ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

DESIGN AND ANALYSIS OF INEXACT FLOATING-POINT ADDERS

DHANAVATH KIRAN KUMAR NAIK
Executive Engineer, WAPCOS limited(Govt.of India)
kiran00418@gmail.com

Abstract: Floating-point applications are a growing trend in the FPGA community. Power has become a major constraint in nanoscale integrated circuit design due to the increasing demands for mobile computing and higher integration density. Low-power is an imperative requirement for portable multimedia devices employing various signal processing algorithms and architectures. As an emerging computational paradigm, an inexact circuit reduces both dynamic and static power dissipation for error-tolerant applications. In this paper, an inexact floating-point adder is proposed by approximately designing an exponent subtractor and mantissa adder. Related operations such as normalization and rounding are also dealt with in terms of inexact computing. High dynamic range images are then processed using the proposed inexact floating-point.

Keywords—*Inexact circuits, floating-point adders, low power, error analysis, high dynamic range image*

I.INTRODUCTION

With progression and advancement of creative computerized coordinated circuits, control utilization has drastically expanded; control has turned into a key plan requirement because of the appeal for versatile registering and higher combination thickness. Conventional plans apply completely exact figuring to a wide range of uses; nonetheless, mistake tolerant applications including human mediation, (for example, picture handling) don't require full exactness. Thus, it is conceivable to perform calculation with inaccurate circuits; in these cases, vague processing is an alluring way to deal with spare power and territory, while accomplishing enhanced execution contrasted with exact outlines.

The number juggling unit is the center of a processor, and its energy to a great extent decides the energy of the entire processor. Late research on vague settled point adders has demonstrated that inaccurate preparing equipment with a relative mistake of 7.58 percent can be about 15 times more proficient as far as speed, zone and vitality item than an exact chip. Estimated chips are littler, speedier and expend less vitality. Albeit settled point number-crunching circuits have been examined as far as inaccurate registering, skimming point (FP) number-crunching circuits are altogether more power hungry and they have not been completely considered for estimated figuring. The FP arrange offers a high unique range for computationally escalated applications; FP adders and multipliers are ordinarily utilized as a part of DSP frameworks However, its application to implanted DSP frameworks is constrained because of the powerful utilization.

A low power outline of a FP multiplier was researched by Tong et al.; this plan includes the truncation of equipment and a decrease of the bit width portrayal of the FP information. A probabilistic FP multiplier was proposed by Gupta et al. generally as a vitality proficient outline. A lightweight FP configuration stream utilizing bit-width enhancement was proposed for low power flag preparing applications. Low exactness FP numbers have likewise been utilized for MP3 disentangling to decrease memory usage and power utilization. Notwithstanding, to the best of the creators' learning, there has been no examination to date on an estimated FP viper plan. In this paper, viper plans are examined as a beginning stage for estimated FP number juggling; a few vague adder outlines are proposed and surveyed for application to high unique range pictures. The upper bound blunder because of the estimated configuration is broke down for the normal case to control the outline of inaccurate FP adders. A subjective visual distinction indicator metric is utilized to quantify the aftereffects of

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

picture expansion; in addition, a method is presented for planning inaccurate FP number juggling circuits.

Fixed Point and Floating Point Representations

Each genuine number has a whole number part and a portion section; a radix point is utilized to separate between them. The quantity of paired digits alloted to the whole number part might be diverse to the quantity of digits doled out to the fragmentary part. A nonexclusive paired portrayal with decimal change is appeared in Figure 1.

	Integer Part				Binary Point	Fraction Part			
Binary	...	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}	2^{-3}	...
Decimal		8	4	2	1		½	¼	⅛	...

Fig. 1: Binary representation and conversion to decimal of a numeric

Basic Format

There are two fundamental arrangements portrayed in IEEE 754 configuration, twofold exactness utilizing 64-bits and single-accuracy utilizing 32-bits. Table 1 demonstrates the examination between the essential parts of the two portrayals.

Table 1

Single and double precision format summary

Format	Precision (p)	E_{max}	E_{min}	Exponent bias	Exponent width	Format width
Single	24	+127	-126	127	8	32
Double	53	+1023	-1022	1023	11	64

To assess diverse adder calculations, we are just keen on single accuracy arrange. Single-accuracy organize utilizes 1-bit for sign piece, 8-bits for example and 23-bits to speak to the part as appeared in Figure 2.

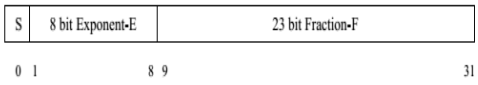

Fig. 2: IEEE 754 single precision format

The single- precision floating-point number is calculated as (-1)S× 1.F × 2(E-127). The sign bit is either 0 for non-negative number or 1 for negative numbers. The exponent field represents both positive and negative exponents. To do this, a bias is added to the actual exponent. For IEEE single-precision format, this value is 127, for example, a stored value of 200 indicates an exponent of (200-127), or 73. The mantissa or significand is composed of an implicit leading bit and the fraction bits, and represents the precision bits of the number. Exponent values (hexadecimal)of 0xFF and 0x00 are reserved to encode special numbers such as zero, de normalized numbers, infinity, and NaNs. The mapping from an encoding of a single-precision floating-point number to the number's value is summarized in Table 2.

Table 2

IEEE 754 single precision floating-point encoding

Sign	Exponent	Fraction	Value	Description
S	0xFF	0x00000000	$(-1)^S \infty$	Infinity
S	0xFF	F≠0	*NaN*	Not a Number
S	0x00	0x00000000	0	Zero
S	0x00	F≠0	$(-1)^S \times 0.F \times 2^{(E-126)}$	Denormalized Number
S	0x00 < E < 0xFF	F	$(-1)^S \times 1.F \times 2^{(E-127)}$	Normalized Number

Standard Floating Point Addition Algorithm

This section will review the standard floating point algorithm architecture, and the hardware modules designed as part of this algorithm, including their function, structure, and use. The standard architecture is the baseline algorithm for floating-point addition in any kind of hardware and software design.

II.LITERATURE SURVEY

In spite of the fact that the vague outline of settled point adders has been widely examined, little research has been led on inaccurate gliding point

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

math plan. A low power outline of a skimming point multiplier was researched by Tong et al. which include truncating equipment; the adjusting unit was found to require half of the equipment of a correct gliding point multiplier. In this way, the adjusting unit is a contender for expulsion to spare power, like an inaccurate plan. A probabilistic gliding point multiplier was proposed by Gupta et al. as a vitality proficient outline. Be that as it may, to the best of the creators' learning, there has been no examination to date on a vague drifting point adder outline, which has a more intricate structure than a skimming point multiplier. An inalienable issue of paired drifting point math utilized as a part of money related figurings is that most decimal coasting point numbers can't be spoken to precisely in double skimming point arrangements, and blunders that are not worthy may happen over the span of the calculation. Decimal gliding point number juggling tends to this issue, however a debasement in execution will happen contrasted with parallel coasting point operations actualized in equipment. In spite of its execution hindrance, decimal drifting point number-crunching is required by specific applications that need comes about indistinguishable to those ascertained by hand. This is valid for money change, saving money, charging, and other budgetary applications. [3] proposed a system which can achieve a higher throughput and higher energy efficiency. The S-BOX is designed by using Advanced Encryption Standard (AES). The AES is a symmetric key standard for encryption and decryption of blocks of data. In encryption, the AES accepts a plaintext input, which is limited to 128 bits, and a key that can be specified to be 128 bits to generate the Cipher text. In decryption, the cipher text is converted to original one. By using this AES technique the original text is highly secured and the information is not broken by the intruder. From that, the design of S-BOX is used to protect the message and also achieve a high throughput, high energy efficiency and occupy less area.

In some cases, these prerequisites are ordered by law; different circumstances, they are important to keep away from vast bookkeeping errors. In view of the significance of this issue various decimal arrangements exist, both equipment and programming. Programming arrangements incorporate C#, COBOL, and XML, which give decimal operations and information sorts. Likewise, Java and C/C++ both have bundles, called Big Decimal and dec Number, individually. Equipment arrangements were more conspicuous prior in the PC age with the ENIAC and UNIVAC. Be that as it may, later illustrations incorporate the CADAC, IBM's z900 and z9 models, and various other proposed equipment usage. More equipment illustrations can be found, and a more top to bottom discourse is found in Wang's work.

Design tradeoff analysis of floating-point adder in FPGAs

Field Programmable Gate Arrays (FPGA) are progressively being utilized to outline top of the line computationally serious microchips equipped for taking care of both settled and skimming point numerical operations. Expansion is the most complex operation in a coasting point unit and offers real postponement while taking huge zone. Throughout the years, the VLSI people group has created many skimming point viper calculations basically planned to decrease the general inertness. A proficient outline of coasting point adder onto a FPGA offers real region and execution overheads. With the current headway in FPGA design and region thickness, idleness has been the principle center of consideration so as to enhance execution. Our examination was situated towards contemplating and actualizing standard, Leading One Predictor (LOP), and far and close information way gliding point expansion calculations. Every calculation has complex sub-operations which lead altogether to general inertness of the outline. Each of the sub-operation is examined for various usage and afterward incorporated onto a Xilinx Virtex2p FPGA gadget to be decided for best execution. This postulation examines in detail the most ideal FPGA usage for all the three calculations and will go about as an imperative outline asset.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

The execution standard is inactivity in every one of the cases. The calculations are looked at for general inertness, zone, and levels of rationale and broke down particularly for Virtex2p design, one of the most recent FPGA structures gave by Xilinx. As indicated by our outcomes standard calculation is the best execution as for region however has general huge inertness of 27.059 ns while involving 541 cuts. Trim calculation enhances inertness by 6.5% on included cost of 38% zone contrasted with standard calculation. Far and close information way usage demonstrates 19% change in idleness on included cost of 88% in range contrasted with standard calculation. The outcomes obviously demonstrate that for territory proficient plan standard calculation is the best decision yet for outlines where idleness is the criteria of execution far and close information way is the best option. The standard and LOP calculations were pipelined into five phases and contrasted and the Xilinx Intellectual Property. The pipelined LOP gives 22% better clock speed on an additional cost of 15% region when contrasted with Xilinx Intellectual Property and accordingly a superior decision for higher throughput applications.

III.PROPOSED SYSTEM

Design Of inexact floating-Point adders:

The inexact design of an FP adder originates at an architectural level. It consists of designing both the mantissa adder and exponent subtractor by using approximate fixed-point adders. At the same time, related logic including the normalizer and the rounder should also be considered according to the inexact mantissa and exponent parts. The circuit level inexact designs are discussed in detail in the following sections.

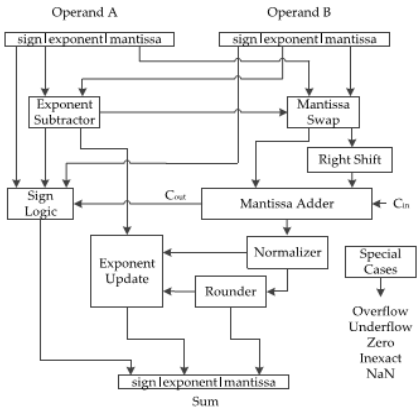

Fig. 3: the accurate FP adder architecture

Exponent Subtractor

The type subtractor is utilized for example correlation and can be actualized as a viper. An inaccurate settled point adder has been widely contemplated and can be utilized as a part of the type viper; vague adders, for example, bring down part-OR adders (LOA), inexact mirror adders, rough XOR/XNOR-based adders, and equivalent division adders can be found in the writing. For a quick FP adder, an updated LOA viper is utilized, on the grounds that it altogether decreases the basic way by disregarding the lower convey bits.

A k-bit LOA comprises of two sections, i.e., a m-bit correct viper and a n-bit vague adder. The m-bit adder is utilized for the m most huge bits of the whole, while then-piece viper comprises of OR doors to register the expansion of the slightest huge n bits (i.e., the lower n-bit viper is a variety of n two-info OR entryways). In the first LOA outline, an extra AND door is utilized for producing the most huge convey bit of then-piece adder; in this work, all convey bits in then-piece vague viper are overlooked to additionally diminish the basic way. The type is prevailing in the FP arrange, on the grounds that it decides the dynamic range. The rough outline of the example subtractor must be painstakingly thought to be because of its significance in the number arrangement. The aftereffects of the expansion are altogether influenced by applying a rough outline to

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

just a couple of the minimum critical bits of the type subtractor under a little information run.

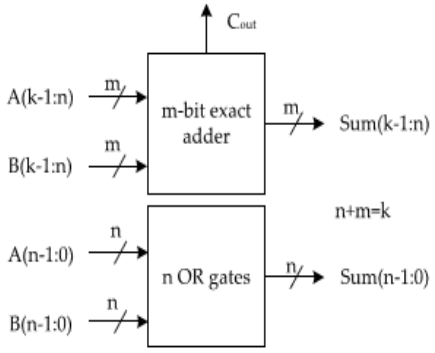

Fig. 4:The revised LOA adder structure.

Mantissa Adder

The overhauled LOA viper can likewise be utilized as a part of the mantissa adder for an estimated outline. Contrasted with a type subtractor, the mantissa adder offers a bigger outline space for estimated plan, in light of the fact that the quantity of bits in the mantissa viper is essentially bigger than the example subtractor. As appeared in Table 1, the quantity of mantissa bits is bigger than the quantity of type bits. For the IEEE single exactness arrange, the type subtractor is a 8-bit viper, while the mantissa adder is a 25-bit viper (for two 24-bit significances). Besides, the estimated plan in the mantissa adder has a lower affect on the blunder than its example partner in the lower information go, on the grounds that the mantissa part is less noteworthy than the type part. In this manner, an inaccurate plan of a mantissa viper is more fitting. An itemized investigation of blunders presented by each part is additionally talked about in the following area.

Normalizer:

Standardization is required to guarantee that the expansion comes about fall in the right range; the total or contrast might be too little and a multi-bit left move process might be required. A decrease of the example is additionally fundamental. The standardization is performed by a main zeros counter that decides the required number of left moves. As the mantissa viper is now not correct for then minimum huge bits, the location of the main zeros can likewise be improved in the vague outline, i.e., surmised driving zero including rationale can be utilized.

Rounder

An adjusting mode is required to suit the inaccurate number that a FP organization can speak to. An appropriate adjusting keeps up three additional bits (i.e., protect bit, round piece and sticky piece). The adder may require a further standardization and example modification after the adjusting step, along these lines the equipment for adjusting is critical. In any case, it doesn't influence the consequences of the estimated expansion as the lower noteworthy n bits are now inaccurate. In this manner, adjusting can be overlooked in the inaccurate outline of a FP adder.

Overall Inexact FP Adder Architecture

Based on the previous discussion, an inexact FP adder can be designed by using approximate adders in the exponent subtractor and mantissa adders, an approximate leading zero counter in the normalizer and by ignoring the rounder. The inexact FP adder architecture is shown in Fig. 5.

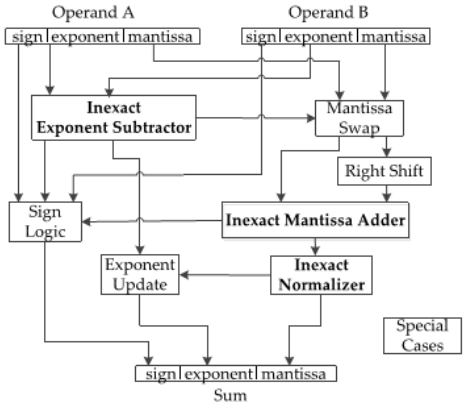

Fig. 5: The inexact FP adder architecture.

IV.ERROR ANALYSIS

The errors due to an inexact design must be carefully considered because the results can be significantly affected by the inexact design. To understand the effects of each inexact part, an upper bound error analysis is carried out in this section for the average case. The errors in the inexact design are

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

mostly from the exponent and mantissa parts. The error contribution from related logic (including normalization) can be included in either the exponent, or the mantissa adder for the upper bound analysis of the average case; therefore, only errors from the inexact exponent adder (IEA) and the inexact mantissa adder (IMA) are considered. The errors are closely related to the number of inexact bits in both adders. Initially, the errors from IEA and IMA are considered separately. Then, the error relationship between the two parts is studied.

Consider a general precision FP format. The numbers of exponent and mantissa bits are given by E and M, respectively, where E>=1, M>=1. The error distance (ED) [16] is given as ED=|S~-S|, where S~ and S are the sum of the approximate and accurate adders, respectively.

Errors from Inexact Mantissa Adder

To analyze the errors from IMA, a floating point adder architecture using an inexact mantissa adder and an exact exponent adder is considered. The errors in the IMA are derived as follows. Assume m is the number of inexact bits in the IMA. The maximum error introduced by the ith bit is $2^{-(M-i+1)}$, where $i = 1, 2, \ldots, m, m < M.$. The local error distance introduced by the m inexact bits in the IMA is at most:

$$\begin{aligned} \mathrm{ED_{MA}} &= 2^{-M} + 2^{-(M-1)} + \cdots + 2^{-(M-m+1)} \\ &= 2^{m-M} - 2^{-M} \approx 2^{m-M}, m = 1, 2, \ldots, M. \end{aligned} \tag{1}$$

Assume that the input data is uniformly distributed in the full range of the FP format. The upper bound of the total (global) error distance introduced by IMA takes into account the exact exponent part and is given by:

$$\begin{aligned} \mathrm{ED_{IMA}} &= \frac{2^{-(2^{E-1}-2)} + 2^{-(2^{E-1}-3)} + \cdots + 2^{(2^{E-1}-2)} + 2^{(2^{E-1}-1)}}{2^E - 2} \times \mathrm{ED_{MA}} \\ &\approx \frac{2^{(2^{E-1}-1)} - 2^{-(2^{E-1}-1)}}{2^{E-1} - 1} \times 2^{m-M} \approx 2^{2^{E-1}-E+m-M} \end{aligned} \tag{2}$$

Errors from Inexact Exponent Adder

For the errors from IEA, a floating point architecture using inexact exponent and exact mantissa adders is considered. The exponent part has a bias of $(2^{E-1}-1)$ the IEEE-754 standard. Therefore, the real exponent can be obtained by subtracting the bias from the exponent part of an operand. The error caused by the jth bit in IEA is 2j, and the total error distance of an IEA with n inexact bits is at most:

$$\begin{aligned} ED_{IEA} &= 2^{(2^0+2^1+\cdots+2^{n-1})} \\ &= 2^{(2^n-1)}, n = 1, 2, \ldots, E. \end{aligned} \tag{3}$$

Error Relationship between IMA and IEA

As the data range of a floating-point format is significantly larger than a fixed-point format, a relative error distance (RED) is defined next; RED is used to facilitate the analysis of the relationship between errors of IMA and IEA. It is given as $RED=\log_2(ED)$, where RED stands for the relative error distance. Therefore, the REDs of IMA and IEA are:

$$RED_{IMA} = 2^{E-1} - E + m - |M, \tag{4}$$

$$RED_{IEA} = 2^n - 1. \tag{5}$$

The exponent part of the floating-point format can be negative, hence RED can also be negative. The data range largely determines the design strategy of an inexact floating-point adder. The relative error distance due to IMA and IEA using different numbers of inexact bits in the IEEE FP types (i.e., half, precision and double precision) are shown in Fig. 6; the errors from IEA (IMA) increase exponentially (linearly). Although the exponent part is more dominant in the FP format, the error is nearly the same independently of the number of inexact bits in IMA for larger precision formats such as double precision. So, the errors from IMA are significantly larger than from IEA when the number of inexact bits is smaller than E-2. This is due to the high dynamic range that a larger precision formats offer. Even one inexact bit (for example, m= 1) in the mantissa adder causes more errors than that from its exponent counterpart (i.e., when n=1). However, for smaller precision formats (such as half precision), the errors from IEA and IMA are comparable; so, it is better to apply more inexact bits in IMA, because IMA provides a larger design space with similar effects due to errors.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

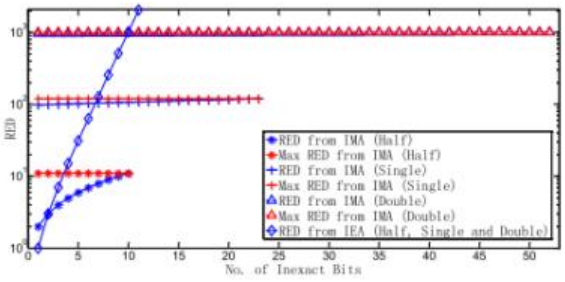

Fig. 6: Relative errors introduced by both IMA and IEA with different numbers of inexact bits in three IEEE basic FP types: half precision (E= 5, M= 10), single precision (E= 8, M= 23) and double precision (E= 11, M= 52).

In many real DSP applications (such as voice recognition and image processing), the data range is usually limited to 2^4~2^{10}; for example, the dynamic range of luminance that a human can perceive is up to 10^5. Therefore for different applications, the errors introduced by IMA are very different. For a data range DR_i,

i.e., $2^{-(2^{E-i}-2)} \sim 2^{(2^{E-i}-1)}, 1 \le i \le E-1$, the errors from IMA are:

$$RED_{IMA}(i) = (2^{E-1} - E + m - M) - 2^{E-1}(1 - 2^{1-i}). \quad (6)$$

A detailed evaluation is shown in Fig. 7 by using the IEEE single precision format for establishing the relationship of the IMA and IEA errors under various data ranges. In the reduced data range, the errors due to IMA are less than from IEA. When the data range is 2^{-30}~2^{31}, then the errors introduced by one inexact bit in both adders are the same. For a smaller data range, i.e., 2^{-14}~2^{16} (i.e., the dynamic range widely used in multimedia), the errors from IMA are significantly smaller. So, even when all bits in the mantissa adders are inexact (i.e., m=23), the errors from the mantissa parts are similar to the IEA with three inexact bits (i.e., n=3). For a data range smaller than 2^{-6}~2^7, when all bits in the mantissa adder are inexact, then the errors from the mantissa part are still less than for an IEA with only one inexact bit. The upper bound error analysis for the average case can be used to provide a guideline for the inexact design of floating point adders. Generally, for a larger data range, a larger precision format should be used and more inexact bits can be applied in the exponent part. For a smaller data range, a smaller precision format should be used, so more inexact bits can also be used in the mantissa adder. For some applications, it is acceptable to have all inexact bits in the mantissa adder; hence, a design strategy is strongly dependent on the application data range.

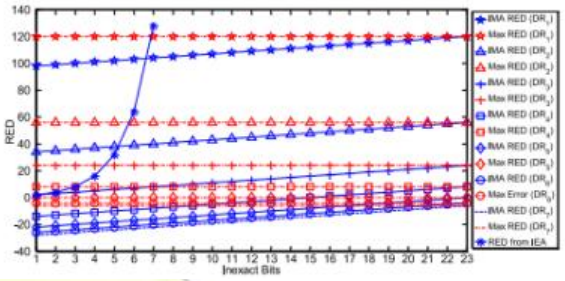

Fig. 7: Relative errors introduced by IMA and IEA with different inexact bits in various data ranges (IEEE single precision: E= 8, M= 23).

V.INEXACT DESIGN METHODOLOGY

The desired inexact design can be achieved through this iterative process. The design can start from an initial inexact adder with a small number of inexact bits (as determined by the error analysis discussed in Section 4). The accuracy can then be improved by using more exact bits in the mantissa and exponent adders until the accuracy requirements for an application are met. An inexact FP adder design can be modeled using a hardware description language such as VHDL or Verilog. The results are then analyzed to assess whether they meet the desired metrics; the accuracy can be adjusted by increasing the number of exact bits in the adders and accordingly changing the related logic. The design procedure for inexact FP adders is shown in Fig. 8.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Issue 3, March 2017

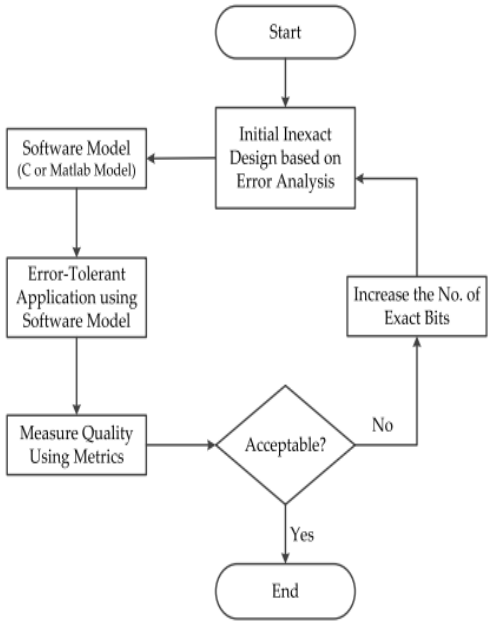

Fig. 8: A design procedure for the inexact design of FP adders

VII.CONCLUSION

Inexact FP adder designs have been investigated in this paper. Approximate designs of the mantissa and exponent adders have been proposed and consideration has been given to normalization and rounding. The mistake connection between the mantissa and type parts has been contemplated for the normal case; this is an essential component in directing an estimated FP number juggling outline. Two extraordinary cases for the inaccurate outline of FP adders have been contemplated. The principal configuration utilizes an all-piece inaccurate mantissa viper; the second outline utilizes a vague LSB in the type subtraction. The two plans have been connected to high unique range pictures and the outcomes have demonstrated that both vague FP adders are low power outlines. These plans require a little territory and offer higher execution than their proportional correct outlines. All things considered they are appropriate for high powerful picture applications. It has been demonstrated that the type part is a predominant part in the FP number arrangement; in any case it has a littler plan space for a vague outline contrasted with the mantissa viper.

REFERENCES

[1] K. Palem and A. Lingamneni, "Ten years of building broken chips: The physics and engineering of inexact computing,"ACM Trans. Embedded Comput. Syst., vol. 12, no. 2, article 87, 2013.

[2] A. Lingamneni, K. Muntimadugu, C. Enz, R. Karp, K. Palem, and C. Piguet, "Algorithmic methodologies for ultra-efficient inexact architectures for sustaining technology scaling," inProc. ACM Int. Conf. Comput. Frontiers, 2012, pp. 3–12.

[3] Christo Ananth, H. Anusuya Baby, "S-Box using AES Technique", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 3, March – 2014, pp 285-290

[4] V. Gupta, D. Mohapatra, S. Park, A. Raghunathan, and K. Roy, "IMPACT: IMPrecise adders for low-power approximate computing," in Proc. Int. Symp. Low Power Electron. Des., 2011, pp. 1–3.

DHANAVATH KIRAN KUMAR NAIK received his Bachelor's degrees in Electronics and communication from Nagarjuna Institute of Technology and Sciences. He received his Master's degree in VLSI system design from Swami Ramananda Tirtha Institute of Science and Technology. He is currently working as a executive engineer in WAPCOS limited(Govt.of India).His current research interests include very large scale integration (VLSI) low power design, test automation and fault-tolerant computing.