LUT-Based Area-Optimized Accurate Multiplier Design for Signal Processing Applications

B. V. V. Satyanarayana, B. Kanaka Sri Lakshmi, G. Prasanna Kumar(✉), and K. Srinivas

Department of ECE, Vishnu Institute of Technology, Bhimavaram 534202, India
godiprasanna@gmail.com

Abstract. Multipliers play a role in various aspects of smart cities, which can be used in many applications like Traffic management, energy management and environmental management etc. The wide variety of applications of multipliers are in the field of signal processing and image processing. FPGA design of multiplier is one of the complex tasks in Digital electronics. Most of the designs uses DSP blocks, these multipliers are complex and occupies much area in FPGA. Accurate multiplier design with low area on FPGA is the challenging task. The proposed method is accurate multiplier design, which is designed only using lookup table (LUT). The proposed design has low power and reduced area because of using simple LUT's for generating partial product. The proposed accurate multipliers reduce 10% less Hardware on vertex 7 FPGA compared to existing designs.

Keywords: Multiplier · FPGA · LUT · Power · vertex 7

1 Introduction

In arithmetic operators' multiplier is one basic operator, which is very important and crucial. In the digital era multiplier design has challenges in accuracy and complexity. FPGA implementation of multipliers has two major challenges, one is hardware complexity and another is accuracy. In some applications in the area of image processing and signal processing has that much significance of accuracy. The approximate multipliers are best suited for such applications. This work majorly focuses on the accuracy, no approximation techniques are used.

Many applications in the field of signal processing [1] and image processing requires high speed and efficient sub components. Those are mainly adders and multipliers. In [2] Fast Fourier transform requires high speed and accurate twiddle factor multiplier for generation of higher order FFT. Many applications that's example presented in [3] gives the counter on FPGA, which gives the detecting and counting in industries.

In [4] authors proposed a fast multiplier, in which parallel counter algorithm is introduced to compress the columns. The design is efficient with respect to LUT utilization and DSP blocks, but utilization of high-power resources is the one issue found in this work. In [5] efficient soft-core multiplier on FPGA vertex-6, in which compressed tree

P. Pareek et al. (Eds.): IC4S 2023, LNICST 536, pp. 309–317, 2024.
https://doi.org/10.1007/978-3-031-48888-7_26

algorithm is used instead of partial products. This structure has complex and more routing delay compared to existing structures. The main advantage of this design is fast carry chain and simultaneous multiplication and accumulation. In [6] authors present 16-bi multiplier on FPGA Spartan 3 which is consumes more LUTs and dsp blocks. In [7] large multipliers was designed on FPGA which uses less resources, the main components in this design are DSP blocks and small multipliers. The integer linear programming problem was solved using this method. The main findings in this method are usage more DSP blocks.

In multiplier design Power and area are major parameters, high level components like DSP blocks in FPGA uses more power and area. In [8] area and power optimized multiplier is designed proposed which is approximate multipliers based on wallac tree algorithm. In the process of reducing the area, in this work compromising the accuracy. This work suitable for applications where accuracy is not a big issue, example in image processing applications. In [9] efficient multiplier approximation is proposed in which soft logic is introduced to implement the multiplier. The high-level performing resources has limitation in size and quantity. This work majorly focused on performance analysis on embedded multiplier and soft-core multiplier and clearly address how to overcome the problem using soft core techniques. Hybrid multiplier combining dadda algorithm and booth [10], which is superior than the traditional booth multiplier. There is no significant reduction in high level hardware, but speed is the main advantage in this method.

The performance analysis of Baugh Wooley multiplier is presented in [11], in which parallel architecture is used to produce high speed output. Efficiency in the design of adders is also one significant modification in this method. The improved speed and low power are main advantage in this method. This multiplier is very popular and widely used in various fields of applications.

In [12] proposed a new multiplier design for signed and unsigned data, which is very optimizes power and delay, in [13,14] multiplier design techniques are based on LUT only. In [15] default multiplier in Xilinx vivado version.

The proposed multiplier is a LUT based multiplier which optimizes the area because of no DSP blocks use and efficient carry chain generation using LUT.

The paper organized as follows Sect. 1 gives the detailed introduction and literature review, Sect. 2 describes the proposed method, Sect. 3 gives the simulation results and comparisons of proposed method with existing designs. Section 4 concludes the project with future scope.

2 Proposed Method

The literature review of existing methods gives the following identifications, most of the accurate multipliers uses the high-level hardware components on FPGA, approximate multiplier is the solution for reducing the hardware complexity and utilizing the resources efficiently. The proposed method is accurate multiplier design using efficient resource utilization. The general structure for multiplication is shown in Fig. 1. In which M is the multiplicand and N is the multiplier.

The proposed method uses two types of LUT's for partial product generation, one is Type-1 and another is Type-2 these LUT's generates the partial products and accumulation of two stages. The Type-1 LUT is LUT6_2 in Xilinx FPGA vertex-7 series.

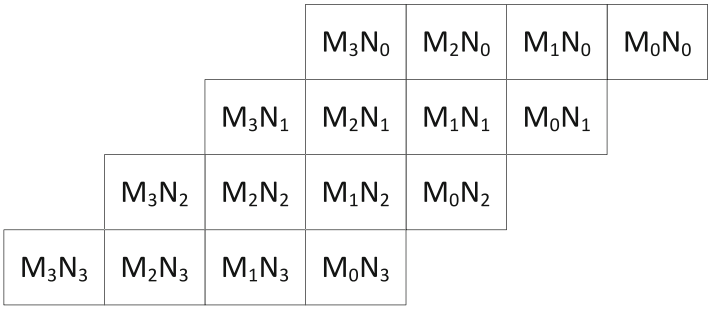

Fig. 1. Binary multiplier of size 4 bit

It is a 6 input and 2 output LUT. The Type-1 LUT is shown in Fig. 2, carry (C) and sum (S) are the output ports of this LUT, which are O5 and O6 ports in LUT6_2. The detailed configuration is shown in Table 1, the logical and gates performs the partial products and the resultant outputs C and S outputs are generated by O5 and O6 ports of LUT. The INIT attribute defines the functioning of LUT, it is 64-bit hexadecimal value in which least 32 bits defines the carry O5 and the remaining sum O6. The Verilog INIT value of Type-1 LUT 64'h7888788880008000 will produce our desired output of O6 = 7888 and O5 = 8000. The functionality of Type-2 LUT is shown in Fig. 3 and the configuration is shown Table 2. The accurate multiplier was designed using Type-1 and Type-2 multiplier.

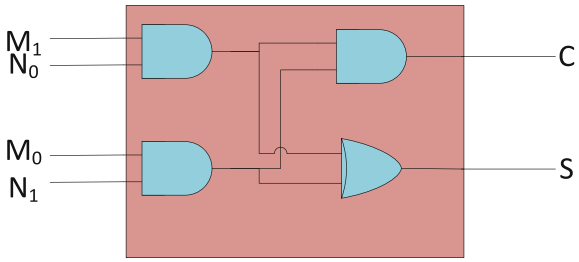

Fig. 2. Type-1 LUT structure

The main blocks in accurate multiplier design is partial product generation (PPG). The PPG for 4x4 accurate multiplier is shown in Fig. 4, generally for a 4x4 multiplier there are 4 stages of products are generated as shown in Fig. 1. Each Type-1 LUT generates two products and accumulation of stage-2 output. So that the proposed design requires two layers of four LUT's to generate all the partial products. Initially accumulation not required so Type -2 LUT used. The output of each stage is given to carry chain module to propagate the carry. The accumulated result of partial products of first two stages is p00 to p04 and for last two stages is p01 to p14. The final result is obtained from PPG unit using full adders and half adders.

The final product generation of the proposed multiplier is done by using LUT based slice adder which was shown in Fig. 5. The final output bits of multiplier is

Table 1. Configuration of Type-1 LUT

M1	N0	M0	N1	M1 N0	M0 N1	S(O6)	C(O5)
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	1	0	1	1	0
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	0	0	0	0	0
0	1	1	1	0	1	1	0
1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0
1	0	1	1	0	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	0
1	1	1	0	1	0	1	0
1	1	1	1	1	1	0	1

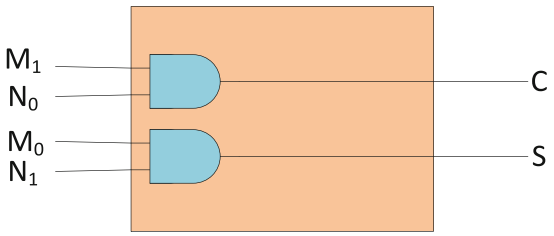

Fig. 3. Type-2 LUT structure

Table 2. Configuration of Type-2 LUT

M_1	N_0	M_0	N_1	M_1 N_0	M_0 N_1	S	C
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0

(*continued*)

Table 2. (*continued*)

M_1	N_0	M_0	N_1	M_1 N_0	M_0 N_1	S	C
0	0	1	1	0	1	0	1
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	0	0	0	0	0
0	1	1	1	0	1	0	1
1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0
1	0	1	1	0	1	0	1
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	0
1	1	1	0	1	0	1	0
1	1	1	1	1	1	1	1

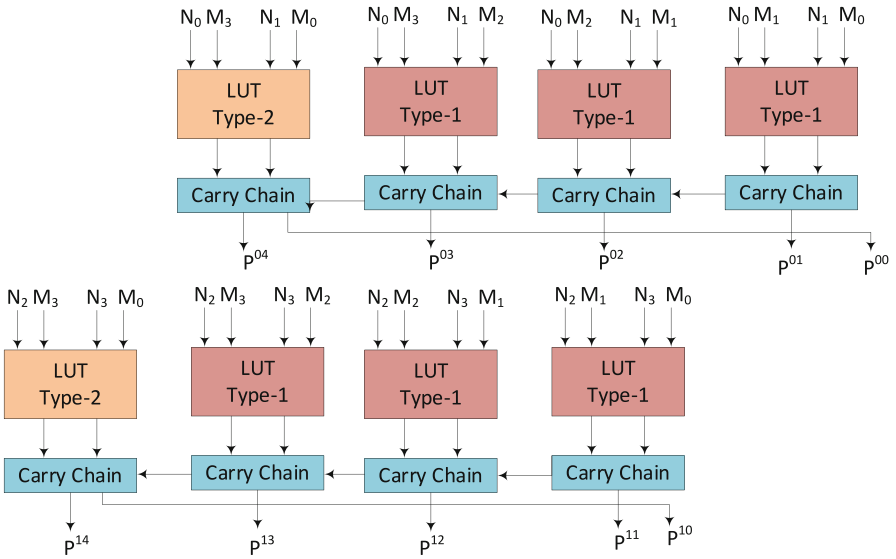

Fig. 4. Proposed Architecture 4x4 accurate multiplier

P^6,P^5,P^4,P^3,P^2,P^1 and P^0 in which P^6,P^5,P^1 and P^0 are directly obtained from the outputs of two stage adders but P^4,P^3 and P^2 requires additional circuit that is slice adder. The LUT takes the inputs from two stage adders and gives the final output based on the carry. The carry inclusion can be done by using exor gate and multiplexer.

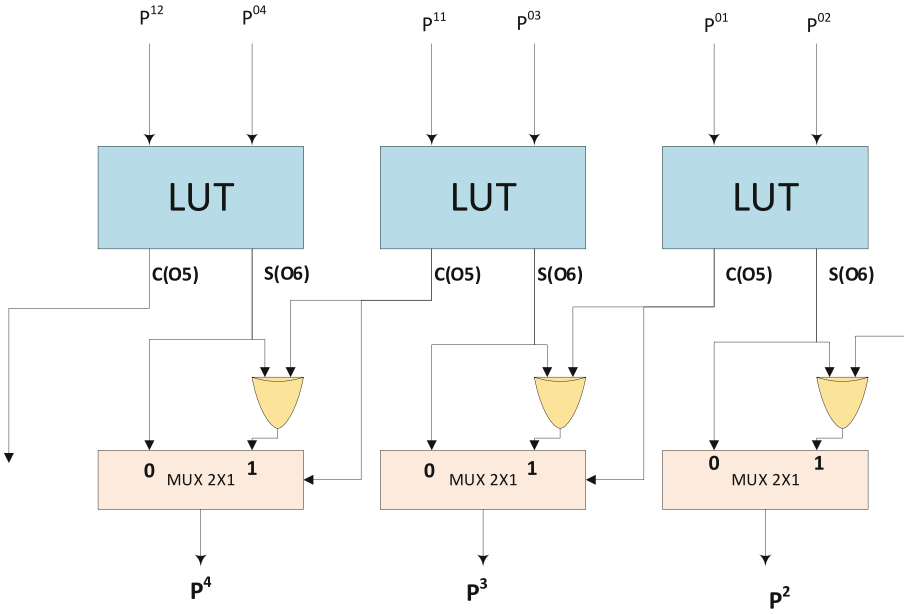

Fig. 5. Final product generation using slice adder

3 Results and Discussions

The proposed accurate multiplier is designed in Verilog HDL coding and synthesized in vertex-7 7v300T device. The proposed design was implemented in various bit lengths of 4, 6 and 8. The simulation results of 6x6 multiplier is shown in Fig. 6 and Fig. 7. The RTL schematic is shown in Fig. 8 and Fig. 9, where A and B are the inputs and C is the output.

Fig. 6. Simulation result of Accurate multiplier Result in binary

The RTL schematic of Type-1 and Type-2 LUT is shown in Fig. 9.

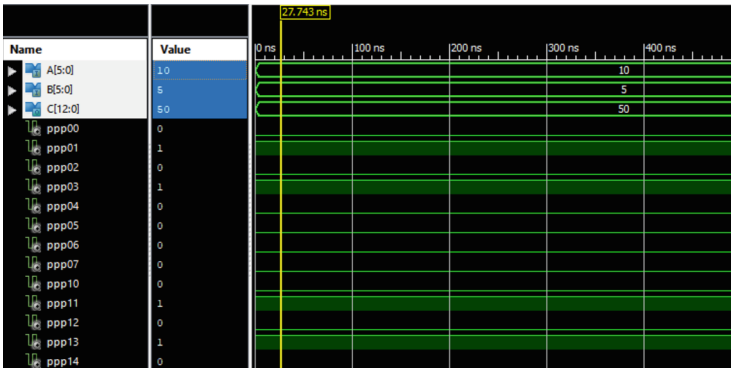

Fig. 7. Simulation result of Accurate multiplier Result in decimal

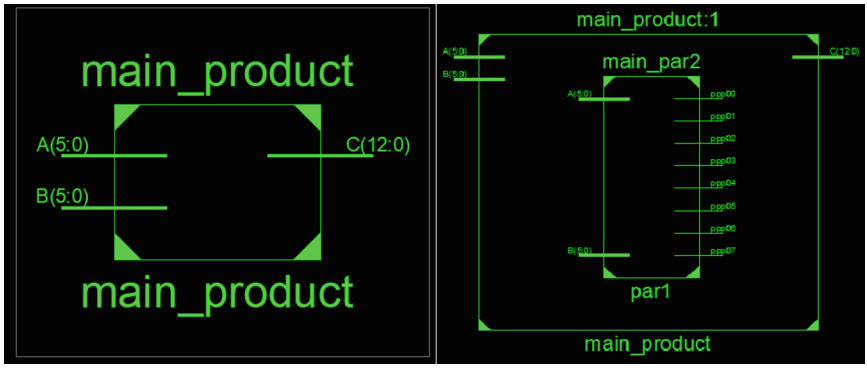

Fig. 8. RTL Schematic of main module with PPG unit

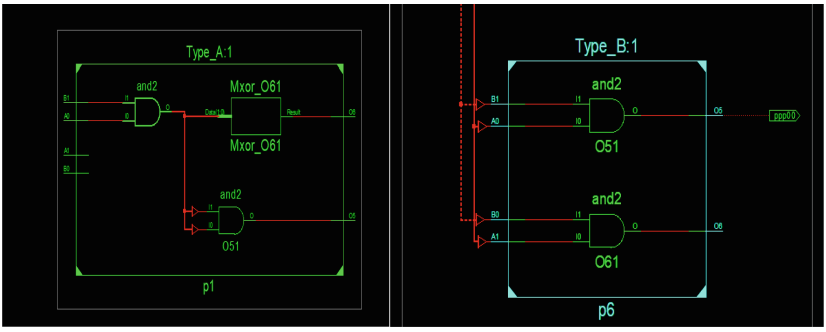

Fig. 9. RTL of Type-1 and Type-2 LUT

The performance of proposed method is evaluated by taking the parameters as number of LUT's, Delay and Power. The detailed comparison of 4x4 and 8x8 multiplier is presented in Table 3 and Table 4 respectively. The proposed method has significant

reduction in LUT compared to [13] and [14]. When power is concerned which is lower than the [15] that is predefined multiplier in xilinx.

The following are the significant advantages of proposed method.

- Number of DSP blocks are zero
- Number of LUTs are reduced than predefined multiplier
- Low power than predefined multiplier
- Delay is high

Table 3. Performance analysis of 4x4 accurate multiplier

Design	LUT	Delay (ns)	Power(pJ)
[13]	18	1.65	1.13
[14]	14	2.59	1.31
[15]	18	2.91	2.25
Proposed	14	2.71	1.14

Table 4. Performance analysis of 8x8 accurate multiplier

Design	LUT	Delay (ns)	Power(pJ)
[13]	66	2.8	6.06
[14]	54	4.5	7.26
[15]	88	3.48	9.07
Proposed	48	4.7	6.82

4 Conclusion

This paper mainly focuses on the accurate multiplier design with low area. The following conclusions were made after detailed analyses of proposed accurate multiplier. The proposed accurate multiplier was designed with only LUT's without using DSP blocks on FPGA. The area and power of the proposed design is reduced compared to existing designs. This accurate multiplier best fit for applications in signal and image processing. In future this work can be extended to application level in image processing for convolution operation.

References

1. Bhagyasri, G., Prasannakumar, G., Murthy, P.S.N.: Underwater image enhancement using SWT based image fusion and colour correction. In: 2019 International Conference on Intelligent Computing and Control Systems (ICCS), IEEE, pp. 749–754 (2019)

2. Godi, P.K., Krishna, B.T., Kotipalli, P.: Design optimisation of multiplier-free parallel pipelined FFT on field programmable gate array. IET Circ. Devices Syst. **14**, 995–1000 (2020)
3. Arvapally, S., Khan, M.A., Vemula, P.C., Sonnaila, P. Chary, U.G.: FPGA implementation of industry automated bottle counter. In: 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), vol. 2, IEEE, pp. 461–465 (2017)
4. Kakacak, A., Guzel, A.E., Cihangir, O., Gören, S., Ugurdag, H.F.: Fast multiplier generator for FPGAs with LUT based partial product generation and column/row compression. Integration **57**, 147–157 (2017)
5. Kumm, M., Abbas, S., Zipf, P.: An efficient softcore multiplier architecture for Xilinx FPGAs. In: 2015 IEEE 22nd Symposium on Computer Arithmetic, IEEE, pp. 18–25 (2015)
6. Ram, G.C., Subbarao, M.V., Kumar, D.G., Terlapu, S.K.: FPGA implementation of 16-Bit wallace multiplier using HCA. In: Advances in Micro-Electronics, Embedded Systems and IoT: Proceedings of Sixth International Conference on Microelectronics, Electromagnetics and Telecommunications (ICMEET 2021), Vol. 1, pp. 419–427 (2022)
7. Kumm, M., Kappauf, J., Istoan, M., Zipf, P.: Resource optimal design of large multipliers for FPGAs. In: 2017 IEEE 24th Symposium on Computer Arithmetic (ARITH), IEEE, pp. 131–138 (2017)
8. Bhardwaj, K., Mane, P.S., Henkel, J.: Power-and area-efficient approximate wallace tree multiplier for error-resilient systems. In: Fifteenth international symposium on quality electronic design, IEEE, pp. 263–269 (2014)
9. Parandeh-Afshar, H., Ienne, P.: Measuring and reducing the performance gap between embedded and soft multipliers on FPGAs. In: 2011 21st International Conference on Field Programmable Logic and Applications, IEEE, pp. 225–231 (2011)
10. Millar, B., Madrid, P.E., Swartzlander, E.E.: A fast hybrid multiplier combining Booth and Wallace/Dadda algorithms. In: [1992] Proceedings of the 35th Midwest Symposium on Circuits and Systems, IEEE, pp.158–165 (1992)
11. Biradar, V.B., Vishwas, P.G., Chetan, C.S., Premananda, B.S.: Design and performance analysis of modified unsigned braun and signed Baugh-Wooley multiplier. In: 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), IEEE, pp. 1–6 (2017)
12. Ullah, S., Schmidl, H., Sahoo, S.S., Rehman, S., Kumar, A.: Area-optimized accurate and approximate softcore signed multiplier architectures. IEEE Trans. Comput. **70**, 384–392 (2020)
13. Ullah, S., Nguyen, T.D.A., Kumar, A.: Energy-efficient low-latency signed multiplier for FPGA-based hardware accelerators. IEEE Embed. Syst. Lett. **13**, 41–44 (2020)
14. Ullah, S., Rehman, S., Shafique, M., Kumar, A.: High-performance accurate and approximate multipliers for FPGA-based hardware accelerators. IEEE Trans. Comput. Des. Integr. Circ. Syst. **41**, 211–224 (2021)
15. LogiCORE, I.P.: LogiCORE IP V12. 0. San Jose, CA, USA Xilinx (2015)