurable Pipelined Modified Baugh-Wooley Multipliers", *Advances in Computational Sciences and Technology*, Vol. 3, No. 2, pp. 223-235, ISSN: 0973-6107.

2. Amaricai A, Vladutiu M and Boncalo O (2010), "Design Issues and Implementations for Floating-Point Divide-Add Fused", *IEEE Trans. Circuits Syst. II–Exp. Briefs*, Vol. 57, No. 4, pp. 295-299.

3. Cavanagh J J F (1984), *Digital Computer Arithmetic*, McGraw-Hill, New York.

4. Haung Z and Ercegovac M D (2005), "High Performance Low Power Left to Right Array Multiplier Design", *IEEE Rans. Computer*, Vol. 54, No. 3, pp. 272-283.

5. Kwon O, Nowka K and Swartzlander E E (2002), "A 16-bit by 16-bit MAC Design Using Fast 5: 3 Compressor Cells", *J. VLSI Signal Process. Syst.*, Vol. 31, No. 2, pp. 77-89.

6. Magnus Sjalander and Per Larson-Edefors (2008), "The Case for HPM-Based Baugh-Wooley Multipliers", March, Chalmers University of Technology, Sweden.

7. Myoung-Cheol Shin, Se-Hyeon Kang and In-Cheol Park (2010), "An Area-Efficient Iterative Modified-Booth Multiplier Based on Self-Timed Clocking", Industry, and Energy Through the Project System IC 2010, and by IC Design Education Center (IDEC).

8. Nikolaidis S, Karaolis E and Kyriakis-Bitzaros E D (2000), "Estimation of Signal Transition Activity in FIR Filters Implemented by a MAC Architecture", *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, Vol. 19, No. 1, pp. 164-169.

9. Soojin Kim and Kyeongsoon Cho (2010), "Design of High-Speed Modified Booth Multipliers Operating at GHz Ranges", *World Academy of Science, Engineering and Technology*, Vol. 61.

10. Swartzlander E E and Saleh H H M (2012), "FFT Implementation with Fused Floating-Point Operations", *IEEE Trans. Comput.*, Vol. 61, No. 2, pp. 284-288.

International Journal of Engineering Research and Science & Technology

Hyderabad, INDIA. Ph: +91-09441351700, 09059645577

E-mail: editorijlerst@gmail.com or editor@ijerst.com

Website: www.ijerst.com