Design and implementation of Karatsuba multipliers a modular adaptive architecture by using effective VLSI technology and carry look ahead adders

[1]Navya Sree Veerapaneni, [2]Manasa Thorrikonda, [3]Shalini Kusuma, [4]Alala Ramya
[1,2,3]Assistant Professor, [4]UG Student, [1,2,3,4]Department of Electronics and Communication Engineering, Rishi MS Institute of Engineering and Technlogy for Women, Kukatpally, Hyderabad.

ABSTRACT
Effective binary multiplication is required for today's computation-intensive applications, including DSP, image processing, floating point processors, and communication technologies. This operation is often the most time-, space-, and energy-intensive construction component. In this project, a technique to unsigned binary multiplication that uses less space and power is presented. Based on the Vedic Karatsuba algorithm, a 16-bit multiplier has been created. It is optimised using a carry look ahead adder structure, reversible logic, and an adaptive, recursive method. The designs were created using Xilinx and Verilog.
Keywords - Carry look ahead Adder, Karatsuba Multiplier, Recursive adaptiveKaratsuba algorithm.

INTRODUCTION

Today's digital signal processing and a number of other applications rely heavily on multipliers. With technological advancements, numerous researchers have attempted and continue to attempt to design multipliers that offer either of the following design targets: high speed, low power consumption, regularity of layout and therefore less area, or even a combination of them in one multiplier, making them suitable for various high speed, low power, and compact VLSI implementation. The "add and shift" algorithm is a widely used technique for multiplying numbers. The primary factor affecting the performance of parallel multipliers is the required number of partial products. To reduce the number of partial products to be added, Vedic multiplier is one of the mostpopular methods. To attain pace
enhancements Wallace Tree set of rules may be used to reduce the variety of sequential adding levels. Further, by using combining both Modified Booth algorithm and Wallace Tree approach we can see advantage of both algorithms in a single multiplier. On the opposite hand
‑serial-parallel‖ multipliers compromise velocity to gain higher performance for region and electricity consumption. The selection of a parallel or serial multiplier simply relies upon on the nature of application. In this lecture we introduce the multiplication algorithms and architecture and compare them in terms of speed, area, power and combination of these metrics.

OBJECTIVE

The binary multiplication also happens in same way of digit multiplication as shown in below examplehere by getting partial products and gates are used and we are using adder (half adder, full adder) adding the columns.

```
          1 0 1 0 x 1 0 1
          ---------------
            1 0 1 0
          0 0 0 0
        1 0 1 0
      ---------------------
      1 1 0 0 1 0
```

An example of 4-bit multiplicationmethod is shown below:

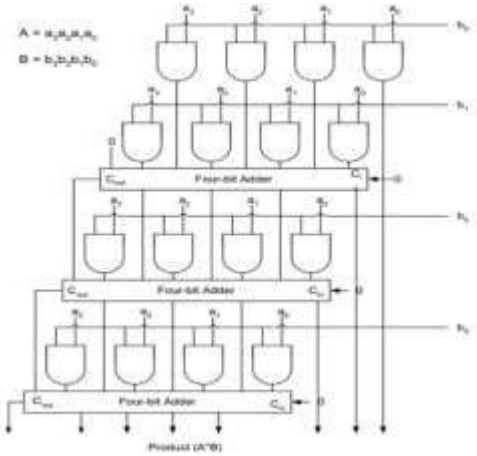


Fig 1: multiplication with adders Although the method is simple as it can be seen from this example, theaddition is done serially as well as in parallel. To improve on the delay and areathe CRAs are replaced with Carry Save Adders, in which every carry and sumsignal is passed to the adders of the next stage.

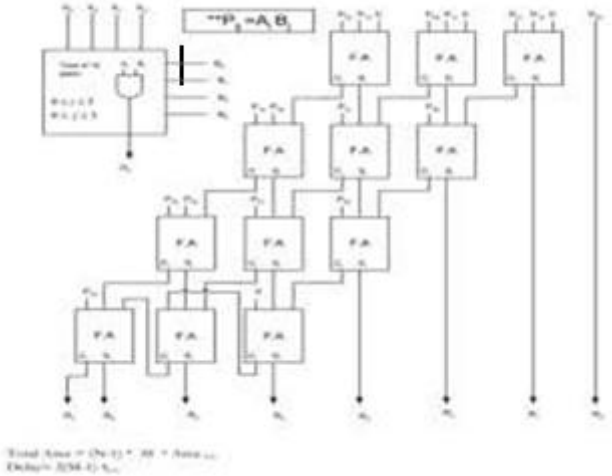


Fig 2: array multiplication with full adders

Final product is obtained in a final adder by any fast adder (usually carry ripple adder). In array multiplication, we need to add, as many partial products as there are multiplier bits. This arrangement is shown in the figure below: In applications like multimedia signal processing and data mining which can tolerate error, exact computing units are not always necessary. They can bereplaced with their approximate counterparts. Research on approximate computing for error tolerant applicationsis on the rise. Adders and multipliers form the key components in these applications. In, approximate full adders are proposed at transistor level and they are utilized in digital signal processing applications.

LITERATURE REVIEW

Vijay Kumar Reddy Modified High Speed Vedic Multiplier Design and Implementation The proposed research work specifies the modified version of binary Vedic multiplier using Vedic sutras of ancient Vedic mathematics. Itprovides modification in preliminarily implemented Vedic multiplier. The modified binary Vedic multiplier is preferable has shown improvement in the terms of the time delay and also device utilization. The proposed technique wasdesigned and implemented in Verilog

\

HDL. For HDL simulation, model sim tool is used and for circuit synthesis, Xilinx is used. The simulation has been done for 4-bit, 8-bit, 16-bit, multiplication operation. Only for 16-bit binary Vedic multiplier technique the simulation results are shown. This modified multiplication technique is extended for larger sizes. Theoutcomes of this multiplication technique is compared with existing Vedic multiplier techniques.

2. C. Liu, J. Han, and F. Lombardi,

–A Low-Power, High-Performance Multiplier with Configurable Partial ErrorRecovery‖, Proc. of IEEE Design, Automation & Test in Europe Conference& Exhibition (DATE), [Approximate circuits have been considered for error- tolerant applications that can tolerate some loss of accuracy with improved performance and energy efficiency. Multipliers are key arithmetic circuits in many such applications such as digital signal processing (DSP). In this paper, a novel multiplier with a lower power consumption and a shorter critical path than traditional multipliers is proposed for high-performance DSP applications. This multiplier leverages a newly-designed approximate adder that limits its carry propagation to the nearest neighbors for fast partial product accumulation. Different levels of accuracy can be achieved through a configurable error recovery by using different numbers ofmost significant bits (MSBs) for error reduction. The multiplier has a low mean error distance, i.e., most of the errors are not significant in magnitude. Compared to the Wallace multiplier, a 16-bit multiplier implemented in a 28nm CMOS process shows a reduction in delay and power of 20% and up to 69%, respectively. It is shown that by utilizing an appropriate error recovery, the proposed multiplier achieves similar processing accuracy astraditional exact multipliers but with significant improvements in power and performance.

PROPOSED VEDIC MATHEMATICS

Vedic Mathematics is one of the most ancient methodologies used by the Aryans in order to perform mathematical calculations. This consists of algorithms that can boil down large arithmetic operations to simple mind calculations. The above said advantage stems from the fact that Vedic mathematics approach is totally different and considered very close to the way a human mind works. The efforts put by Jagadguru Swami Sri Bharati Krishna Tirtha Maharaja to introduce Vedic Mathematics to the commoners as well as streamline Vedic Algorithms into 16 categories or Sutras needs to be acknowledged and appreciated. The Vedic algorithm is one such multiplication algorithm which is well known for its efficiency in reducing the calculations involved.

PROPOSED RECURSIVE KARATSUBA MULTIPLICATION

Recursive Karatsuba is based on incorporating Karatsuba algorithm repeatedly at every stage to improve speed when bit size is high. The algorithm works on separating the bits
(N) into groups of half-the-number-of-bits(N/2) and then following the same Karatsuba procedure with the segmented bits recursively. For a 16-bit multiplication, e.g., it breaks the number down to 8-bit multiplication, which is further divided into 4-bit and finally reduced to 2-bit which is the last stage for normal multiplication to be performed. Atevery stage we have implemented adaptive Karatsuba for the 3rd product term.

PROPOSED STANDARD KARATSUBA MULTIPLIER

Let X and Y are inputs of ‗n' bits. Assuming decomposition of X and Y into
2 equal parts; X_H, Y_H represent the higher order bits and X_L, Y_L the lower order bits. Their product can be computed

$$XY = \left(2^{\frac{n}{2}}X_H + X_L\right)\left(2^{\frac{n}{2}}Y_H + Y_L\right)$$
$$= 2^n(X_HY_H) + 2^{\frac{n}{2}}(X_HY_L + X_LY_H) + (X_LY_L) \qquad (1)$$

as:

In Karatsuba algorithm the computation is rewritten as:

$X_H Y_L + X_L Y_H = (X_H + X_L)(Y_H + Y_L) — X_H Y_H — X_L Y_L$ (2)

So, 4×n/2-bit multiplications can be reduced to 3×n/2-bit multiplications: $(X_H + Y_H)$ $(X_L + Y_L)$, $X_H Y_H$ and $X_L Y_L$.

Fig. 1 shows the standard Karatsuba multiplier at a stage when inputs are n- bits.

Time complexity of conventional multiplication algorithm requires:

$O(n) = n^2$(3)

Whereas Karatsuba multiplication algorithm requires:

$O(n) = n^{1.58}$(4)

Where n is the number of bits and O is order of complexity, considering O(1) = 1.

This shows analytically that Karatsuba algorithm with a complexity of $n^{1.58}$ (due to logarithmic dependency of n) is faster than the standard multiplication due to thelogarithmic power of n.

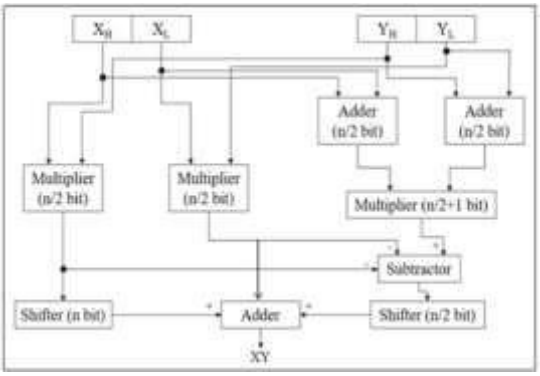


Fig.3. Standard Karatsuba Multiplier for n-Bits

VERILOG INTRODUCTION

Verilog synthesis tools can create logic-circuit structures directly from Verilog behavioral description and target them to a selected technology for realization (I.e., translate Verilog to actualhardware).

Using Verilog, we can design, simulate and synthesis anything from a simple combinational circuit to a complete microprocessor on chip.

Verilog HDL has evolved as a standard hardware description language. Verilog HDL offers many useful features for hardware design.

Verilog HDL is a general-purpose hardware description language that iseasy to learn and easy to use. It is similar in syntax to the C programming language.Designers with C programming experience will find it easy to learn Verilog HDL.

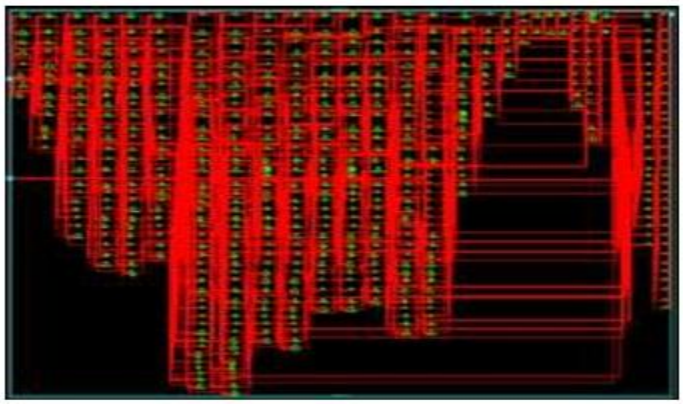
Verilog HDL allows different levels ofabstraction to be mixed in the same model. Thus, a designer can define a hardware model in terms of switches, gates, RTL, or behavioral code. Also, a designer needs to learn only one languagefor stimulus and hierarchical design.

Most popular logic synthesis tools support Verilog HDL. This makes it the language of choice for designers. All fabrication vendors provide VerilogHDL libraries for post logic synthesis simulation. Thus, designing a chip in

Verilog HDL allows the widest choice of vendors.

The Programming Language Interface (PLI) is a powerful feature that allows the user to write custom C code to interact with the internal data structures ofVerilog. Designers can customize a Verilog HDL simulator to their needs with the PLI.

\



Xilinx software is used by the VHDL/VERILOG designers for performing Synthesis operation. Any simulated code can be synthesized and configured on FPGA. Synthesis is the transformation of HDL code into gate level net list. It is an integral part of current design flows.

RESULTS

RTL SCHEMATIC:

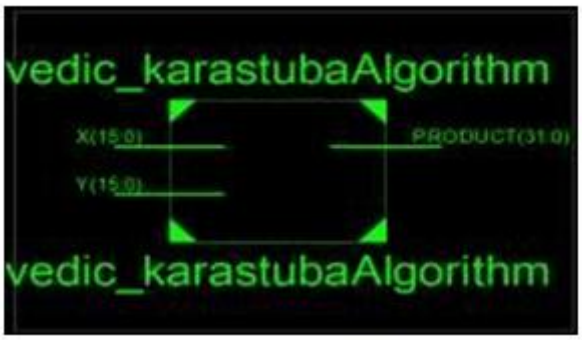


Fig: RTL Schematic of ProposedKaratsuba multiplier

The RTL schematic is abbreviated as the register transfer level it denotes the blue print of the architecture and is usedto verify the designed architecture to the ideal architecture that we are in need of development. The RTL schematic eve specifies the internal connection blocksfor better analyzing. The figure represented below shows the RTL schematic diagram of the designedarchitecture.

TECHNOLOGY SCHEMATIC

The technology schematic makes the representation of the architecture inthe LUT format, where the LUT is consider as the parameter of area that is used in VLSI to estimate the architecture design.

Fig: View Technology Schematic ofproposed Karatsuba multiplier

LUT is consider as an square unit the memory allocation of the code isrepresented in there LUT s in FPGA.

Parameter	Karatsuba Multiplier Using SRCSLA	Karatsuba Multiplier using CLA
Power (m Watt)	3.683	3.453
Number ofLUT's	208	195

SIMULATION

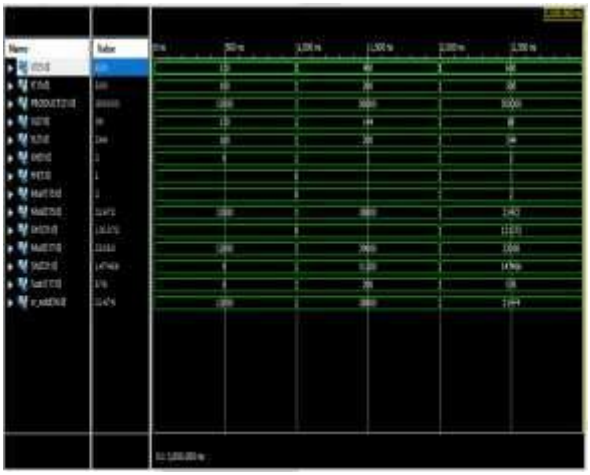


Fig: Simulated Waveforms of proposedKaratsuba multiplier

The simulation window is launchedas shifting from implantation to the simulation on the home screen of the tool,and the simulation window confines the output in the form of the wave forms.Here it has the flexibility of providing the different radix number systems.

PARAMETERS

Consider in VLSI the parameterstreated are area, delay and power, based on these parameters one can judge the onearchitecture to other. Here the area and power are considered the parameter is obtained by using the tool XILINX 14.7 and the HDL language is Verilog

The simulation is the process which is termed as the final verification inrespect to its working where as the schematic is the verification of the connections and blocks. language.

ADVANTAGES

The Karatsuba Vedic multiplier Using CLA is the fastest and novel algorithm implanted by ancestors easy to calculate than traditional and area, power are also less compared to conventional multiplier.

CONCLUSION

Vedic algorithms have been useful in designing function circuits for achieving high speed and simplified architecture. However, the challenge had been that all these algorithms are based on decimal number systems and as such binarization often led to trade-off of the speed-advantage and simplified- architecture due to conversion-hardware- overhead. Yet renewed interest has been observed recently in Vedic algorithms particularly due to clever circuit realization – primarily multipliers. The present endeavour assumes importance in that context where a known algorithm (Karatsuba) has been modified by these authors to include adaptive aspect allowing recursive operation to reduce the order of complexity from square to logarithmic value of power of bit-length.

FUTURE SCOPE

A 16×16-bit multiplier using carry look ahead adder has been proposed and designed to showcase the technique with the primary objective of minimizing the area and reducing the power consumptionso that it can find application in DSP, Image Processing and computation intensive ASIPs. It is based on the Vedic Karatsuba algorithm that generates lesser number of partial product terms. The algorithm is further optimized using adaptive concept for computation of the third product term to yield faster speed. And The implementation, synthesis and simulation are performed in XILINX-ISE tool in Verilog HDL language. In future the implementation of this multiplier employed which eliminates gate delays and adding the approximation to the architecture can enhance the performance in DSP applications, image processing, filters and cryptographic applications. Area and speed-based applications, It willbe used in future.

REFERENCES

1. Swami Bharati Krishna Tirthaji Maharaja, ―Vedic Mathematics‖, Motilal Banarsidass Publishers,

\

1965.

2. Rakshith T R and Rakshith Saligram, –Design of High-Speed Low Power Multiplier using Reversible logic: a Vedic Mathematical Approach‖, International Conference on Circuits, Power and Computing Technologies (ICCPCT- 2013), ISBN: 978-1-4673-4922-2/13, pp.775-781.
3. M.E. Paramasivam and Dr. R.S. Sabeenian, –An Efficient Bit Reduction Binary Multiplication Algorithm using Vedic Methods‖, IEEE 2nd International Advance Computing Conference, 2010, ISBN: 978-1-4244-4791-6/10, pp. 25-28.
4. Sushma R. Huddar, Sudhir Rao Rupanagudi, Kalpana M and Surabhi Mohan, –Novel High Speed Vedic Mathematics Multiplier using Compressors‖, International Multi conference on Automation, Computing, Communication, Control and Compressed Sensing(iMac4s), 22-23 March 2013, Kottayam, ISBN: 978-1-4673-5090-7/13, pp.465-469.
5. L. Sriraman and T. N. Prabakar, –Design and Implementation of Two Variables Multiplier Using KCM and Vedic Mathematics‖, 1st International Conference on Recent Advances in Information Technology (RAIT -2012), ISBN: 978-1-4577-0697-4/12.
6. Prabir Saha, Arindam Banerjee, Partha Bhattacharyya and Anup Dandapat, –High Speed ASIC Design of Complex Multiplier Using Vedic Mathematics‖, Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January,2011, IIT Kharagpur, ISBN: 978-1-4244-8943-5/11, pp.237- 241.
7. Soma BhanuTej, –Vedic Algorithms to develop green chips for future‖, International Journal of Systems, Algorithms & Applications, Volume 2, Issue ICAEM12, February 2012, ISSN Online: 2277-2677.