ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 5, Issue 2, February 2018

Multi-Precision Floating Point Arithmetic Logic Unit for Digital Signal Processing: A Review

[1] Shaikh Shoaib Arif, [2]Dr.B.B.Godbole,
[1] Assistant Professor, Deogiri Engineering College Aurangaba , [2]Associate Professor, K.B.P.C.O.E. Satara

Abstract: **-- In a wide range of DSP applications includes processing of sensor array processing, audio and speech signal processing, control of systems, radar and sonar signal processing, spectral estimation, digital image processing, seismic data processing, biomedical signal processing, statistical signal processing, signal processing for communications, Filter designing & many high accuracy based operations. Floating point operations are used due to its huge dynamic range, high accuracy and straightforward operation rules. With the increasing needs for the floating point operations for the high-speed signal processing and the scientific operation, the requirements for the high-speed hardware floating point arithmetic units have become more useful.**

Index Terms— Arithmetic Logic Unit, Digital Signal Processing, Floating Point, FPGA, Multiplier, Super computing, Synergistic Processor, Quadruple Precision,

I. INTRODUCTION

The implementation of the floating point arithmetic has been appropriate within the floating point high level languages; however the execution of the arithmetic by hardware is difficult task. With the expansion of the very large scale integration (VLSI) technology have become the most effective choices for implementing floating hardware arithmetic units due to their high integration density, high performance, low worth and versatile applications needs for prime precious operation.
The IEEE 754 standard presents two completely different floating point formats, Binary interchange format and Decimal interchange format. This section focuses solely on single precision normalized binary interchange format. Figure 1 shows the IEEE 754 single precision binary format representation, it consists of a one bit sign (S), an eight bit exponent (E), and a twenty three bit fraction (M) or Mantissa.

II. VFLOAT: A VARIABLE PRECISION FIXED- AND FLOATINGPOINT LIBRARY FOR RECONFIGURABLE HARDWARE

In variable precision floating-point library (VFloat) that supports general floating-point formats as well as IEEE standard formats. optimum reconfigurable hardware implementations could need the utilization of arbitrary floating-point formats that don't essentially adjust to IEEE standard sizes. Most antecedently printed floating-point formats to be used with reconfigurable hardware square measure subsets of our format. Custom data paths with optimum bit widths for every operation may be designed mistreatment the variable exactitude hardware modules within the VFloat library, enabling a better level of similarity. The VFloat library includes three varieties of hardware modules for format management, arithmetic operations, and conversions between fixed-point and floating-point formats. The format conversions gives hybrid fixed- and floating-point operations during a single style [1].

III FAST , EFFICIENT FLOTING POINT ADERS AND MULTIPLIERS FOR FPGA

In implementation details for Associate an IEEE-754 floating-point adder Multiplier for FPGAs and FPU applications a growing trend within the FPGA community. As such, it's become vital to form floating-point units optimized for FPGA technology. the FPGA style area is completely different from the VLSI style space; so, optimizations for FPGAs will take issue considerably from optimizations for VLSI. specifically, the FPGA setting constrains the planning area such solely restricted similarity may be effectively exploited to scale back latency.
Obtaining the correct balances between clock speed, latency, and space in FPGAs may be notably difficult. The styles given here modify a Xilinx Virtex4 FPGA (-11 speed grade) to attain 270 MHZ IEEE compliant double exactitude floating-point performance with a 9-stage adder pipeline and 14-stage multiplier pipeline. the world demand is close to 500 slices for the adder and beneath 750 slices for the multiplier [2].

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 5, Issue 2, February 2018

IV SPEED-UP IN FPGA BASED BIT-PARALLEL MULTIPLIERS

This technique take into account the technology-dependent optimizations of fixed-point bit-parallel multipliers by completing their implementations by considering embedded primitives and macro support that are useful in modern-day FPGAs. FPGAs are the best option proving to be replacement of Application Specific Integrated Circuits (ASIC) primarily due to the low Non-recurring Engineering (NRE) prices related to FPGA platforms.

This has prompted FPGA vendors to enhance the capability of the underlying primitive material and embody specialised macro support and material possession (IP) cores in their offerings. However, most of the work associated with FPGA implementations doesn't take full advantage of those offerings. Their implementation targets three completely different FPGA families viz. Spartan-6, Virtex-4 and Virtex-5. The implementation results indicate that a substantial speed up in performance will occur these embedded FPGA resources.

The speedy evolution of reconfigurable computing places a good demand for Floating purpose Multipliers (FPMs) capable of supporting big selection of application domains from scientific computing to multimedia system applications. whereas former wants the support of upper exactitude formats like Double Precision(DP) / Extended Precision(EP), the latter wants Single Instruction Multiple information (SIMD) feature in Single exactitude (SP) mode[3].

V. A DUAL-MODE QUADRUPLE PRECISION FLOATING POINT DIVIDER

This section presents a multi-mode floating point multiplier operating efficiently with every precision format specified by the IEEE 754-2008 standard. The design performs one quadruple precision multiplication, or two double precision multiplications in parallel, or four single precision multiplications in parallel.

The proposed multiplier is pipelined to achieve implementation of one quadruple multiplication in 3 cycles and either two double precision operations in similar or four single precision operations in similar in only 2 cycles. The planned design improves the throughput by a factor of two compared to a double precision multiplier and by four compared to a single precision multiplication. An example execution on VLSI verifies the plan and it achieves a maximum operating frequency of 505 MHz [4].

VI.A DUAL-MODE QUADRUPLE PRECISION FLOATING POINT DIVIDER

Many scientific applications need additional correct computations than double precision or double-extended precision floating-point arithmetic. The design of a dual-mode quadruple precision floating-point divider that also supports two parallel double precision divisions. A radix- 4 SRT division algorithm with negligible redundancy is used to implement the dual-mode quadruple precision floating-point divider.

To approximation area and bad case delay, a double, a quadruple, a dual-mode double, and a dual-mode quadruple precision floating-point division units are implemented in VHDL and synthesized. The synthesis results show that the dual-mode quadruple precision divider requires 22% more area than the quadruple precision divider and the bad case delay is 1% more. A quadruple precision division takes 59 cycles and two parallel double precision division take 29 cycles.

A technique and modifications used to plan the dual-mode quadruple precision adder are applied to execute a dual-mode double precision adder, which supports one double precision and two similar single precision operations. To estimate area and worst case delay, the usual and the dual-mode double and quadruple precision adders are implemented in VHDL and synthesized.

The exactness of all the designs is also tested and verified through extensive simulation. Synthesis results show that the dual-mode quadruple precision adder requires roughly 14% more area than the conventional quadruple precision adder and a worst case delay is 9% more [5].

VI. ACCURACY PARAMETERIZABLE LINEAR EQUATION SOLVERS

In section FPGA flexibility within the context of fast the answer of the many little systems of linear equations, a tangle central to model prognostic management (MPC). the most observation exploited by this work is that the distinction between accuracy (meaning the degree of correctness of a

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 5, Issue 2, February 2018

final procedure result) and exactitude (meaning the degree of correctness of every atomic computation).

Using unvaried strategies for finding linear systems, one will get improved accuracy either by running additional iteration or by mistreatment additional precise internal computations, in contrast to direct strategies, wherever accuracy is simply a operate of operation exactitude. Thus, in unvaried strategies, for a given accuracy demand author conduct fewer iterations during a higher exactitude, or additional during a lower exactitude. we tend to argue that this suits FPGA architectures ideally, as low exactitude operations end in bigger similarity for any fastened space constraint.

They show that they'll so optimize the performance by equalization iteration count and operation exactitude, leading to a several-fold speed improvement over a double-precision implementation, however with a similar ending accuracy. Exploring this trade-off it's doable to produce a speed-up of twenty six X on the average, 14Xin the worst case and 36X within the best, compared to a high-end computer hardware running at three.0 GHz. This has the potential to permit fashionable high performance management techniques to be utilized in novel settings like craft and diesel engines [6].

In this section examine FPGA flexibility within the context of fast multiple solutions of little systems of linear equations, with application to Model prognostic management. Model prognostic management (MPC) may be a technique used for dominant multi-variable propelling systems. This technique has become Associate in Nursing business customary (mainly within the organic compound industry) thanks to its intrinsic capability for handling onerous constraints. the most plan behind MPC is to decide on the management action by repeatedly finding Associate in Nursing optimum management drawback over a particular horizon. From this resolution, solely the primary action is enforced. This action may be affected to an outlined vary representing real physical limitations. a brand new output sample is then measured and therefore the method is perennial. [7].

In the second class, acceleration is provided by exploiting mixed-precision operations. The key plan behind these mixed-precision schemes is to perform as several operations as doable in lower exactitude, and solely execute variety of crucial operations within the slower high-precision [8]. Studies have shown that these schemes will get precisely the same resolution accuracy as if the whole computation was performed in high-precision [9]. Authors have enforced Associate in Nursing FPGA-based mixed-precision convergent thinker mistreatment the Cray-XD1 and over that it's doable to attain 2 to a few times higher performance than a double-precision style running on a computer hardware [10].

In distinction, this work uses one operative exactitude, however expressly tunes this exactitude to optimize performance: too low and therefore the rule can take an excessive amount of iteration to converge, too high and therefore the similarity obtainable are going to be reduced. the chance to use this trade-off between precision and iteration count with similarity arises solely in iterative methods. This work exploits this trade-off technique in the context of fast Model prognostic management.

VIII. FPGA SUPERCOMPUTING

For certain applications, custom procedure hardware created mistreatment field programmable gate arrays (FPGAs) produces vital performance enhancements over processors, leading some in domain and business to necessitate the inclusion of FPGAs in supercomputing clusters. This section presents a comparative analysis of FPGAs and ancient processors, specializing in floating purpose performance and procurance prices, revealing economic hurdles within the adoption of FPGAs for general superior Computing (HPC).
Supercomputers have full-fledged a recent revival, oxyacetylene by government analysis bucks and therefore the development of inexpensive supercomputing clusters. in contrast to the Massively Parallel Processor (MPP) styles found in Cray and agency machines of the 70s and 80s, that includes proprietary processor architectures, several fashionable supercomputing clusters square measure created from goods computer processors, considerably reducing procurance prices [11].

In an endeavor to enhance performance, many corporations provide machines that place one or additional FPGAs in every node of the cluster. Configurable logic devices, of that FPGAs square measure one example, allow the device's hardware to be programmed multiple times once manufacture. a large body of analysis over 20 years has repeatedly incontestable vital performance enhancements for sure categories of applications once enforced among Associate in Nursing FPGA's configurable logic[12].

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 5, Issue 2, February 2018

VII. HIGH-RADIX IMPLEMENTATION OF IEEE FLOATING-POINT ADDITION

In this section proposing a micro-architecture for high performance IEEE floating-point addition that's supported a (non redundant) high-radix illustration of the floating purpose operands. the most improvement of the projected IEEE FP addition implementation is achieved by avoiding the computation of full alignment and standardisation shifts that impose major delays in typical implementations of IEEE FP addition.

This reduction is achieved at the price of wider quantity interfaces Associate in Nursing an inflated complexness for IEEE compliant rounding error. They gift a close discussion of Associate in Nursing IEEE FP adder implementation mistreatment the projected high-radix format and justify the particular edges and challenges of the planning.

Floating-point addition and subtraction square measure the foremost frequent floating-point operations. each operations use a floating-point (FP) adder. Therefore, latency and turnout of FP adders square measure vital for superior FP support and ton of effort has been spent on rising the performance of FP adders. We square measure demonstrating the chances for improvement of floating-point addition concerning the new format. We present a close discussion of Associate in Nursing IEEE FP adder implementation mistreatment the projected high-radix format and justify the particular edges and challenges of the planning[13].

IX. FLOATING-POINT UNIT IN THE SYNERGISTIC PROCESSOR

The floating-point unit (FPU) within the synergistic processor part (SPE) of a CELL processor may be a absolutely pipelined 4-way single-instruction multiple-data (SIMD) unit designed to accelerate media and information streaming with 128-bit operands. It supports 32-bit single-precision floating-point and 16-bit number operands with 2 completely different latencies, six-cycle and seven-cycle, with eleven FO4 delay per stage.

The FPU optimizes the performance of vital single-precision multiply-add operations. Since precise rounding error, exceptions, and de-norm range handling don't seem to be vital to multimedia system applications, IEEE correctness on the single-precision floating-point numbers is sacrificed for performance and easy style. It employs fine-grained clock gating for power saving. the planning has 768K transistors in one.3 mm2, made-up SOI in 90-nm technology. Correct operations are ascertained up to five.6 GHz with one.4 V and 56°C, delivering forty four.8 GFlops. design, logic, circuits, and integration square measure co-designed to satisfy the performance, power, and space goals [Fully, 14]

X. HARDWARE IMPLEMENTATIONS OF DENORMALIZED NUMBERS

Denormalized numbers the foremost troublesome form of numbers to implement in floating-point units. they're therefore advanced that some styles have elective to handle them in software package instead of hardware. This has resulted in execution times within the tens of thousands of cycles, that has created denormalized numbers useless to programmers. This doesn't got to happen. With atiny low quantity of further hardware, denormalized numbers and underflows may be handled about to the speed of normalized numbers. we are going to summarize the miscroscopic famous techniques for handling denormalized numbers. Most of the techniques mentioned have solely been mentioned in filed or unfinished patent applications[15].

In this section gift the planning and implementation of a floating-point adder that's compliant with this draft revision of this customary. they supply synthesis results indicating the calculable space and delay for our style once it's pipelined to numerous depths.Their work is a vital style resource for development of floating-point adder hardware on FPGAs. All sub elements among the floating-point adder and famous algorithms square measure researched and enforced to produce skillfulness and suppleness to designers as another to material possession wherever they need no management over the planning. The VHDL code is open supply and might be utilized by designers with correct reference. every of the sub-operation is researched for various implementations then synthesized onto a Spartan FPGA device to be chosen for best performance.Our implementation of the quality rule occupied 370 slices Associate in Nursingd had an overall delay of thirty one ns. the quality rule was pipelined into fivestages to run at a hundred megacycle per second that took a region of 324 slices and power is 30mw [16].

In these section projected a unique design that includes synchronous , speculative computation of 3 doable results of

Associate in Nursing FP addition. These results square measure typified by the time complexities of their operations: the hardware realization of bound cases of floating purpose addition may be simplified so the execution time of such operations is reduced. With this, the variable latency design produces results among one, two or three cycles, by virtue of that the typical latency of FP additions is reduced [17,18,19].

CONCLUSION

Improvement in Floating-point operations by minimizing the time consumed for FPU operations, power consumed in floating point operations and space utilization which will enhance the working of digital signal processing & other many operations. Existing floating point operations have limitations that it can implement on only one type of hardware either 32 bits, 64 bits & 128 bits. for different number of bits architectures must be change this problem can be solve by making a inbuilt architectures which support 32 bits, 64 bits & 128 bits of operations. This improvement in hardware & software will be utilized in digital signal processing units, sonar and radar signal processing, sensor array processing, spectral estimation, statistical signal processing and high precision based applications and results will improve.

REFERENCES

[1] X. Wang and M. Leeser, "Vfloat: A variable precision fixed- and floating point library for reconfigurable hardware," ACM Trans. Reconfigurable Technol. Syst., vol. 3, no. 3, pp. 16:1–16:34, Sep. 2010.

[2] K. S. Hemmert and K. D. Underwood, "Fast, efficient floating-point adders and multipliers for FPGAs," ACM Trans. Reconfigurable Technol.Syst., vol. 3, no. 3, pp. 11:1–11:30, Sep. 2010.

[3] A. Baluni, F. Merchant, S. K. Nandy, and S. Balakrishnan, "A fully pipelined modular multiple precision floating point multiplier with vector support," in Proc. ISED, 2011, pp. 45–50.

[4] K.Manolopoulos, D. Reisis, and V. Chouliaras, "An efficient multiple precision floating-point multiplier," in Proc. 18th IEEE Int. Conf. Electron.,Circuits Syst., 2011, pp. 153–156.

[5] A. Isseven and A. Akkas, "A dual-mode quadruple precision floatingpoint divider," in Proc. 40th ACSSC, 2006, pp. 1697–1701.

[6] A. R. Lopes, A. Shahzad, G. A. Constantinides, and E. C. Kerrigan, "More flops or more precision Accuracy parameterizable linear equation solvers for model predictive control," in IEEE Symposium on Field Programmable Custom Computing Machines, Napa, California, 2009.

[7] J. Maciejowski, "Predictive Control with Constraints,"Prentice Hall, Pearson Education Limited, Harlow, UK, 2001.

[8] R. Strzodka and D. G¨oddeke, "Pipelined mixed precision algorithms on FPGAs for fast and accurate PDEsolvers from low precision components," in IEEE Symposiumon Field-Programmable Custom ComputingMachines (FCCM 2006), Apr. 2006, pp. 259–268.

[9] A. Buttari, J. Dongarra, J. Kurzak, P. Luszczek, and S. Tomov, "Using mixed precision for sparse matrix computations to enhance the performance while achieving 64-bit accuracy," ACM Trans. Math.Softw.,vol. 34, no. 4, pp. 1–22, 2008.

[10] J. Sun, G. Peterson, and O. Storaasli, "High performance mixed-precision linear solver for fpgas,"IEEE Trans. on Computers, vol. 57, no. 12, pp. 1614–1623, 2008.

[11] Zachary, K.B. and K.P. Viktor, "Time and area efficient pattern matching on FPGAs", in Proceedings of the 2004 ACM/SIGDA 12th international symposium on Field programmable gate arrays. 2004, ACM Press:Monterey, California, USA.

[12] S. Craven and P. Athanas, "Examining the Viability of FPGA Supercomputing," EURASIP Journal on Embedded Systems, vol. 2007, pp. 1-8, 2007.

[13] P. M. Seidel, G. Even, "Delay-Optimization Implementation of IEEE Floating-Point Addition," IEEE Transactions on computers, pp. 97-113, February 2004, vol. 53, no. 2.

[14] H.-J. Oh et al., "A fully pipelined single-precision floating-point unit in the synergistic processor element of a cell processor," IEEE J. Solid-State Circuits, vol. 41, no. 4,

pp. 759–771, Apr. 2006.

[15] E. Schwarz, M. Schmookler, and S. Trong, "Hardware implementations of denormalized numbers," in Proc. 16th IEEE Symp.Comput. Arithmetic, 2003, pp. 70–78.

[16] Ali Farmani," High Performance Hardware Design Of IEEE Floating Point Adder In FPGA With VHDL", International Journal of Mechatronics, Electrical and Computer Technology, Vol. 3(8), Jul, 2013, pp 81 – 101

[17] Xilinx, http://www.xlinix/com.

[18] L.Louca, T.A.Cook, W.H.Johnson, "Implementation of IEEE Single Precision Floating Point Addition and Multiplication on FPGAs," FPGAs for Custom Computing, 1996.

[19] W.B. Ligon, S.McMillan, G.Monn, F.Stivers, and K.D.Underwood, "A Re-evaluation of the Practicality of Floating-point Operations on FPGAs," IEEE Symp.On Field-Programmable Custom Computing Machines, pp. 206–215, April 1998.