FPGA Realization of High Speed Hybrid Multiplier

Mr. M.Prasad[1], G Surendra Reddy[2], Y Gowtham Reddy[3], V Sumanth reddy[4], P Sathvika[5]
[1]Asst. Professor, Dept of ECE
[2, 3, 4, 5]Dept of ECE
[1, 2] Anna University Regional Campus Coimbatore

Abstract- *An innovative hybrid multiplier concept is put forth in this brief. The hybrid multiplier combines two different multiplier types. Modern computing systems need multipliers that use less power, space, and time. In this study, we use Wallace-Dadda and Vedic multipliers to expand a novel concept of high-performance hybrid multiplier. To get quicker outcomes, partial items are added by being divided into smaller groupings. By creating an 8-bit hybrid multiplier with four subgroups for the partial products, the suggested solution is demonstrated. In this analysis, two different multipliers are applied to alternative groups. Finally, the carry look ahead adder (CLA) is used to reduce carry propagation delay in the proposed hybrid multiplier. The proposed hybrid multiplier has been synthesized using the Quartus -II 9.1 ISE tool. This hybrid multiplier is faster, consumes less power, and occupies less area as compared to the conventional hybrid multipliers*

Keywords- Multiplier, FPGA, High speed, Booth, Timing delay

I. INTRODUCTION

The design of multipliers with reduced delay, smaller in area, and low power consumption is crucial since the multiplication action is more time-critical, more area-consuming, and more power-intensive. The multiplication process is divided into three steps.: 1) generating partial products; 2) reducing the partial products to two rows; 3) and finally adding the two rows the product is obtained [1].“

Many algorithms and architectures have been developed to decrease the delay, the area, and the power consumption because the speed of the multiplier depends on the speed of the partial product generator (PPG), the speed of the partial product reduction tree (PPRT), and the speed of the carry propagate adder (CPA).

Numerous high performance strategies [2] have been published because the multiplier's performance is based on the PPG's speed. Designing high speed PPRT can also significantly improve the multiplier's performance; numerous high performance topologies [3] have been described. And finally, a number of high speed Carry Look-ahead (CLA) adder approaches have been presented [4] because the multiplier's maximum speed is dependent on the performance of the Carry Propagate Adder (CPA).

ere exist multipliers that can only exThere exist multipliers that can only execute multiplication operations on unsigned values, such as the Braun array multiplier and the array multiplier. There exist multipliers that can only multiply signed numbers, such as the Modified Booth Encoder (MBE) multiplier and the Baugh-Wooley (BW) multiplier. The number of partial products is halved when using an MBE multiplier. However, extra hardware is needed for the encoder and decoder logic. Due to its complexity, the Baugh-Wooley algorithm is rarely extensively used.The need for signed and unsigned number multiplication is that the signed multiplier operates on the data range of $\{+2^{n-1}$ -1 to $-2^{n-1}\}$ but the unsigned multiplier performs multiplication on the data of magnitude $\{0$ to 2^n -1$\}$. For an example the signed number in 2’s complement number system with n = 4, the data operand range is from {+7 to -8}, but for the unsigned number the data operand magnitude is from {0 to 15} [5]. Hence, in this dissertation, the proposed array multiplier, the NMBE multiplier, pipelined multipliers have been designed to operate on both the signed and unsigned number system.

II. LITERATURE REVIEW

“A novel Modified Booth Encoding (MBE) scheme known as the Partial Product Generator (PPG) has been put out by W. C. Yeh and C. W. Jen [1]. They have suggested employing encoder and decoder logic in the MBE. Encoder and decoder logic optimises the delay path. To enhance the functionality of the parallel multiplier, they have also suggested the Multiple-Level Conditional-Sum Adder (MLCSMA) as the Carry Propagate Adder (CPA). All of the steps of the full adder's carry are provided in parallel by the MLCSMA's parallel carry generator network. In comparison to previous parallel multipliers, the suggested MBE algorithm and MLCSMA method can optimise the delay and ultimately lower it by 8%. This suggested multiplier can multiply operands with signed numbers. “

“A novel Modified Booth Encoding (MBE) scheme known as the Partial Product Generator (PPG) has been put out by W. C. Yeh and C. W. Jen [1]. They have suggested

employing encoder and decoder logic in the MBE. Encoder and decoder logic optimises the delay path. To enhance the functionality of the parallel multiplier, they have also suggested the Multiple-Level Conditional-Sum Adder (MLCSMA) as the Carry Propagate Adder (CPA). All of the steps of the full adder's carry are provided in parallel by the MLCSMA's parallel carry generator network. In comparison to previous parallel multipliers, the suggested MBE algorithm and MLCSMA method can optimise the delay and ultimately lower it by 8%. This suggested multiplier can multiply operands with signed numbers. However, additional logic circuits have been utilised to transform a non-standard partial product into a standard partial product. Due to the fact that this employs ordinary MBE as the PPG, it has been claimed to function better than standard MBE multipliers.

A well-organized MBE multiplier architecture has been put up by Wang, Shyh-Jye Jou, and Chung-Len Lee [3]. In order to eliminate an unnecessary partial product row like paper, an improved booth encoder logic and booth selector logic have been developed in this paper [2]. The design of the spare-tree approach was also suggested in this paper for the complementation operation between two. As a result, the area was reduced and the speed of the signed multiplier was increased as a result of the removal of an extra partial product row and the design of the sparse-tree approach. “

III. METHODOLOGY

This hybrid multiplier's primary goals are to speed up the computation and decrease partial products. MBM immediately reduces the area and speeds up the computation by generating n/2 partial products for "n" inputs. Since 3:2 compressors are utilised, the Wallace tree is made to hold save adders (CSA), which are used for quick addition. It is in charge of quick addition.

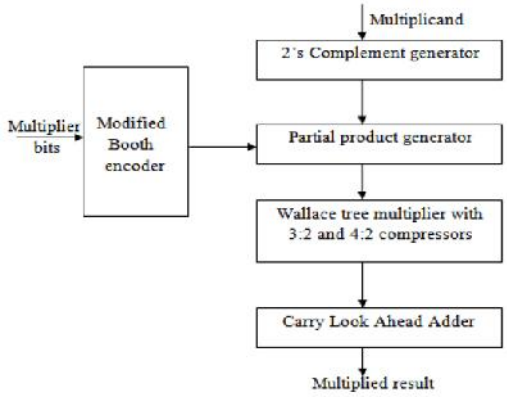

Figure 1: Architecture of the proposed method

By using the Carry Look-Ahead adder, which is mostly suggested for final accumulation, the results of CSA are eventually computed. Multiple logic circuits are used to build the adder. The term "hybrid-adder" refers to this type of adder. The architecture of the proposed method is shown in figure 1.

IV. RESULTS & DISCUSSIONS

The number of look-up tables is used to calculate the design's area. Nanoseconds are used to measure the latency. The area taken up by the design increases with the number of look-up tables. The compilation report includes information on the number of look tables, registers, multipliers (measured in terms of DSP blocks), and memory used by the design.
The combinational path delay of the design is disclosed in the timing analyzer report. The design verification is shown in the simulation report.

The results of the various models used in the current work are shown in the table below, which includes data on the compilation report, timing analyzer report, and simulation waveforms for the array multiplier, Wallace multiplier, booth multiplier, hybrid multiplier of 8 bits, as well as hybrid multipliers of 16, 32, and 64 bits.

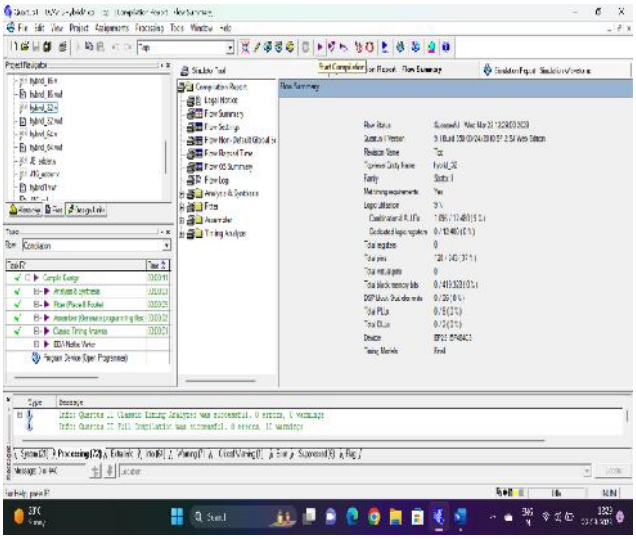

Figure2: Compilation Report of the 32 bit Hybrid Multiuplier

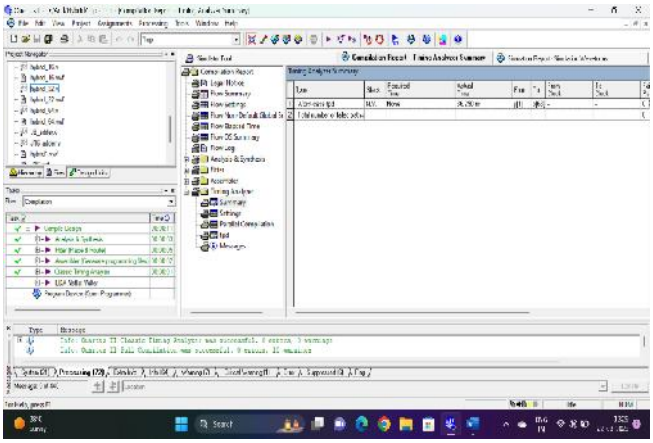

Figure3: Timing Analyzer report for the Hybrid 32 bit Multiplier

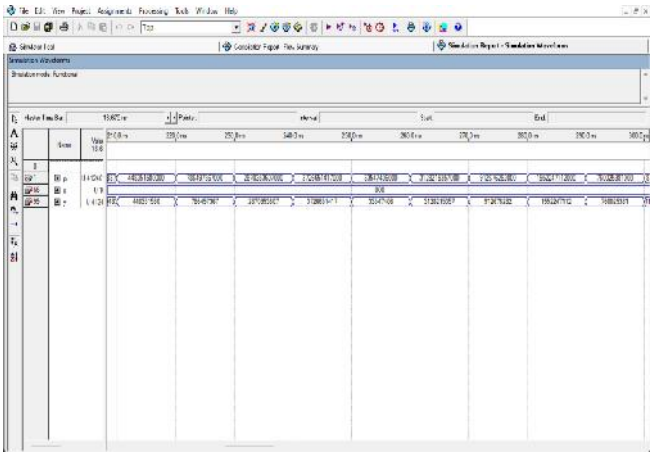

Figure4: Simulation report for the Hybrid 32 bit Multiplier

In figure 1, the compilation report is displayed. Figure 2 displays the report from the timing analyzer. In Figure 3, the simulation waveform is displayed. The array multiplier, booth multiplier, and Wallace 8-bit multiplier were contrasted with the hybrid multiplier. Among all the models now in use, the hybrid multiplier shows to be the best. 16 bits, 32 bits, and 64 bits are added to the hybrid multiplier. In table 1, a summary of the various multipliers is presented.

Table 1 : Summary of different multipliers

S. No	Architecture	Area	Delay (ns)
1	Array – 7 bit	147	22.113
2	Wallace – 8 bit	124	16.917
3	Booth – 8 bit	112	16.65
4	Hybrid – 8 bit	132	13.52
5	Hybrid – 16 bit	257	31.649
6	Hybrid – 32 bit	1093	36.29
7	Hybrid – 64 bit	4501	44.918

VI. CONCLUSIONS

The combination of high speed multipliers and adders is known as a hybrid architecture. Due to its speedier processes, the design can be employed to overcome the difficulties in all signal processing applications. Since the partial products are decreased, the computation proceeds more quickly. The final product of the multiplier is calculated by simultaneously adding each two subsequent multiplicand bits of partial products using different-sized hybrid adders, which reduces the multiplier's delay and area.

REFERENCES

[1] Hou-Jen Ko and Shen-Fu Hsiao, “Design and Application of Faithfully Rounded and Truncated Multipliers With Combined Deletion, Reduction, Truncation, and Rounding”, IEEE transactions on circuits and systems—ii: express briefs, vol. 58, no. 5, may 2011.

[2] P.Anitha and Dr.P. Ramanathan, “A new hybrid Multiplier using Dadda and Wallace method”, International Conference on Computer Communication and informatics, jan 2014

[3] k. Ram Prakash and A.V. Sanju , “Design and Implementation of Signed, Rounded and Truncated Multipliers using Modified Booth Algorithm for Dsp Systems”,International Journal of Emerging Technology and advanced engineering,vol.4,Feb 2014.

[4] Naveen Kr.Gahlan, Prabhat Shukla and Jasbir Kaur, “Implementation Of Wallace Tree Multiplier Using Compressor”, international Journal Computer Technology& Applications (lJCTA), Vol.3(3), 1194-1l99,May-June 2012.

[5] M. J. Schulte and E. E. Swartzlander, Jr., “Truncated multiplication with correction constant,” in VLSI Signal Processing VI. Piscataway, NJ: IEEE Press, 1993.

[6] R. Devarani and Mr.C.S. ManikandaBabu, “Design and Implementation of Truncated Multipliers for Precision Improvement and Its Application to a Filter Structure”,International Journal of ModernEngineering Research (IJMER), Nov-Dec 2012.

[7] Dadda L. (1965), “Some Schemes for Parallel Multipliers”, Alta Frequenza, Vol. 34, 1965.