K.Raja Kameswari et al. International Journal of Recent Research Aspects ISSN: 2349-7688, Vol. 4, Issue 4, Dec 2017, pp. 259-263

Implementation of Efficient Architecture for Dual-Mode Floating Point Division

K.Raja Kameswari [1], Dhanbal.R[2]
[1] M.Tech VLSI, Vellore Institute of Technology, Vellore.
[2]Vellore Institute of Technology, School of Electronics Engineering, Vellore

Abstract: - This paper proposed an architecture for the double precision floating point division it works for both single and dual mode. Floating point division is used in various applications like engineering and scientific applications. This proposed architecture works for dual mode which can calculate on two sets of single precision operands and one set of double precision operands in parallel. This architecture works on the principle of series expansion multiplicative methodology for mantissa computation. In this a Radix- 4 Modified Booth Multiplier is used, which is used in dual mode mantissa computation in iterative fashion and also in floating point division the key elements used are leading zero counter, dynamic left and right shifters , rounding etc. The dual mode architecture which is proposed in this paper is synthesized in Synopsys .Compared to the other double precision division architectures. The proposed architecture performs better in factors like area, time period and throughput.

Index terms- Floating point Division, Dual –Mode functionality, Modified Booth Algorithm.

I. INTRODUCTION

Floating point arithmetic is mostly used in many scientific, engineering and signal processing applications its large vital range and suitable platform for designers to register their algorithms. Moreover executing floating point arithmetic numbers is very difficult. The arithmetic operations like addition, subtraction, multiplication and division. Among these division is generally more challenging but division is mostly used in scientific and engineering applications. So, it is the need to implement efficient architecture for floating point. Generally dividers are designed to use in the iteration and latency is more compared to other arithmetic operations. Floating point division [1] is the arithmetic core used in scientific and engineering applications. The hardware complexity of the division in floating point is high compared to other operations like addition, subtraction and multiplication and also division architecture require more area and compared to be low performance. So, based on the area parameter we aim to unified and configurable architecture is proposed. In this paper we going to propose an architecture which computes both single and double precision vectors used for vector processing as an alternative different vectors of single and double precision, it can form a configurable floating point block, each configurable floating point arithmetic blocks can be used as double precision or two parallel single precision. Because of this block we can improve area and gives the best performance. The proposed architecture is based on multiplicative division algorithm series expansion methodology [2]algorithm .this one of the efficient methods in multiplicative division method like Goldschmidt and Newton raphson methods[10]. The implementation may do in different methods like digital recurrence, multiplication based and other techniques. Coming to the digital recurrence algorithm SRT is the iterative form it takes more implementation time and coming to multiplication based upon approximation and inverse of division in iteration fashion. series expansion algorithm is based on computing initial quotient and then computing the quotient by using remainder and add the quotients .this expansion method will give the best performance, less area, lower latency and also compared to above methods this method require less memory requirement and faster compared to the digital recurrence method. In this architecture dual mode, Modified booth multiplier Radix 4 in the use of mantissa division, which has less area and performance is high over single-mode multiplier. since the underlying integer multiplier in mantissa division unit .an architecture iterative proposed by using 1-stage integer multiplier to get area efficiency. This paper proposed an architecture for the division which can be either a parallel two single precision and double precision and this architecture is designed based on the series expansion method .this architecture DpdSP division architecture supports subnormal and normal operands with round-to-nearest rounding method .a design only normal support has also implemented and compared with proposed architecture implemented designs like infinity,divide-by-zero,zero.proposed architecture is designed for both normal and subnormal which is helpful for rounding with round-to-nearest rounding for both single and double precision results. All the major blocks like dynamic right and left shifters, leading one detection and mantissa division are designed for efficient purpose .a single mode double precision based on the computational flow. A Floating point arithmetic computation involves separating the sign, exponent and mantissa part of the operands and finally combine them after normalization and rounding .the IEEE

standard format for floating point numbers[6] in the proposed architecture are single and double precision numbers are as follows

$$SP: \overbrace{1-bit}^{\text{Sign}}\ \overbrace{8-bit}^{\text{Exponent}}\ \overbrace{23-bit}^{\text{Mantissa}}$$

$$DP: \overbrace{1-bit}^{\text{Sign}}\ \overbrace{11-bit}^{\text{Exponent}}\ \overbrace{52-bit}^{\text{Mantissa}}$$

II.BACKGROUND

The algorithm for floating point division architecture is shown below

1: $(IN1$ *(Dividend)*, $IN2$ *(Divisor)*) Input Operands;
2: **Data Extraction & Exceptional Check-up:**
$\{S1(\text{Sign1}), E1(\text{Exponent1}), M1(\text{Mantissa1})\} \leftarrow IN1$
$\{S2, E2, M2\} \leftarrow IN2$
Check for Infinity, Sub-Normal, Zero, Divide-By-Zero
3: **Process both Mantissa for Sub-Normal:**
Leading One Detection of both Mantissa ($\rightarrow$ L_Shift1, L_Shift2)
Dynamic Left Shifting of both Mantissa
4: **Sign, Exponent & Right-Shift-Amount Computation:**
$S \leftarrow S1 \oplus S2$
$E \leftarrow (E1 - L\_Shift1) - (E2 - L\_Shift2) + BIAS$
$R\_Shift\_Amount \leftarrow (E2 - L\_Shift2) - (E1 - L\_Shift1) - BIAS$
5: **Mantissa Computation:** $M \leftarrow M1/M2$
6: **Dynamic Right Shifting of Quotient Mantissa**
7: **Normalization & Rounding:**
Determine Correct Rounding Position
Compute ULP using Guard, Round & Sticky Bit
Compute $M \leftarrow M + ULP$
1-bit Right Shift Mantissa in Case of Mantissa Overflow
Update Exponent
8: **Finalizing Output:**
Determine STATUS signal & Resolve Exceptional Cases
Determine Final Output

This algorithm is helpful to implement for both subnormal and normal processing .algorithm includes case handling and processing. Floating point arithmetic implementation computing the sign, exponent and mantissa parts of the operands and combined them after normalization and rounding.

III. DPDSP DIVISION ARCHITECTURE

The proposed architecture is shown in Fig1.it consists of three pipelining stages. In the architecture each stage is discussed below. coming to the operands there are two 64 –bit operands ,one dividend(in1) and divisor (in2)are the initial inputs along with a control signal is available dp_sp(dual precision or dual single precision).both the operands consists of two parallel SP operands (two sets of 32-bit)and DP operands(64-bit) is displayed in Fig2. Algorithm is divided into two stages. Coming to first stage of architecture is contains stage 2 and stage 3 of algorithm. It includes data extraction, sub normal processing and exceptional case handling[3] and also includes mantissa division.

The data extraction computation is shown in Fig 3 .primary operands and extract the signs (sp1_s1,sp1_s2,sp2_s1,sp2_s2,dp_s1,dp_s2) and exponents (sp1_e1,sp1_e2,sp2_e1,sp2_e2,dp_e1and dp_e2)and mantissa(sp1_m1,sp_m2,sp2_m1,sp2_m2,dp_e1anddp_e2). sizes of the above components is shown in Fig2

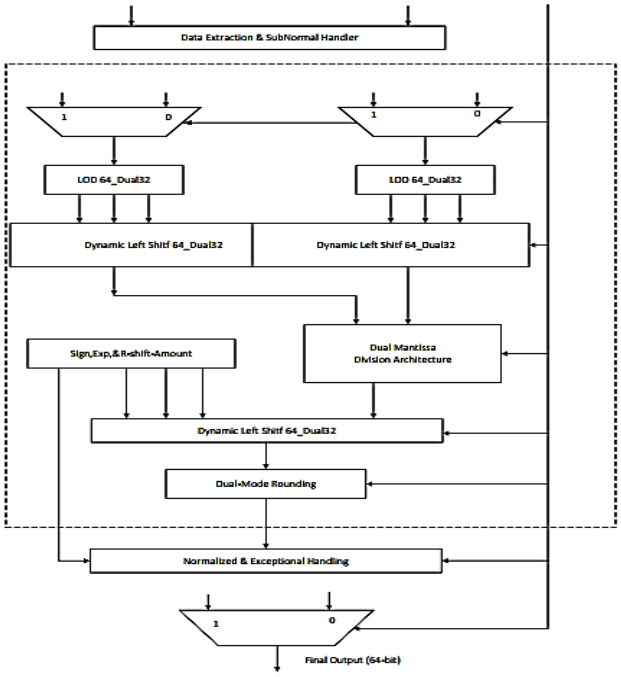

Fig1:Double precision with dual sinle precision Division architecture

s:sign,_e:exponent,_m:mantissa,,dp_sp:mode(DP/Dual-sp)dp:double precsion,sp:single precision,_ls:leftshift,_rs:right shift.

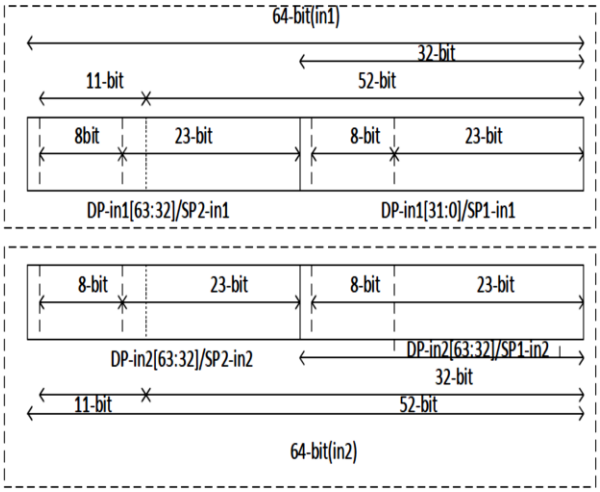

Fig2:Dpdsp Input Output Format

The input and output encoding is taken under the format of IEEE standard format.the subnormal handling and exceptional check computation checks subnormal,NaN(Not a Number)and infinity.it also checks for division by zero and have to shared among both SP and DP.After data extraction and sub normal and exceptional handling.uni mantissa (M1and M2) is obtained by using mux as shown in architecture.beacause of these uni mantissa is helpful in further blocks which is helpful in efficient resource sharing. The next two blocks are leading one dectector and dynamic shifter helps to convert the sub normal format subnormal format .firstly it provide the amount of left shift amount by using the block leading zero counter and the amount given by the LOD and then dynamic left shifter shifts the mantissa.the architecture of the LOD is designed in a fashion that firstly it starts with basic block of 2:1 LOD and finally 64:6 LOD which has 32:5 LOD .inital bloch consists basic gates like AND,OR and NOT gates which makes for both sp's and dp's.the final result is costs is minimal compared to DP LOD. The uni mantissa are given to the Dynamic left shifter dual mode .the shifter shifts the amount of left shifting given by the LOD(as in Fig 1).in the processing of DP,the SPs left shifts amounts are kept to be zero same wise during SPs DP left shift amount are kept to zero.the left shifter dual mode is shown in [8].this shifter consists of 6 stage barrel shifter in which 5 are dual mode ansd first shifter is single mode.in th stage one it only used to preform for double precision coming to dual mode stage consiss of multiplexers ,which shifts the bits correspnding both SP and DPs The main block in the DPdSP division architecture is Mantissa division architecture for dual mode after shifting the mantissa is converted into normalised form m1,m2.the normailsed inputs are given to two LUT tables avaliable in mantissa architecture .in these Look Up tables[7] one is used for both dp and sp2 and other look up tabel is used for sp1 only is shown in Fig.3.

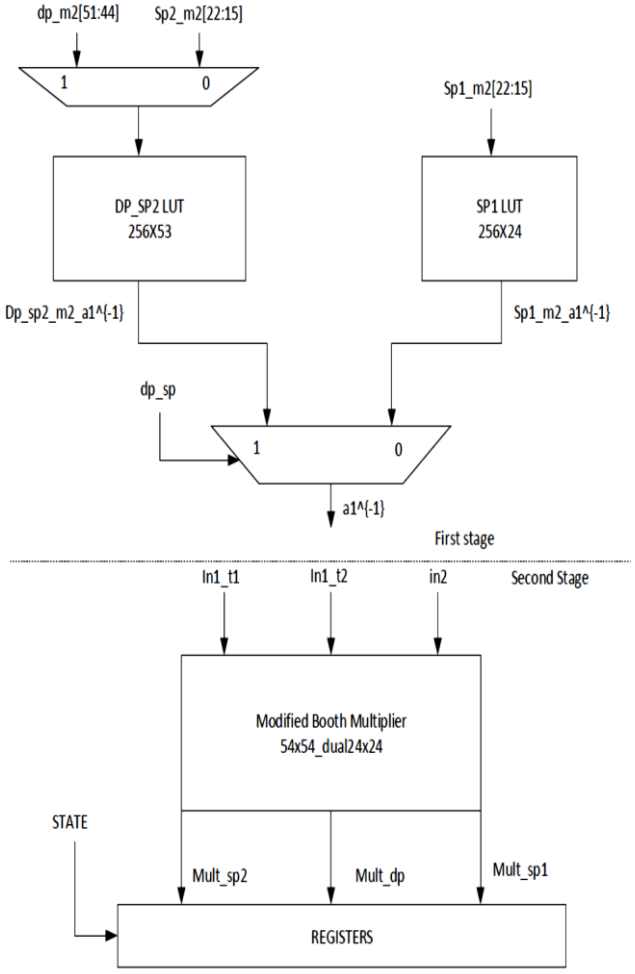

Fig.3.DpdSP Dualmode Mantissa architecture for division

In the second pipelining stage(4 &5 stages) it consists of sign, exponent and right shift amount and result is helpful for the remaining stages .the computed sign bits is simple XOR operation .mantissa generating is very difficult part in division arithmetic. Mantissa obtained from the first stage of mantissa division architecture is given to the modified booth multiplier. For the effective inputs to the multiplier a Dual mode FSM is used. The booth multiplier is shown in the below figure which is used to compute in dual mode, booth multiplier is helpful to reduce partial products and for DP processing Here, it is a 54-bit integer multiplier and DADDA tree is having 8 levels which compress entire partial points into two operandsand futher is given to parallel prefix adder .the final product contains both dual SP and DP results in Fig.4.

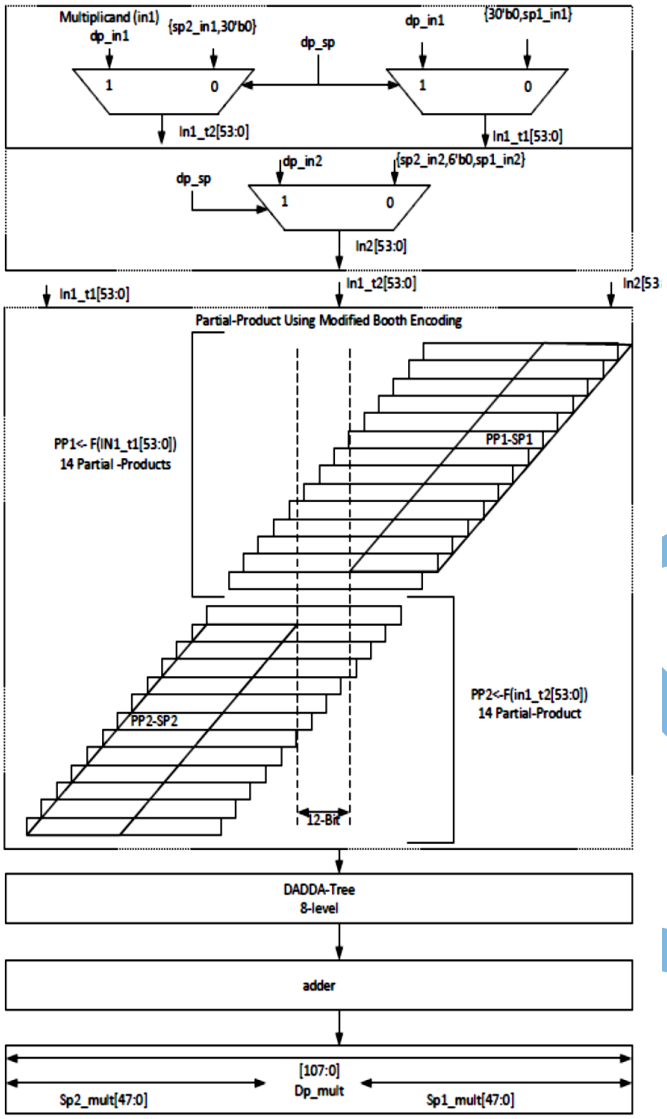

Fig.4.Dual-Mode Modified Booth Multipiler Architecture

Coming to the third satge architecture consists og 6,7,8 stages in algorithm in this acse the exponent underflow and mantissa quotient is given as the input to the dynamic right shifter or dual mode and then rounding is done which undergoes normalisation and exceptional processing. To the dynamic right shifter mantissa quotient and right shift amount is given as the inputs.right shifter is similar to dual mode dynamic left shifter also has 6 stages 5 stages are dual mode ans initial stage is single mode.top muxs provide SP quotient and third muxcombined and give DP quotient to produce right shifting The proposed dual mode rounding[3] produces round to the nearest method and it contains two steps ULP and ULP addition(unit-at-last-place) with mantissa quotients.ULP is based on the guard ,round and sticky bits.ULP is done for dual modes for DP and both SPs. mantissa quotient contains both SPs and DP quotients.ULP additions consists of 32-bit incrementer.the rounded quotient obtained from dual mode rounding is futher normalized into SPs and DP.if there is any overflow due to the unit-at-last place addition we requie 1 bit right shift for overflow. Finally, the normalized results for double precision and two parallel single precision are multiplexed by the MUX(2:1) to produce 64-bit output which contains quotients of single and double precision.

III. RESULTS

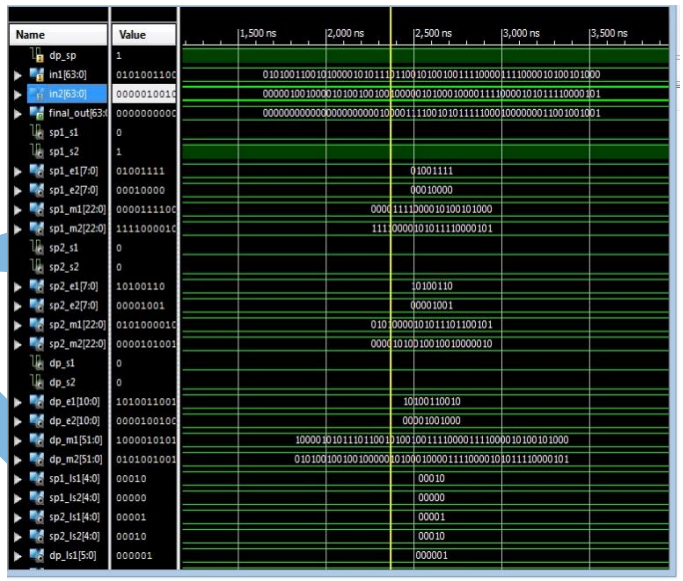

Fig.5.Simulation results of DPdSP architecture

	[4]	[5]	Proposed Multiplier	
Sub normal	X	✓	✓	X
Technology	90nm	90nm	90nm	90nm
Gate Count(based on minimum size inverter)	212854	118085	66416	62649
Period(in ns)	1.41	19	1.72	1.67
Power(mW)	-	11.92	30.9	33.45
Throughput(in Clock Cycles)	29/15	1/1	10/8	

Tabel.1.Comparision tabel of different divider architectures

The simulation results of DPdSP architecture is shown in Fig.5. and Area, Power results of the proposed and related architecture are given below in Tabel.1.By the results we can understand that the DpdSP architecture is area efficient and produce less power and through put compared to the remaining architectures

IV.CONCULSION

This paper proposed dual mode double precision architecture for floating point division arithmetic.it is used for both dual single and double precision operands.this architecture is madde based on series expnasion mantissa division .A dual mode modified radix 4 booth multipiler is used which will help in low overhead.entire architecture is used to perform dual mode computation and less hardware. By the DPdSP architecture for dual mode division we can use other multipilcative algorithms like goldschmidt and newton raphson this the future work

REFERENCES

[1] J.-C. Jeong, W.-C. Park, W. Jeong, T.-D. Han, and M.-K. Lee, "A costeffective pipelined divider with a small lookup table," IEEE Trans.
Comput., vol. 53, no. 4, pp. 489–495, Apr. 2004.

[2] M.K.Jaiswal ,M.Balakrishnan and Paul ,:series expansion based efficient architectures for double prescision floating point dicision"Circuits ,signal processing ,vol.33.no.11,pp.3499-3526,2014.

[3]Manish Kumar Jaiswal and Hayden K."Area Efficient architecture for Dual –Mode Double Precision Floating Point Divison",IEEE transcations on circuits and systems,VOL.64,Feb 2017

[4] A.Isseven and A.Akka,"A dual mode quadruple Folating point Divider,"in proc .2006,pp.1697-1701
[5]M.Jaiswal ,R.Cheung,"configurable architecture for double/two parallel single precision floating point division,"in proc.IEEE Computes ,VLSI(ISVLSI).Jul.2004,pp.332-337

[6]IEEE Standard for Floating point Arithmetic,IEEE Standard 754-2008,Aug.2008,pp.1-70.

[7] B.Pasca ,"Correctly rounded floating point division for DSP enabled FPGAs,"in proc conf Field program ,Aug.2012 ,pp.249-254.

[8]M.Jaiswal,B.Varma,M.balakrishnan,K.pauland"Configura bles architectures for multi mode floating point adders"VOl.62, Aug2015.

[9]A.Akka,s,"Dual mode quadruple precision floating point adder"in Proc,Euromicro symp,Digit ,syst,design ,2006,pp.211-220 .

[10]J.M.Muller"Handbook for floating point arithmetic,1st ed.basel,2009.

