Certain Investigations on Power Optimization Techniques for Multipliers in VLSI

Prerana P. Autade[1], Dr.Anupama A. Deshpande[2], Dr.Satish M.Turkane[3]
[1]*Research Scholar, Department of Electronics & Telecommunication Engineering, Shri Jagdishprasad Jhabarmal Tibrewala University, India*
[2]*Department of Electronics & Telecommunication Engineering, Shri Jagdishprasad Jhabarmal Tibrewala University, India*
[3]*Pravara Rural Engineering College, Loni, India*

Abstract **- In VLSI circuit, space, power consumption, & speed are all significant design considerations. On overall performance of circuits, design component has contradictory effect. Compromises in various components can be used to optimise power dissipation. In VLSI circuits (such as multipliers), power consumption is also data dependent. goal of this study is to compare different design techniques & suggest modular strategy for reducing power usage. It has been discovered that algorithm-based design reduces gate switching activity and, as result, reduces multiplier power consumption. While utilising partly guarded methodology, power consumption is decreased by 10-44 percent with 30-36 percent less area overhead, while using temporal tilling method, array multiplier delay & power dissipation are observed to rise by 50 percent & 30 percent, respectively. Wallace tree multiplier recorded by Booth is determined to be 67% quicker than Wallace tree multiplier, 53% faster than Vedic multiplier, & 22% faster than radix 8 booth multipliers. For Wallace multiplier, bypassing multiplier, modified booth multiplier, & Vedic multiplier, we investigate several optimization approaches. Arithmetic operations, particularly multiplication operations, use significant amount of processing time in conventional processor central processing unit. Multiplication is fundamental mathematical operation that needs significantly more hardware & processing time than addition & subtraction.**

Index Terms **- Multipliers, VSLI Design, Reducing Technique, Power Optimization.**

INTRODUCTION

Rapid multiplication necessitates implementation of increasingly complex digital VLSI signal processing applications on single chip. Multiplier block is required for spread of contemporary Smartphone's, wearable tiny gadgets, electronics devices, & Digital Signal Processors. In VLSI digital signal processing & battery-operated power devices, multipliers are increasingly often utilised computational blocks. Multiplier can be expressed in variety of ways. [1] On performance front, improved constant multiplier architecture is required in order to create fast & low-power devices. As word size grows larger, complexity of multiplication logic grows as well, making it extremely difficult to implement using hardware. For higher order word-size, proposed constant multiplier method is investigated with least number of steps possible. Suggested pseudo code is implemented as basis of Vedic sutra & Reconfigurable Constant Multiplier to improve flavour higher order multiplier. In every known digital signal processor, multiplier is important component. Every customer wants faster gadgets with lower power usage. [2] If we can improve performance of devices by speeding up their components & lowering their power consumption, devices overall performance will improve. If correctly optimised multiplier is not utilised multiplier consumes majority of power in any digital circuit & causes latency. There are many distinct types of multipliers, each with its own algorithm & structure. Varied multipliers have different performance characteristics & each of them may be fine-tuned to get even greater results. Different types of multipliers have been invented & refined by different researchers. We're going to look at many multipliers that they've created or improved. [3] Lowering in VLSI circuits, power dissipation is critical design consideration. Design parameters have opposing effects on system's overall performance. Different optimization

techniques might be used depending on component & function. In multiplier, for example, power consumption is data dependent because gate switching activity contributes to higher power consumption. Different gate configurations can be used to optimise gate switching activity. Various strategies can be used to decrease gate switching activity. For example, with multiplier, multiplication method employed has impact on power usage. In multiplier, Booth algorithm & Modified Booth algorithm, in addition to main school technique of bit multiplication, can be employed for efficient multiplication. [4]

The gate level design of circuit can be utilised to choose between various circuit combinations & their related power consumption. Genetic Algorithm may be used to investigate various combinational circuits. In this work, several methods to power consumption in VLSI circuits are examined. Multiplier circuit is used as test case to investigate alternative techniques. Multiplication time is still most important element in determining DSP chip's instruction cycle time. With ever-increasing need for more processing power on battery-powered mobile devices, focus has switched away from maximising traditional delay time area size & toward reducing power dissipation while retaining good performance. [5] Traditionally, shift & add method has been used in design; however it is not suited for VLSI implementation or from latency standpoint. Booth multiplier, array multiplier, & Wallace tree multiplier are some of key algorithms presented in literature for VLSI implementable rapid multiplication. fast speed & little power Different logic styles can be used to construct VLSI. Power, area, & delay are three most significant factors in VLSI design. There are several suggested logics (or) low power dissipation & fast speed, & each logic style has its own speed & power benefits. [6]

Power Optimization Based on Data

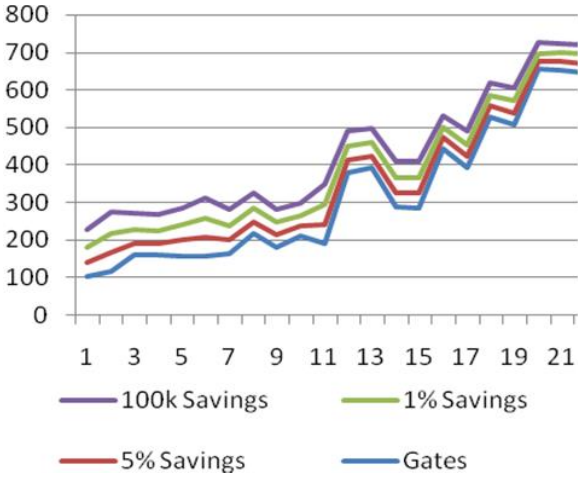
Gate switching activity in circuit is influenced by complexity of data. Design components of circuit at gate level can be minimised by using efficient computing technique. Different designs & mathematical representations are being investigated to see if they may decrease power fluctuations. Power consumption may be simulated using model. By simulating conventional designs & comparing them to best, better design component can be discovered. To assess power usage in each option, gate switching for all starting states & all inputs may be simulated. When it comes to gate design complexity, taking data dependency into account is beneficial. order in which gate inputs are connected has impact on both power & delay. Based on transistor reordering, Parsed has proposed techniques to optimise power or latency of logic gates. As result, correct transistor ordering can result in significant gains in power & latency. [7] Late arriving signals, for example, can be moved closer to output to reduce gate propagation delay. Another way to save power is to think about gate size, which has big influence on circuit delay & power dissipation. Latency of gate can be reduced by increasing size of transistors in gate, but power dissipated in gate & manufacturing space increase. As result, by correctly sized transistors, optimal equilibrium may be established. Slack at each gate in circuit is computed, & slack of gate relates to how much gate may be slowed down without impacting circuit's critical delay. Alternatively, in various sub-circuits when slack is higher than zero, size of transistors is lowered until slack is zero or transistors reach minimal size. [8]

Design of Combinational Gate Level

When designing circuit at gate level, different combinations of logical gates may give same circuit output but varying power consumption values. To reduce power consumption, path balancing, factorization, & don't care optimization can be used. By eliminating delay at each input gate, path balance may be achieved. By designing Fitness Function, Genetic Algorithm may be utilised to identify alternative combinations of gates & power usage. Genetic Algorithm is used to design combinational logic circuit. Crossover & mutation may be used to evolve diverse combinations of logic circuits by specifying chromosomal development schemes. [9] This technique is more efficient (in some circumstances & restrictions) than human designer since alternative design circuit constraints may be developed based on fitness function. Study of indicates that using Genetic Algorithm on 2-bit adder & 2-bit multiplier with certain ‘cardinality,' 56 percent decrease in number of gates for circuit may be accomplished.

IN CMOS, NUMBER OF GATES VS. POWER SAVING – BASED ON ISCAS-89 BENCHMARK CIRCUITS

The number of gates is clearly related to amount of power utilised, as seen in graph (Graph-1). As we may achieve low power consumption by reducing number of gates in design. On other hand, power reduction can be "Give & Take" situation; if we accomplish decrease in power provided, we may lose some speed, efficiency, or other characteristics, & so reducing power without sacrificing other resource parameters is need of hour. [10]



(Graph-1) Based on effort, gate vs. Power

Power Reduction in supply voltage	Speed Lose	Constraints/ Specifications
Leakage power reductions up to 54%	Not Reported	"Logic design to reduce the leakage power of CMOS circuits that use clock gating to reduce the dynamic power dissipation tested on ISCAS-89 benchmark circuits"
0.13 V or 800 times	19 times	"0.5-fim gate length and static logic"
"1.1 V supply and consumes less than 5 mW-which is more than three orders of magnitude lower power compared to equivalent commercial Solutions."	Not Reported	-------

(Table-1) Speed loss vs. power reduction

VERIFICATION OF DESIGN'S FUNCTIONALITY

The Incisive Unified Simulator from Cadence EDA Tool is used to simulate test-case verification of constant multiplier. suggested algorithm's simulation software was written in verilog HDL. To complete design time simulation for set of stimulus input values, output of design & test statistic of multiplier outcomes are proved. When input string length is up to 128 bits, simulation waveform shows that our architecture is valid, & it is scalable further. Input data (a1, b1) is forced when reset is set to one. Clock conventional multiplication constant multiplication has positive edge. [11]

RESULTS OF SYNTHESIS

The design is then synthesised at gate level using RTL compiler. Table II shows synthesis findings. Table III shows low power dissipation & rapid switching performance. [12] As result, as compared to standard multiplier, suggested method consumes significantly less power. Table II shows that utilising suggested design, power savings may be realised as bit sizes grow. In terms of latency & power consumption, synthesise result reveals that it is significantly better than various Vedic multiplier architectures. Proposed approach uses retiming technique to boost constant multiplier's speed even further. Main relevance of PDP is that it depicts constant multiplier's overall performance. Because PDP in Table-III is very tiny, it indicates minimal power dissipation & rapid switching speed. When compared to traditional multiplier, constant multiplier saves 46.34 percent in power. It also used retiming approach to accomplish high-speed multiplication. When compared to conventional multiplier & Vedic multiplier, Simultaneously Constant Multiplier has shortest latency. Our approach optimises time while decreasing design performance after re-ordering FF's. [13] In digital VLSI signal processing applications, VLSI systems are extremely helpful. Because suggested technique requires less hardware for any input size. [14]

Type of Multiplier	No. of Bits	Total Power (nW)
Conventional Multiplier	32	309360.671
	64	1519532.826
	128	7252391.568
Constant Multiplier	32	296609.771

	64	1149594.202
	128	3891931.823
Constant Multiplier with Retiming	32	298938.009
	64	1180869.161
	128	4233793.452

TABLE 2 MULTIPLIER POWER DISSIPATION

Type of Multiplier	No. of Bits	Delay (ns)	Power – Delay Product
Conventional Multiplier	32	0.13185	40789.20
	64	0.25228	383347.70
	128	0.38273	2775708.00
Constant Multiplier	32	0.13119	38912.24
	64	0.24229	278535.20
	128	0.37437	1457023.00
Constant Multiplier with Retiming	32	0.13079	39098.10
	64	0.24151	285191.70
	128	0.08587	363555.80

TABLE 3: MULTIPLIER'S POWER-DELAY PRODUCT

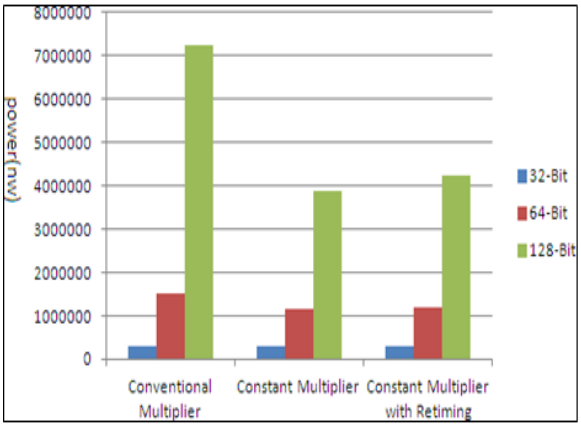


Figure 1: Power histogram for various word sizes

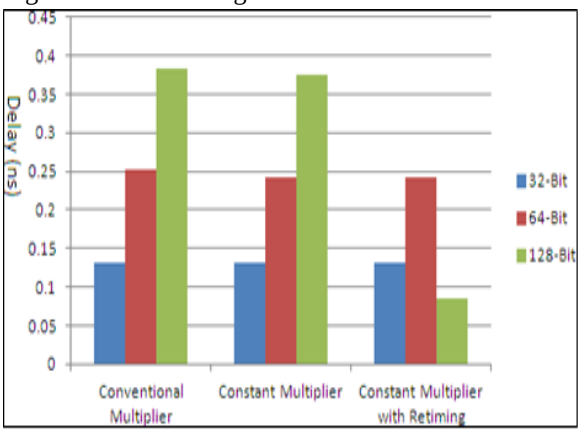


Figure 2: Delay histogram for various word sizes

CONCLUSION

Booth Multiplier is superior in every way, including speed, latency, area, complexity, & power consumption. Array Multiplier, on other hand, consumes more power & produces smallest number of components, but it has longer delay than Wallace Tree Multiplier. [15] As result, Booth's multiplier is recommended for low power & low delay requirements. Ancient Indian Vedic Mathematics provides efficient multiplication methods or equations that enhance speed of gadgets. Data complexity & various combinations of gate level digital circuits are shown to have significant influence on power dissipation. Aside from that, chip's physical design may be improved by assessing placement options using Genetic Algorithms, subject to optimum space allocation. As result of data complexity, choosing Booth Algorithm & Modified Booth Algorithm may minimise power usage. Modified Booth Algorithm is shown to consume less power in multiplier circuits than other techniques of multiplication. [16]

REFERENCES

[1] A.D. Booth, "A Signed Binary Multiplication Technique," Jour. Of Mech. Appl. Math., vol. 4, pp. 236-240, Oxford University Press,1951.

[2] Anantha P. Chandrakasan & Robert W. Brodersen, fellow, IEEE, "Minimizing Power Consumption in Digital CMOS Circuits" proceedings of IEEE, vol. 83, No. 4, April 1995.

[3] Dake Liu & Christer Svensson, "Trading Speed for Low Power by Choice of Supply & Threshold Voltages, "IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 28, NO. I . JANUARY 1993.

[4] Dake Liu & Christer Svensson, "Trading Speed for Low Power by Choice of Supply & Threshold Voltages" IEEE Journal of Solid-State Circuits, vol. 28, no. I, January 1993.

[5] Fayez Elguibaly, "A Fast Parallel Multiplier-Accumulator Using Modified Booth Algorithm", IEEE Trans. Circuits & Sys. II, Analog & Digital Signal Processing, vol. 57, no. 9, pp. 902-909, Sept., 2000.

[6] George Economakos, Dimitris Bekiaris & Kiamal Pekmestzi, "A Mixed Style Architecture for Low Power Multipliers Based on Bypass Technique", IEEE Int. Conf. on Design & Technology of Integrated Syst. in Nanoscale, pp. 1-6, March, 2010.

[7] Jin Tai Yan & Zhi-Wei Chen, "Low-Cost Low-Power Bypassing- Based Multiplier Design", IEEE Proc. Int. Symp., Circuits & Systems, pp. 2338-2341, May, 2010.
[8] Jonathan P. Halter & Farid N. Najm "A Gate-Level Leakage Power Reduction Method for Ultra-Low-Power CMOS Circuits" Custom Integrated Circuits Conference, 1997. Proceedings of IEEE 1997, pp. 475-478.
[9] K. Y. Chao & D. F. Wong. "Lowpower consideration in floor plan design." in Proceedings of 1994 International Workshop on Low Power Design, pages 45-50, April 1994.
[10] L. Bisdounis, S. Nikolaidis, & O. Koufopavlou, "Propagation delay & short–circuit power dissipation modeling of CMOS inverter," IEEE Transaction on Circuits & Systems, vol. 45, pp. 259–270, Mar.1998.
[11] Naresh R. Shanbhag "A Fundamental Basis for Power-Reduction in VLSI Circuits" Coordinated Science Laboratory/ECE Dept. University of Illinois at Urbana- Champaign, Urbana, IL 61801.
[12] O.L. Macsorley, "High-Speed Arithmetic in Binary Computers", Proc. IRE, vol. 49, no. 1, pp. 67-91, Jan., 1961.
[13] Ravindra P Rajput & M.N Shanmukha Swamy, "High speed Modified Booth Encoder multiplier for signed & unsigned numbers", 14th IEEE Int. Conf. on Modelling & Simulation, pp. 649-654, 2012.
[14] Shiann-Rong Kuang & Jiun-Ping Wang, " Design of Power-Efficient Configurable Booth Multiplier", IEEE Trans., Circuits & Sys. II Regular Papers, vol. 57, no. 3, pp. 568-580, March, 2010.
[15] Wen-Chang Yeh & Chein-Wei Jen, "High-Speed Booth Encoded Parallel Multiplier Design", IEEE Trans. on Compt., vol. 49, no. 7, pp. 692-701, July, 2000.
[16] Zhou Shun, Oliver A. Pf¨ander, Hans Jorg Pfleiderer & Amine Bermak, "A VLSI architecture for Run-time Multi-precision Reconfigurable Booth Multiplier", 14th IEEE Int. Conf. on Electronics, Circuits & Systems, pp. 975-978, Dec., 2007.