Volume 1, Issue 2, April 2014 ISSN: 2348-8565 (Online)
International Journal of Modern Engineering and Research Technology
Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

High Performance ALU Design Implementation on FPGA Using Ancient Egyptian Multiplier

Surya Kumar Chaturvedi
M. Tech. Scholar,
Shri Ram Institute of Technology
Jabalpur, (M.P) [India]
Email: schaturvedi326@gmail.com

Divayanshu Rao
Assistant Professor
Shri Ram Institute of Technology
Jabalpur, (M.P) [India]

Prof. Ravi Mohan
Associate Professor,
Department of Electronics & Communication Engg,
Shri Ram Institute of Technology,
Jabalpur, (M.P.) [INDIA]
Email: ravimohan7677@yahoo.co.in

ABSTRACT

A Hybrid FPGA have fix course grain modules and it use wide data paths, like for any instance 32 bits wide resources, sometime microprocessor CPUs or a data-stream-driven data path units also known as coarse-grained computing. Choice of FPGA affects our design in term of area and speed so we select a hybrid FPGA for faster and area efficient computation. A FPU module has three sub-modules FM, FA & WB, optimizing these we can optimized overall design we have gone through multiple approaches for floating multiplication and floating addition. We plan to use Peasant Multiplication algorithm for multiplication, also known Ancient Egyptian multiplication and proposed tree adder for addition and we will use course grain of Virtex -II for all logical operations. We have proposal to merge Peasant & Tree Adder techniques for designing FM & FA sub -modules and to design top module of design hierarchy including coarse grain logic modules WB's along with FA & FM. There top module is a 16 bit FPU module.

Keywords:—*Field Programmable Gate Arrays (FPGAs), Configurable Logic Block (CLB). VHDL (VHSIC Hardware Description Language), Integrated Software Environment (ISE), Floating Point Unit (FPU), Look up Table (LUT)Digital Signal Processing (DSP),*

I. INTRODUCTION

A floating-point unit (FPU) is a part of a computer system specially designed to carry out operations on floating numbers. Some systems (microcode-based architectures) can also perform various transcendental functions such as exponential or trigonometric calculations, though in many advance processors these are done with software library routines.

Various FPGA's are available now a day's which give us good hands on research work in the field of ASIC designing. And as we knew a hard core ASIC will gives us a better throughput over the software based library routine if our application is specific. If our application is defined then there is nothing to trade off, for better performance FPGA based IP (Intellectual Property) is better choice.

FPGA Architecture for Floating Point:

In general, FPGA-based floating-point application circuits can be divided into control and data path portions. The data path typically contains floating-point operators such as adders, subtractors, and multipliers, and occasionally square root and division computation. The data line generally occupies most of the area in an implementation of the application. The control circuit is usually much simpler than the data path, and therefore, the area consumption is typically lower. Control is usually implemented as a finite-state machine and most FPGA synthesis tools can produce an efficient mapping from the Boolean logic of the state machine into fine-grained FPGA resources.

Hybrid FPGA: A hybrid FPGA combination of coarse-grained and fine-grained elements, which can connect by various routing tracks. Fine grained fabric normally has an array of identical configurable logic blocks (CLBs). This architecture is similar to the Xilinx Vertex - IV FPGA slice. [3]

Coarse-grained units are more optimized than fine-grained programmable logic if word-level operations required for implementation. As example, an application which requires very high performance floating point calculations can have better speed and density by comprising embedded floating point units (FPUs). Floating number adders & sub-tractors (FAs) and floating point multipliers (FMs).

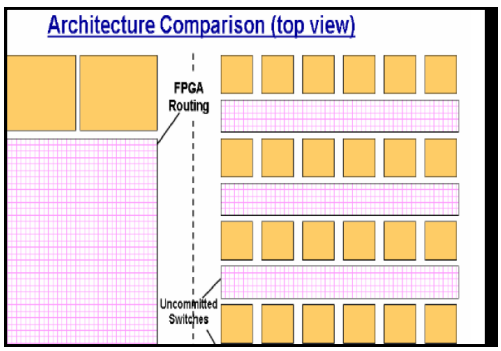

Figure 1: Connecting WBs, FAs and FMs into Coarse-Grained FPU

II. Literature Review

Mr. Chi Wai Yu along with Alastair M. Smith and Wayne Luk published a paper 'Optimizing Floating Point Units in Hybrid FPGAs' there Methodology was to design efficient coarse-grained floating point units (FPUs) at hybrid field-programmable gate array (FPGA) The speed of the system is the very high for design use only FAs and FMs, FPU architecture efficient design shows that though high density FPUs have less speed, they manage advantages and improved area, area-delay product, also throughput.

Mr. Z. Babic along with Mr. A. Avramovic and P. Bulic publish paper entitle 'An iterative logarithmic multiplier' there method was to design Digital signal processing algorithms as DSP often have heavily and have large number of multiplications, and it is time consuming These type of methods consume less time and less power but with some errors. So, it can be used in condition where a shorter time delay is more priority than accuracy.

III. Implementation

Paper work proposing technique to optimize coarse grained floating point units (FPU) on hybrid type field-programmable gate array (FPGA), where the FPU have many interconnected floating point adders (FAs) also floating point multipliers (FMs). The word blocks (WB) include registers and CLB or LUT, which can implement fixed point operations proficiently. Propose work to implement method based on Peasant algorithm for an efficient FM and for addition we use Tree Adder for FA. Xilinx course grain module is the best mix of blocks within an FPU and to study the speed, area and utilization bargaining over a set of floating point test benchmark circuits. We will also then discuss the impact on system if FPU density and flexibility modified as proposed in terms of speed, area and routing resources. In last, we will derive an efficient coarse-grained FPU by with

considering each architectural and system-level issue. This kind of proposed technique can be used to evaluate a number of FPU architecture optimizations. FM is floating multiplication and proposed work used Peasant algorithm with required modification done by us in it.

For Floating Point Multiplication proposed technique is designed by Peasant Multiplication it also known ancient Egyptian multiplication technique for making multiplication of two numbers it does not require the multiplication table, it has only ability to multiply and divide by 2,can be easily obtain by shifting left or right and also need addition. This method is known as Peasant multiplication, coz it has been normally used among those who are unschooled and so never memorized the multiplication tables which required long multiplication. The algorithm was also in used by ancient Egyptian mathematician.

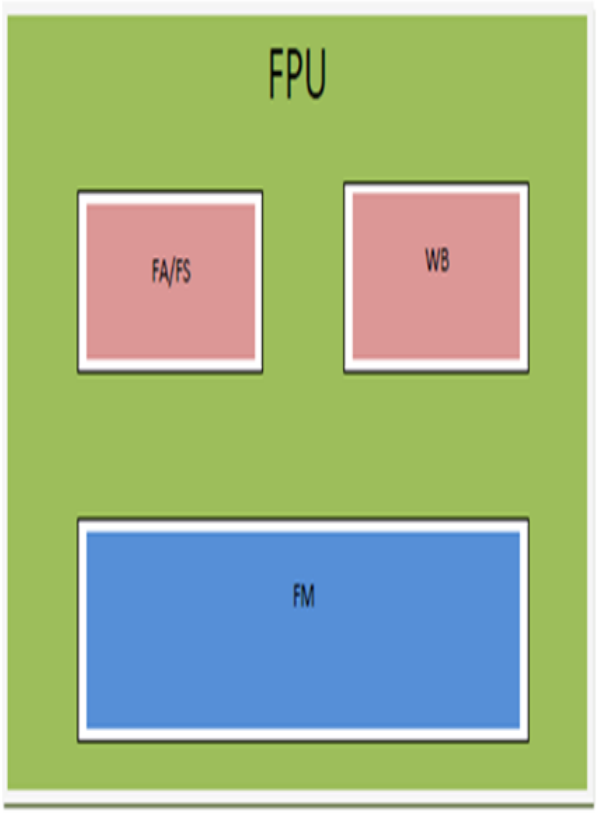

Figure 2: Top View of Proposed design

IV. RESULTS

Table-1: Area & Delay Estimates for Proposed Multipliers

No. of Slices	128
No. of 4 bit LUT	321
IOBs	70
Power Quiescent	333mw
Delay	10.642 ns
Max Frequency	93.967 MHz

Table show the results which are been observed for the proposed multiplication of two 16 bit numbers as can observe that is requires only 321 slices and it works on 94 M. hz frequency as it is better than other work when area concerns.

IV. CONCLUSION

Thesis work measures the area (in LUTs) & delay (in ns) for ancient Egyptian Ancient Egyptian Multiplier, also compare proposed design results with Tree Multiplier & Array Multiplier. And thesis work can conclude that proposed design of Floating Point Unit in Hybrid FPGA along with Ancient Egyptian Multiplier is requires lesser amount of Area (321 slices) & time (10.62 ns) as compared to previous designs. As compared to the tree and array multipliers in reference.

REFERENCES:

[1] "Optimizing Floating Point Units in Hybrid FPGAs" ChiWai Yu, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, July 2012, PP1295-130 Vol. 20, No. 7

[2] "Optimizing Coarse-Grained Units in Floating Point Hybrid FPGA", Chi Wai Yu, Alastair M. Smith, Wayne Luk, Philip H.W. Leong, Steven J.E. Wilton, IEEE, pp57-64,2008.

[3] "Enhanced-functionality multipliers", Mustafa Gok, Science Direct, 2 Feb 2008.

[4] "Design and implementation of a CMOS non-restoring divider", Kudithipudi, Dhireesha; John, Eugene, Region 5 Conference, pp 211 -2172006 IEEE.

[5] "Pipelined Floating-Point Arithmetic Unit (FPU) for Advanced Computing Systems using FPGA", Rathindra Nath Giri, M.K.Pandit, Volume-1, Issue-4, (IJEAT) ISSN: 2249 – 8958, April 2012.

[6] Computer Organisation and Structure, Virtual-Labs, IIT Khadagpur, http:// virtual-labs.ac.in/labs/cse10/wta.html

[7] "Floating-Point FPGA: Architecture and Modeling", Chun Hok Ho, ChiWai Yu, Philip Leong, Wayne Luk, and Steven J. E. Wilton, IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, DECEMBER 2009, pp 1709- 171.VOL. 17, NO. 12.

[8] "A tool for exploring hybrid FPGAs", Chi Wai Yu, in Proc. FPL, pp. 509–510, 2007.

[9] "Hybrid FPGA Architecture", Alireza Kaviani and Stephen Brown, IEEE.

[10] FPGA-Based System Design, Wayne Wolf, Pearson Education Pearson Education India, 2004

* * * * *