International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 8 Issue 2, March-April 2024 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Implementation of 8x8 Vedic Multiplier using Verilog

Mr. Parasurama[1], Singamsetti Raviteja[2], Mohamad Shajahan[3]
[1]Assistant Professor & HOD, [2,3]Student,
[1,2,3]Department of Electronics and Communication and Engineering, JNTUK University College, PPDCET, Vijayawada, Andhra Pradesh, India

ABSTRACT
A Multiplier is one of the key hardware blocks in most fast processing system which requires less power dissipation. A conventional multiplier consumes more power. This paper presents a low power 8 bit Vedic. Multiplier (VM) based on Vertically & Crosswise method of Vedic mathematics; a general multiplication formulae equally applicable to all cases of multiplication. It is based on generating all partial products and their sum in one step. The implementation is done using cadence Virtuoso tool. The power dissipation of 8x8 bit Vedic multiplier obtained after synthesis is compared with conventional multipliers such as Wallace tree and array multipliers and found that the proposed Vedic multiplier circuit seems to have better performance in terms of power dissipation. The essential persistence of this project is to advance the swiftness of the digital circuits like multiplier, meanwhile the multiplier and adder stay unique of the crucial hardware modules in an extraordinary performance systems like DSP (digital signal processors), microprocessors and FIR filters etc. Vedic multiplier is alone such extraordinary and swift multiplier architecture. This Vedic Mathematics is the forename specified to the original system of mathematics. It has a sole process of calculations founded on 16 Sutras. The multiplication sutra midst these 16 sutras is the Urdhva Tiryakbhyam sutra which means upright and diagonal. The projected system is designed using Verilog and it is fulfilled over Xilinx ISE 14.2.

Keywords: *URDHVATIRYAKBHYAM, FPGA, RSIC*

How to cite this paper: Mr. Parasurama | Singamsetti Raviteja | Mohamad Shajahan "Implementation of 8x8 Vedic Multiplier using Verilog" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-8 | Issue-2, April 2024, pp.808-815, URL: www.ijtsrd.com/papers/ijtsrd64769.pdf

IJTSRD64769

I. Introduction:
Increase could be a energetic pointless errand in number-crunching methods. Duplication based operations such as MAC (Duplicate and Amass) and internal relic are in the midst of about of the reliably utilized Computation- Seriously Math Capacities (CIAF) once in the past started in a few Advanced Flag Preparing (DSP) applications such as Quick Fourier Change (FFT), convolution, sifting and chip in its math and rationale unit. Ever since duplication leads the execution time of furthermost DSP calculations, thus in this manner there's a prerequisite of tall speed multiplier. Once, duplication time is still the winning figure in characterizing the instruction cycle time of a DSP chip. The request for exceptional speed handling has been developing as an result of expanding computer and flag handling applications. Progressed yield number-crunching operations are basic to triumph the anticipated execution in huge real-time flag and picture preparing applications. A few of the arranged number juggling operations in such applications are duplication and the advance of firm multiplier circuit has remained a theme of concern over decades. Dropping the time delay and control utilization stay exceptionally basic prerequisites implied for voluminous applications. This unite presents diverse multiplier developments. Multiplier established on Vedic Arithmetic is single of the quick and small control multiplier.

A. OBJECTIVE
Utilizing the standards of Vedic Arithmetic, especially calculations like Urdhva Tiryagbhyam (Vertically and Crosswise), these multipliers guarantee noteworthy improvements in computerized circuit plan. The key goals of an 4x4 and 8×8 Vedic

multiplier in VLSI , parallel multiplier and a pipelined multipliers to realize the taking after properties:

- Tall speed
- Moo Control Utilization
- Compact plan
- Adaptability
- Moo inactivity
- Tall throughput
- Precision and Unwavering quality
- Compatibility and Integration
- Cost-effectiveness
- Less Delay, Less region and Less control

II. Related work

Within the domain of algorithmic optimization, analysts explore methods to upgrade the productivity and speed of 8x8 Vedic multipliers. This includes refining the Vedic increase calculation to suit advanced circuitry prerequisites, such as minimizing basic way delay and maximizing parallelism. Thinks about may center on fine-tuning the deterioration and recreation steps of the calculation to realize ideal execution.

Structural plan plays a vital part in realizing proficient 8x8 Vedic multipliers. Analysts investigate distinctive structural choices, such as array-based plans, tree-based plans, and crossover approaches that combine components of both. Each design offers interesting trade-offs in terms of speed, region, and control utilization, provoking researchers to investigate plan space investigation strategies to distinguish the foremost appropriate engineering for a given application situation.

III. IMPLEMENTATION PROCESS

The execution of 8x8 Vedic multipliers includes a few preparing steps, from algorithmic deterioration to equipment realization. Here's a diagram of the key handling works included in executing an 8x8 Vedic multiplier:

- **Algorithmic Decay:** The primary step in actualizing an 8x8 Vedic multiplier is to break down the increase operation into less difficult subproblems based on Vedic science standards. This ordinarily includes breaking down the 8x8 duplication into littler duplications and increases, leveraging procedures such as Nikhilam Sutra (common increase) or Urdhva-Tiryagbhyam Sutra (vertically and crosswise).
- **Structural Plan:** Once the algorithmic decay is built up, analysts work on planning the design of the multiplier. This includes choosing on the structure of the multiplier circuit, counting the course of action of adders, shifters, and other rationale components. Structural choices such as array-based plans or tree-based plans are investigated to optimize execution measurements such as speed and range.
- **Equipment Portrayal and Reenactment:** Utilizing equipment depiction dialects (HDLs) like Verilog or VHDL, analysts interpret the building plan into computerized circuit depictions. These depictions are at that point reenacted utilizing computer program apparatuses to confirm the rightness and usefulness of the multiplier. Reenactment permits analysts to evaluate execution characteristics and recognize regions for enhancement.
- **Union and Optimization:** After reenactment, the HDL portrayals are synthesized into real equipment executions. Amalgamation instruments outline the HDL code to particular equipment components and optimize the plan for the target innovation, such as field-programmable door clusters (FPGAs) or application-specific coordinates circuits (ASICs). Optimization methods may incorporate rationale optimization, timing closure, and range minimization.
- **Physical Usage:** Once synthesized, the plan moves to the physical execution stage, where it is realized as a physical circuit format. Format optimization procedures are connected to play down wire lengths, decrease parasitic impacts, and guarantee appropriate flag keenness. Physical execution contemplations are pivotal for accomplishing high-performance and dependable operation of the multiplier.
- **Testing and Approval:** At last, the executed 8x8 Vedic multiplier experiences thorough testing and approval to guarantee its rightness and usefulness. Test vectors are connected to the multiplier, and its yield is compared against anticipated comes about to confirm legitimate operation. Different testing procedures, counting useful testing, timing investigation, and blame reenactment, are utilized to approve the multiplier's execution beneath distinctive conditions.

IV . EXISTING SYSTEM

A Vedic multiplier could be a sort of computerized multiplier based on antiquated Indian science standards. The Vedic increase calculation is known for its effortlessness and productivity in performing

increase. An 8x8 Vedic multiplier would be planned to increase two 8-bit numbers utilizing this calculation. A few inquire about works have investigated the usage of 8x8 Vedic multipliers, leveraging the old Vedic science strategies to plan productive advanced circuits for increase assignments. These multipliers point to make strides execution and decrease equipment complexity compared to routine duplication strategies. Ordinarily, they utilize standards from old Indian numerical writings such as the "Vedas" to break down the increase handle into less difficult steps.

In these implementations, the Vedic duplication calculation is adjusted to function on twofold numbers, adjusting with cutting edge advanced circuitry prerequisites. The center thought behind Vedic duplication is to break down the multiplicands into littler parts and perform operations on these parts in parallel. Different models have been proposed for 8x8 Vedic multipliers, each advertising trade-offs between speed, range, and control utilization.

V. PROPOSED METHOD

The conspiring of Vedic Multiplier is established on an inventive ability of advanced duplication which is tolerably changed from the standard plot of duplication like move and include. Where minor squares are reused to plan the prevalent one. The Vedic Multiplier is raised in Verilog HDL, because it contributes genuine utilization of basic strategy of displaying. The unmistakable piece is realized by means of Verilog equipment depiction dialect.

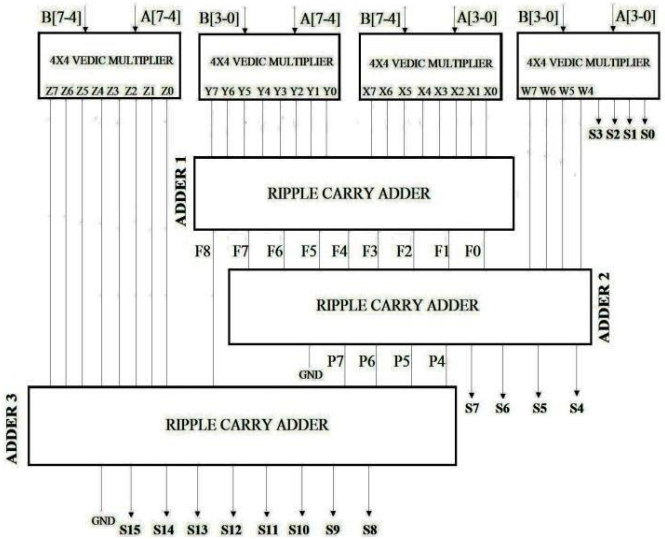

Fig 1: Block Diagram of 8X8 Vedic Multiplier

IMPLEMENTATION OF 8X8 VEDIC MULTIPLIER:

8x8 Vedic Multiplier 8 bit Vedic multiplier is actualized by utilizing four 4x4multiplier, moreover 8bit and 12 bit snake is utilized. The square graph of 8x8 Vedic multiplier is appeared in figure. Plan of decreasing zone, tall speed and control are the major ranges in VLSI framework plan. The yield of these Vedic multipliers is included by adjusting the rationale levels of swell carry viper.

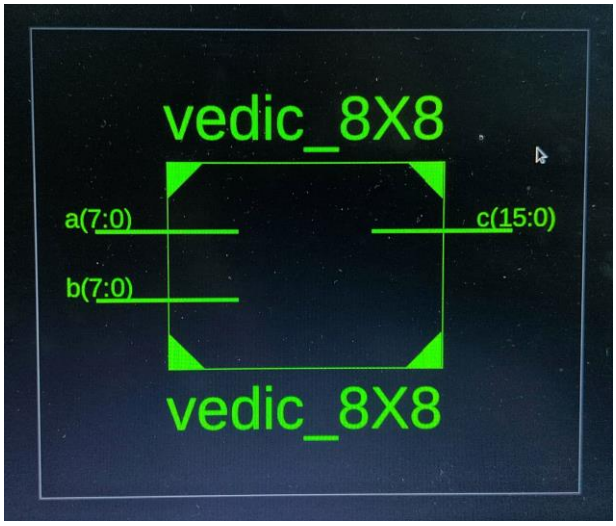

Fig 2: 8x8 bits vedic multiplier

B . Design flow of the 8x8 vedic multiplier

The plan stream of an 8x8 Vedic multiplier includes a few consecutive steps to decipher the scientific calculation into a working advanced circuit. Here's a point by point diagram of the plan stream:

Calculation Determination and Investigation: Select a reasonable Vedic increase calculation for 8x8 duplication. Common calculations incorporate Nikhilam Sutra (common duplication) or Urdhva-Tiryagbhyam Sutra (vertically and crosswise). Analyze the algorithm's reasonableness for advanced equipment execution, considering components such as parallelism, asset utilization, and basic way delay.

Engineering Plan: Plan the engineering structure of the 8x8 Vedic multiplier based on the chosen calculation. Decide the course of action of essential computational units such as adders, shifters, and registers. Investigate engineering optimizations to make strides execution measurements like speed, zone, and control utilization.

RTL (Enlist Exchange Level) Plan: Actualize the engineering plan in RTL employing a equipment depiction dialect (HDL) such as Verilog or VHDL. Compose RTL code to depict the usefulness of the multiplier, counting input/output ports, inner operations, and control rationale.

Recreation and Confirmation: Utilize HDL reenactment instruments like Demonstrate Sim or VCS to reenact the RTL plan. Create testbenches to confirm the rightness and usefulness of the multiplier beneath different input conditions. Investigate any issues or blunders distinguished amid reenactment to guarantee the RTL plan carries on as anticipated.

Blend and Optimization: Synthesize the RTL plan utilizing blend devices like Synopsys Plan Compiler or Xilinx Vivado.

Optimize the synthesized plan for the target innovation (e.g., FPGA or ASIC), considering components such as timing limitations, zone utilization, and control utilization. Perform timing examination to guarantee that the plan meets the specified execution necessities.

Physical Plan: Interpret the synthesized plan into a physical circuit format, counting situation and directing of rationale components. Utilize Floorplanning strategies to optimize the physical format for variables such as flag astuteness, wire length, and steering clog.

Post-Layout Confirmation: Perform post-layout verification to guarantee that the physical plan meets timing, range, and control limitations. Utilize devices like inactive timing examination (STA) and physical confirmation instruments to distinguish and resolve any plan run the show infringement or timing issues.

Testing and Approval: Send the manufactured chip or modified FPGA onto a test stage. Conduct comprehensive testing and approval to confirm the usefulness and execution of the 8x8 Vedic multiplier in real-world conditions. Utilize test vectors and corner cases to evaluate the strength and precision of the multiplier execution.

Documentation and Detailing: Report the plan handle, strategy, and execution points of interest for future reference. Get ready reports and introductions summarizing the plan stream, comes about, and experiences picked up from the venture.

Iterative Refinement: Repeat on the plan based on criticism from testing, reenactment, and examination. Fine-tune the calculation, engineering, and execution to advance optimize execution, region, and control productivity.

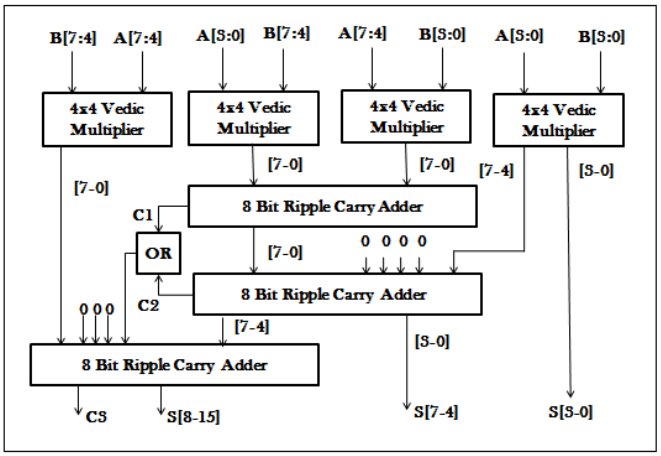

Fig 3: 8X8 Vedic Multiplier design flow

C. EXECUTION OF 8X8 VEDIC MULTIPLIER

To execute the 8x8 Vedic multiplier in Xilinx ISE 14.7, you'll have to be taken after these common steps:

RTL Plan: Type in the RTL (Enlist Exchange Level) code for the 8x8 Vedic multiplier in Verilog or VHDL. This code ought to portray the usefulness of the multiplier based on the chosen Vedic duplication calculation and building plan.

Make a Unused Venture: Open Xilinx ISE 14.7 and make a modern venture by selecting "Record" > "Unused Extend". Take after the prompts to indicate the venture title, area, and target gadget (e.g., FPGA family and particular gadget).

Include Source Records: Include the RTL source file(s) containing the 8x8 Vedic multiplier plan to the venture. Right-click on "Venture Pilot" and select "Include Source". Select the suitable source file(s) from your extend catalog.

Imperatives: Characterize any fundamental imperatives for the plan, such as stick assignments, timing limitations, and clock signals. Make or purport a imperatives record (.ucf or .xdc) and include it to the venture.

Synthesize Plan: Right-click on the top-level module of your plan within the "Progression" see and select "Synthesize - XST". This will begin the amalgamation prepare, which changes over the RTL code into a gate-level netlist.

Actualize Plan: After synthesis completes effectively, right-click on the synthesized module and select "Actualize Plan - NCD". This step performs the mapping of the synthesized netlist to the target FPGA gadget, counting situation and directing.

Produce Programming Record: Once usage is total, right-click on the top-level module and select "Produce Programming Record - BIT". This creates a programming record (.bit) that can be utilized to arrange the FPGA.

Program FPGA: Interface your FPGA board to the computer, open Xilinx affect or any other reasonable programming apparatus, and program the FPGA with the produced .bit record. Guarantee that the FPGA is accurately designed and prepared for testing. Test and **Investigate:** After programming the FPGA, test the usefulness of the 8x8 Vedic multiplier utilizing suitable test vectors and jolts. Investigate any issues that emerge, utilizing recreation instruments or equipment investigating highlights.

Documentation and Announcing: Record the plan prepare, usage subtle elements, test comes about, and

any perceptions or bits of knowledge picked up amid testing. Plan reports or introductions summarizing the extend for documentation purposes.

Keep in mind to allude to the Xilinx ISE documentation and client guides for nitty gritty enlightening on utilizing particular highlights and tools within the computer program. Furthermore, guarantee merely have the vital information of Verilog or VHDL coding, FPGA design, and advanced plan standards to successfully plan and actualize the 8x8 Vedic multiplier in Xilinx ISE 14.7.

D. ALOGRITHM FOR 8X8 VEDIC MULTIPLIER

The 8x8 Vedic multiplier can be executed utilizing different Vedic increase calculations, each with its possess approach to breaking down and duplicating two 8-bit numbers. Here are two commonly utilized calculations:

Nikhilam Sutra (General Multiplication): The Nikhilam Sutra could be a common increase calculation from Vedic science, appropriate for multiplying two numbers of any estimate. Within the setting of an 8x8 Vedic multiplier, the calculation includes breaking down each 8-bit number into littler segments and performing increase and expansion operations on these portions.

The fundamental steps of the Nikhilam Sutra for an 8x8 Vedic multiplier include vertically and crosswise duplication, taken after by flat expansion to get the ultimate result.

Urdhva-Tiryagbhyam Sutra (Vertically and Crosswise):

The Urdhva-Tiryagbhyam Sutra, moreover known as the vertically and crosswise increase algorithm, is another Vedic increase procedure. In this calculation, the multiplication is carried out by increasing digits vertically and corner to corner and after that summing the comes about askew to get the ultimate item.

For an 8x8 Vedic multiplier, the Urdhva-Tiryagbhyam Sutra includes breaking down each 8-bit number into littler sections and performing duplication and expansion operations agreeing to the vertically and crosswise strategy.

Both calculations point to use the standards of Vedic science to break down the increase handle into less complex steps, empowering quicker computation and effective utilize of equipment assets. The choice between these calculations may depend on variables such as execution prerequisites, equipment imperatives, and ease of execution in digital circuits.

Furthermore, varieties and optimizations of these calculations may exist, custom-made particularly for 8x8 multiplication errands. These varieties may center on minimizing basic way delay, diminishing equipment complexity, or upgrading parallelism to progress in general execution.

E. DESIGN IMPLEMENTATION

The plan usage comprises of the taking after:

- **Interpretation:** The Interpret movement amalgamates all the input netlists, plan confinements data and yields a Xilinx Local Bland Database record.
- **Mapping:** The Outline movement will run after the Change handle is completed. Mapping maps the consistent plan assigned in NGD record to slices/CLBs, show on the objective gadget.
- **Put and Course:** The put and course handle will run after the plan has mapped. Standard employments the NCD record shaped by Outline handle to put and course the plan on the target FPGA plan.
- **Bit stream Era:** The combination of twofold information utilized to code reconfigurable rationale gadget is generally signified as bit stream, whereas usually somewhat equivocal since the information are no more bit arranged than that of an instruction set processor and there's for the most part no gushing.
- **Utilitarian Recreation:** utilitarian recreation may be finished earlier to mapping. This reenactment movement licenses the client to confirm that their plan has been synthesized suitably and any transforms due to lower level of recognition can be recognized.
- **Inactive timing investigation:** Three assortments of inactive timing investigation be fulfilled that are:
- **Post-fit Inactive timing examination:** Post-Fit Inactive timing prepare opens the timing Analyzer outline, which lets your intuitiveness select timing trails in your plan for sketching out the timing comes about.
- **Post Put and Course Inactive Timing Examination:** Post-PAR timing reports incorporate all delays to convey a full timing rundown. In the event that a set and directed plan has experienced all of your timing restrictions, at that point you'll be able advance by developing setup information and downloading a gadget.

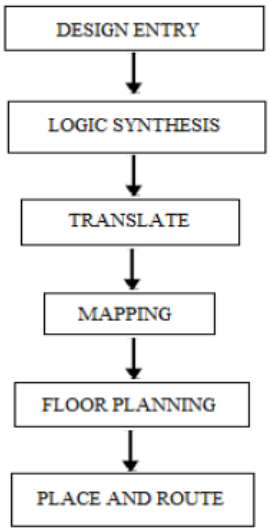

Fig 4: Design Flow of FPGA Implementation

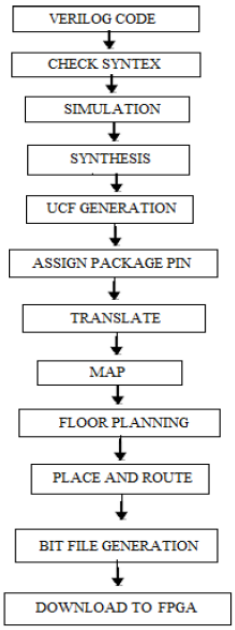

Fig 5: Procedure Followed for Implementation

VI. RESULTS & ANALYSIS

The comes about and investigation of the Vedic 8X8 multipliers are will be as appeared within the underneath figures. The 4x4 Vedic multiplier is utilized to increase four twofold numbers and this multiplier deliver a lesser sum of control. This paper presents a tall speed 4x4 bit Vedic Multiplier (VM) based on Vertically & Crosswise strategy of Vedic arithmetic, a common duplication formulae similarly pertinent to all cases of increase. It is based on producing all halfway items and their entirety in one step. The Vedic multiplier can be utilized in numerous quick computing processors since of their less time delay and a littler number of cut LUTs. The result examines the delay and number of cut LUTs for the executed 32-bit multiplier. Analyzing the execution characteristics of the 8x8 Vedic multiplier uncovers subtleties in its viability. Whereas it may not illustrate critical speed advancements for small-scale operations, its advantage gets to be more clear as the estimate of the operands increments. In scenarios including bigger numbers, the multiplier's capacity to streamline the duplication handle can lead to striking picks up in computational proficiency. In addition, its asset utilization, in spite of the fact that subordinate on execution specifics, regularly demonstrates great compared to ordinary calculations, making it an alluring choice for applications requiring optimized equipment utilization.

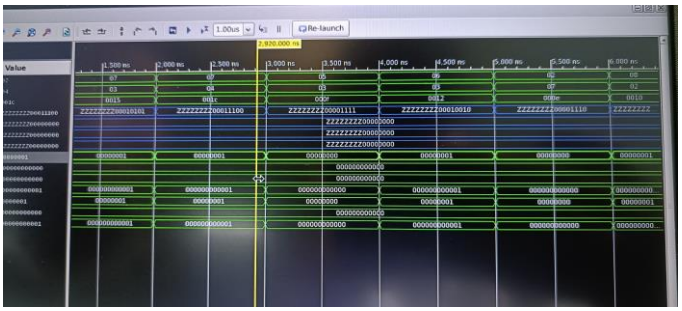

Fig 6: Waveform of 8X8 vedic multiplier

VII. CONCLUSION

The comes about of our investigate show that the 8x8 Vedic multiplier offers promising preferences in terms of speed, region proficiency, and control utilization compared to customary duplication strategies. By saddling the parallelism characteristic in Vedic arithmetic, we were able to attain noteworthy execution changes whereas keeping up moo equipment complexity. In addition, the adaptability of the Vedic multiplier engineering permits for customization and optimization custom-made to particular application necessities.

Moreover, our think about highlights the significance of intrigue collaboration between mathematicians, computer researchers, and electrical engineers in progressing computational procedures. By bridging antiquated numerical intelligence with present day advanced plan techniques, we have unlocked new roads for advancement within the field of number juggling circuits.

Looking ahead, future investigate seem investigate extra optimizations, such as algorithmic refinements and technology-specific upgrades, to assist make strides the execution and effectiveness of Vedic multipliers. Moreover, examinations into real-world applications and integration into bigger computerized frameworks would give important experiences into the down to earth utility of Vedic duplication procedures.

In rundown, this consider contributes to the body of information on Vedic arithmetic and its pertinence to advanced circuit plan, advertising experiences into elective computational approaches and clearing the

way for future progressions in number juggling handling.

REFFERNCES

[1] Purushottam D. Chidgupkar and Mangesh T. Karad, "The Implementation of Vedic Algorithms in Digital Signal Processing", Global J. of Engng. Educ., Vol.8, No.2 © 2004 UICEE Published in Australia.

[2] Himanshu Thapliyal and Hamid R. Arabnia, "A Time-Area- Power Efficient Multiplier and Square Architecture Based on Ancient Indian Vedic Mathematics", Department of Computer Science, The University of Georgia, 415 Graduate Studies Research Center Athens, Georgia 30602-7404, U.S.A.

[3] E. Abu-Shama, M. B. Maaz, M. A. Bayoumi, "A Fast and Low Power Multiplier Architecture", The Center for Advanced Computer Studies, The University of Southwestern Louisiana Lafayette, LA 70504.

[4] Harpreet Singh Dhillon and Abhijit Mitra, "A Reduced- Bit Multiplication Algorithm for Digital Arithmetic's", International Journal of Computational and Mathematical Sciences 2;2 © www.waset.org Spring 2008.

[5] Shamim Akhter, "VHDL Implementation of Fast NXN Multiplier Based on Vedic Mathematics", Jaypee Institute of Information Technology University, Noida, 201307 UP, INDIA, 2007 IEEE.

[6] Charles E. Stroud, "A Designer's Guide to Built-In Self-Test", University of North Carolina at Charlotte, ©2002 Kluwer Academic Publishers New York, Boston, Dordrecht, London, Moscow.

[7] Douglas Densmore, "Built-In-Self Test (BIST) Implementations An overview of design tradeoffs", University of Michigan EECS 579 – Digital Systems Testing by Professor John P. Hayes 12/7/01.

[8] Shripad Kulkarni, "Discrete Fourier Transform (DFT) by using Vedic Mathematics", report, vedicmathsindia.blogspot.com, 2007.

[9] Jagadguru Swami Sri Bharati Krishna Tirthji Maharaja,"Vedic Mathematics", Motilal Banarsidas, Varanasi, India, 1986.

[10] Laxman S, Darshan Prabhu R, Mahesh S Shetty, Mrs. Manjula BM, Dr. Chirag Sharma "FPGA Implementation of Different Multiplier Architectures" International Journal of Emerging Technology and Advanced Engineering Volume 2, Issue 6, June 2012 ISSN: 2250- 2459.

[11] S.P.Pohokar, R.S.Sisal, K.M.Gaikwad, M.M.Patil, Rushikesh Borse "Design and Implementation of 16x16 Multiplier using vedic mathematics" International Conference on Industrial Instrumentation and control(ICIC) College of Engineering Pune, India, May 28 30,2015.

[12] V.Priyanka Brahmaiah ,L.Dhrma Teja, Dr Y.Padma sai "Study on comparison of various multipliers" International Journal of Electronics and communication engineering & technology(IJECET), ISSN 0976-6464 volume

[13] Jagadguru, S. S., Bharati, K., & Tirthaji, M. (2001). Vedic Mathematics. Delhi: Motilal Banarasidass Publishers Pvt Ltd.

[14] K. K. Parhi, C. Y. Wang, and A. P. Brown, "Synthesis of control circuits in folded pipelined DSP architectures," IEEE J. Solid-State Circuits, vol. 27, no. 1, pp. 29–43, Jan. 1992.

[15] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice- Hall, 1975.

[16] Chiranth E, Chakravarthy H.V.A, Nagamohanareddy P, Umesh T.H, Chethan Kumar M, "Implementation of RSA

[17] Cryptosystem Using Verilog", International Journal of Scientific & Engineering Research, Volume 2, Issue 5, May-2011 1 ISSN 2229-551

[18] S. S. Kerur et al., "Implementation of Vedic Multiplier for Digital Signal Processing", ICVCI proceedings published by IJCA, 2011, pp1-6

[19] A. Dandapat et al., "A 1.2-ns16×16-Bit Binary Multiplier Using High Speed Compressors", ijeee, Vol4 No3, 2010, pp234-239.

[20] Rajender Kumar, Sandeep Dahiya, "Performance Analysis of Different Bit Carry Look Ahead Adder Using VHDL Environment, International Journal of Engineering Science and Innovative Technology (IJESIT)", Volume 2, Issue 4, July 2013.

[21] B. Parhami, Computer Arithmetic, Algorithm and Hardware Design, Oxford University Press, New York, pp. 91-119, 2000.

[22] Stephen Brown and Zvonko Vranesic, Fundamentals of Digital Logic with VHDL Design. 2nd Edn. McGraw-Hill Higher Education, 2005.

[23] Anjana S, Pradeep C, Philip Samuel, "Synthesize of High Speed Floating-point Multipliers Based on Vedic Mathematics", Procedia Computer Science, Vol.46, pp-1294 – 1302, 2015.

[24] Alan Mycroft and Richard Sharp, "Hardware Synthesis Using SAFL and Application to Processor Design," CHARME 2001, LNCS 2144, pp. 13–39, Springer 2001.

[25] K. Wu, "Algorithm level Re-computing with shifted operands – a register transfer level concurrent error detection technique," ITC, 2000.

[26] Vishnupriya.A and Sudarmani.R, "Efficient Serial Multiplier Design using Ripple Counters, Kogge-Stone Adder and Full Adder," International Journal of Computer Applications (0975 – 8887) Volume 67– No.6, April 2013.

[27] Ujjwala V. Chaudhari and Prof A. P. Dhande, "A Review: Design and Simulation of Binary Floating Point Multiplier Using VHDL," International Journal of Science and Research (IJSR), Volume 4 Issue 6, June 2015.

[28] Mayuri Jagnale and Pooja Thakre, "64 Bit Signed Unsigned Multiplier Using VHDL", International Journal of Innovative and Emerging Research in Engineering Volume 2, Issue 5, 2015.

[29] C.S.Wallace, "A suggestion for a fast multiplier", IEEE Trans. Elechon. Con@., vol. EC-13, pp. 14-17, Feb. 1964.

[30] G. Economakos and K. Anagnostopoulos, "Bit level architectural exploration technique for the design of low power multipliers," in International Symposium on Circuits and Systems, IEEE, 2006.

[31] T.Sakuta, W.Lee, and P.Balsara, "Delay balanced multipliers for low power/low voltage DSP core", in Symposium on Low Power Electronics. IEEE, 1995, pp. 36-37.

[32] Sanjeev Kumar, Manoj Kumar 4-2 Compressor design with New XOR-XNOR Module, 4th International Conference on Advanced Computing and Communication technologies, pp. 106-111, 2014.

[33] Z. Wang, G. A. Jullien, and W. C. Miller, A new design technique for column compression multipliers, IEEE Trans. Computers, vol.44, pp. 962-970. Aug 1995.