Survey on Decimal Floating Point Multiplier Accumulator Unit Using FMA

[a]Pallavi C M, [b]Divya Prabha, [c]M Z Kurian
[a] M.Tech Student, VLSI & Embedded System, Dept. of ECE, SSIT, Tumkur, India
[b] Associate Professor, Dept. of ECE, SSIT, Tumkur, India, 572105.
[c] HOD, Dept. of ECE, SSIT, Tumkur, India, 572105.

Abstract

Computer arithmetic is convenient to binary representation and it is not natural to humans. The main application of decimal arithmetic is business. In decimal floating point arithmetic, accuracy is the main advantage and there is a growing need of DFA developed its standard IEEE std 754-2008. Decimal floating point can be performed either in software or hardware. The operation with more energy savings increase the speed by using hardware implementation. FMA is the standard operation, Fully parallel decimal floating-point FMA performing the operation (A*B)+C on decimal floating-point operands presents the hardware implementation. Binary and Decimal addition, multiplication and FMA operations can be performed by FMA.

KEYWORDS— Binary floating point, Decimal floating point, Fused Multiply-Add.

I. Introduction

There is a growing need in many commercial applications and database systems the decimal arithmetic witnessed an increased attention in the last decade and binary arithmetic is not sufficient.The inexact mapping between decimal and binary numbers due to inadequacy of binary,for example one tenth is an infinite binary fraction that cannot be represented accurately in a limited precition binary format. To overcome the decimal to binary conversion error software libraries had been developed but hardware binary arithmetic are about 100 to 1000 times faster than software. Decimal floating-point units are expected to be embedded in many processors cores to perform the decimal operation faster than the software packages wit higher accuracy in the near future. IEEE developed its standard for floating-point arithmetic by adding the specification for decimal floating-point arithmetic into IEEE std 754-2008 as the importance and the growing need of decimal arithmetic increases.

The IEEE 754-2008 standard operation has three operands: A, B, C defined one of the operations of Fused Multiply Add (FMA) . (A*B)+C is the result of the FMA , it is equivalent to multiplying A and B then adding the result without rounding to the third operand and performing one final rounding. Accuracy is the main advantage of FMA operation. After the multiplication and addition only one rounding step is performed at the end of the FMA operation instead of performing two rounding. One more advantage of FMA is, it requires single instruction with only one fetch and one decode stages. The addition requires separate fetch and decode cycles for each instruction performing the multiplication.

Using the FMA unit Multiplication and addition can also be performed. The FMA unit can replace the adder and multiplier . Hence, Multiplication can be performed as A * B +zero , and addition can be performed as A* one +c . FMA is used especially in DSP application where the accumulation equation sum=sum+ai*bj appears repeatedly and FMA is vital in many applications.

II. Literature Review

In [1] the intrest in decimal arithmetic increased in great extent . DFP addition , multiplication , FMA , division & square root designed in this paper. IEEE standard for DFP achieved the energy savings by hardware implementation.Energy savings in DFP and hardware implementation of FMA is first to present in this paper. Similar design previously reported is three times faster than Newton-Raphson based divider.

Computers are very natural to binary systems. In a decimal system ten is the natural number base or radix. Only decimal numbers used to perform operations included circuits and data provided by the humans, Even partitioning & memory addressing used decimal numbers only.

There was so many machines related issues & persuasive arguements presented by Buchholz using binary representation instead of decimal in this paper.

Buchholz concludes that “For many diverse applications , high performance tool can be provided by single computers in the combination of decimal and binary arithmetic. For performance goals limited in the interest of economy & computers with a restricted range may be noted that, the conclusion might not be the same.

It’s very small to justify incorporating the difference between binary & decimal operation. For high performance computers the conclusion does appear valid. Hardware implementation is very beneficial using energy delay products in decimal application. Newton-Raphson iterations are the fastest hardware DFP divider & that is first hardware implementation of FMA.

T. Lang and J. D. Bruguera [2] introduced Floating-point multiply-add-fused with reduced latency. The double-precision floating-point multiply-add-fused operation A+(B*C) proposed an architecture for the computation. The anticipation of the normalization step before the addition, combined addition & rounding based on this architecture. Leading-zero-anticipator is not possible to overlap with the adder & normalization is performed before addition.

LZA design can modify to avoid the increase in delay. The normalization can be used to begin the leading bits of its output areee produced first. The load introduced by long connections estimated the delay of the resulting architecture & with respect to previous implementations. They estimate a delay reduction between 15% and 20% .

In many scientific & engineering applications the multiply-add operation is fundamental. Instruction set of DSP processors have been included fixed-point implementation. To do separate multiplication & addition delay should be reduced because it is merged into the adder array of the multiplication. Compare to fixed-point counterpart the floating-point-multiply-add is much more complex.

A unified floating-point multiply-add-fused (MAF) unit of several commercial processors includes as a key feature. There is a single instruction A+(B*C) , executes the double precision with no intermediate rounding.

In [3] R.Samy,H.Fahmy introduced A fully parallel decimal-floating-point fused-multiply-add unit performing operation (A*B)+C on decimal floating-point operands is the first hardware implementation presented in this paper. The proposed design supports the two standard formats decimal64 & decimal128 & it is fully compliant with the IEEE 754-2008 standard. In last decade decimal arithmetic witnessed an increased attention because its growing need in many database system and commercial applications & binary arithmetic is not sufficient. There is inexact mapping between decimal and binary numbers due to inadequacy of binary. Many processors cores to perform the decimaloperation faster than the software packages with high accuracy of decimal-floating-point (DFP) units are expected to be embedded. IEEE std 754-2008 developed its standard for floating-point arithmetic by adding the specification for DFP arithmetic.

Decimal fully parallel floating-point fused-multiply-add are presenting the world's first standard complient decimal single FMA, a simple adder/subtractor, or multiplier is controlled to operate FPGEN cases supplied by IBM as well as random patterns & corner test cases used to verified & tested the functionality of FMA. FPGA and TSMC 90nm technology is used & proposed to synthesize the design.

In [4] A. Akkas and M. Schulte introduced A decimal floating-point fused multiply-add unit with a decimal leading-zero anticipator & in this paper a new decimal LZA presented zero-digit position & reduces the latency of the DFPFMA operation & correctness of the new decimal LZA is fully tested. The delay of the KS-adder used in the decimal FMA unit & the decimal LZA+LZD unit is almost same & the correctness of the new decimal LZA algorithm is fully tested.

L. Dadda [5] introduced Multioperand parallel decimal adder due to the large amount of data in commercial applications. Multi oprand parallel decimal addition with an approach that uses binary arithmetic, suggested by the adoption of binary-coded decimal numbers is consider the problem. In order obtain the binary-to-decimal conversion & it involves corrections for a large number of addendsadopt the latter approach & it is particularly efficient.

D. Lutz [6] introduced Fused multiply-add micro architecture comprising separate early normalizing multiply and add pipelines & FMAs more quickly by splitting them into separate multiplies and adds by perform floating-point unit. The traditional FMA takes less cycles than split FMA. The main advantage of the split implementation is that it does not compromise the performance of any other floating-point operations.

III. Decimal floating point FMA

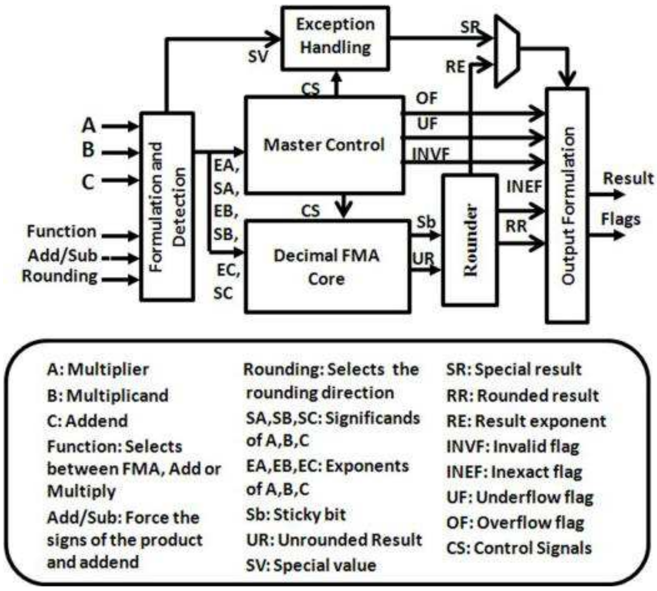

Figure: System overview of Decimal floating-point FMA

There are three operands in proposed decimal floating-point: sign,exponent and significant & also contains two main paths: significant path & exponent path. Significant path to generate the output significant of the result and the exponent path to generate output exponent of the result & the corresponding flags.
The Master control unit calculates the shift amount needed, digit count to be needed,resultant exponent & appropriate output flags.
The output formulation multiplexers between rounded results & special values based on exception conditions. The output is formulated & encoded to match IEEE std 754-2008 format.

The FMA design divided into four stages- Master control, FMA core , special result generation rounding & result encoding. This design supports both 64-bit & 128-bit decimal floating point operands. Here the result encoding stage & the formulation stage removed reduce the FMA size & enhance the speed of operation.

The main function of formulation and detection is operands decoding and special values detection. The operands are reading in DPD format .

FMA core and special result generation contains several units : Special result generation unit is used either quiet Not-a-Number or signalling Not-a-Number. If one operand is qNaN or sNaN, its payload value is bypassed to the result encoding unit .

Decimal FMA core unit receives the decoded multiplier,addend & multiplicand. In this unit addend C is shifted either to the left or right to align.

The Master control unit calculates Right/Left amount for the addend & with the intermediate product the addend is shifted to be aligned. Right shift is needed if the addend exponent is similar than the sum of the multiplier & multiplicand exponents.

- The advantages are: (1) Accuracy is the main advantage of FMA (2) Instead of performing two rounding steps only one rounding step is performed at the end of the FMA operation and (3) FMA is single instruction with only one fetch and one one decode stages, so usually lesser in delay and shorter in area.

V. Conclusion

In this paper, An Efficient Implementation of Decimal Floating Point Multiplier Accumulator Unit presents a new IEEE compliant combined binary/decimal FMA. This FMA can also work as an adder or multiplier, hence it can replace both the binary and decimal floating point units.Till now Literature survey has been done & Implementation of this project will be done in later phases.

References

[1] H. A. H. Fahmy, R. Raafat, A. M. Abdel-Majeed, R. Samy, T. El-Deeb, and Y. Farouk, “Energy and delay improvement via decimal floating point units,” in Proceedings of the 19th IEEE Symposium on Computer Arithmetic, Portland, Oregon, USA, pp. 221-224, June 2009

[2] T. Lang and J. D. Bruguera, “Floating-point multiply-add-fused with reduced latency,” IEEE Transactions on Computers, vol. 53, pp. 988–1003, Aug. 2004

[3] R. Samy, H. Fahmy, R. Raafat, A. Mohamed, T. ElDeeb, and Y. Farouk, "A decimal floating-point fused-multiply-add unit," in 53rd IEEE International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 529–532, Aug. 2010.

[4] A. Akkas and M. Schulte, "A decimal floating-point fused multiply-add unit with a novel decimal leading-zero anticipator," in IEEE International Conference on Application-Specific System, Architectures and Processors (ASAP) , pp. 43-50

[5] L Dadda, "Multioperand parallel decimal adder: A mixed binary and BCD ap[proach," IEEE Transactions on Computers, vol. 56, no. 10, pp. 1320-1328,2007.

[6] D. Lutz, "Fused multiply-add micro architecture comprising separate early-normalizing multiply and add pipelines," in computer Arithmetic (ARITH) , 2011 20TH IEEE Symposium on, pp. 123-128, july 2011