VLSI Architecture for Floating Point Multipliers using Programmable Reversible Gate

[1]Nilesh Shankar Patil, [2]Dr. Vikas Gupta
M. Tech. Scholar, Department of Electronics and Communication, TIT, Bhopal, M.P.[1]
Professor & HOD, Department of Electronics and Communication, TIT, Bhopal, M.P.[2]

Abstract- Reversible logic preserves information and provides low power computing. Floating point multiplier is an important part of arithmetic in DSP applications. IEEE754 provide two basic and widely used formats for representing floating point numbers namely Single Precision Floating point (SPFP) and Double Precision Floating Point (DPFP). In this paper, we have proposed designs for Reversible SPFP and DPFP point multiplier which requires optimum design of 24X24 and 53X53 multiplying circuitry with reversible gates respectively. Also we presented sign bit calculation and exponent bit calculation circuits by reversible logic. Reversible Mantissa bits calculation circuitry requires consumes more area as it has large value of quantum cost and garbage outputs as compare to reversible sign bit and exponent bits calculation circuitry. So, we mainly focused our work on to minimize the area consumed by the reversible mantissa bits calculation circuit by minimizing the value of quantum cost and garbage outputs of it. In this paper, we presented two ways for partial products generation with reversible gates. We propose a new design of reversible 12X12 reversible Wallace tree multiplier for single precision and double precision mantissa multiplier circuit. Also 5X5 and 5X12 reversible Wallace tree multiplier for double precision multiplier. Here we use the optimized designs of reversible half adder, Full adder and 4:2 compressors. In the final summation stage, we carefully shifted the product terms and add them. For this stage we have carefully chosen the adders and compressor in such way to get the optimized results in terms of quantum cost and Garbage output. We have also shown the improvement in single precision floating point multiplier in terms of Quantum cost and Garbage output.

***Index Terms* –- Reversible Half Adder (RHA), Reversible Full Adder (RFA), Reversible 4:2 Compressor (R4:2C), Reversible Partial Products Stage (RPPS), Reversible Partial Products Addition Stage (RPPAS).**

I. INTRODUCTION

Reversible logic circuit is an emerging area of technology. Reversibility has a unique property of having one to one mapping between input and its corresponding output. Reversible gates also have equal number of inputs and outputs. Landauer has shown kTIn2 joules of energy lost if one bit of information has lost. Here, k is the Boltzmann's constant and T is the absolute temperature at which computation has performed. Reversible logic saves this energy by preserving information. Quantum cost and Garbage output are the two basic and important parameters of reversible circuit which are need to be kept less for optimized reversible circuit. Quantum cost represents the number of 1X1 and 2X2 Quantum gates used in the circuit. Garbage outputs are number of output ports which remains unused in the circuit. But these outputs are important as they preserve the reversibility of the circuit [1].

II. METHODOLOGY

There are many types of Reversible gates available with their specific value of quantum cost and garbage outputs. More than one gate may have same value of outputs but they compare on the basis of basic attributes of Reversible gates i.e., quantum cost and garbage outputs [2]. A basic reversible gate used in this paper has been shown in figure [1].

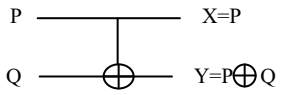

Figure 1 (a): Internal Design of Feynman gate

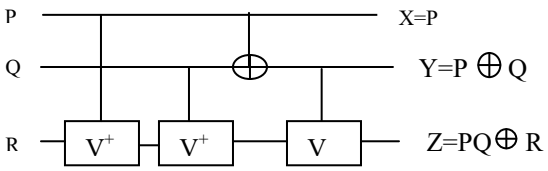

Figure 1 (b): Internal Design of Peres gate

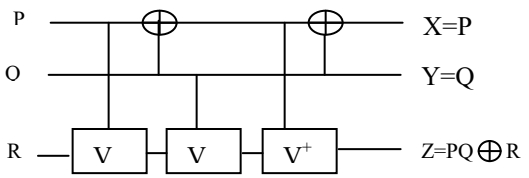

Figure 1(c): Internal Design of Toffoli gate

Some of the important gates used in this paper are the Feynman gate, the toffoli gate, the peres gate and the HNG/RFA gate [3] is shown in figure 1. They are the basic building blocks used in our designs of reversible sign bit, exponent bits and mantissa bits calculation of SPFP and DPFP multiplier.4:2 compressors uses two HNG/RFA gates. The Feynman gate is a 2X2 reversible gate which has quantum cost 1 and garbage output 1 is shown in figure 1(a).

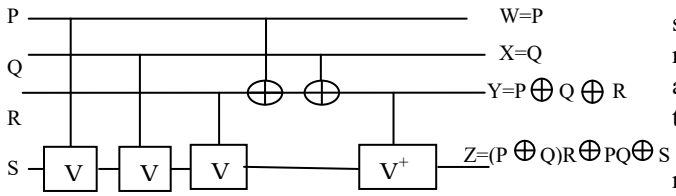

Figure 1 (d): Internal Design of HNG/RFA

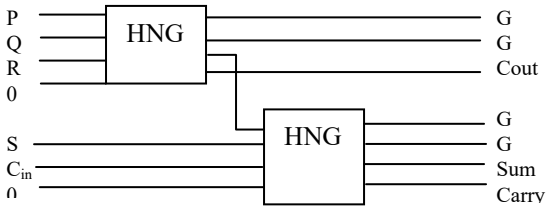

Figure 1(e): Reversible 4:2 compressor

Figure 1: Reversible Gates used

The Peres gate is also a 3X3 reversible gate whose quantum cost is 4 and garbage outputs of 2 is shown in figure 1(b). The Toffoli gate is a 3X3 gate has quantum cost of 5 and garbage outputs of 2 is shown in figure 1(c). The HNG/RFA is 4X4 reversible gate whose quantum cost is 6 and garbage outputs are 6 is shown in figure 1(d). The Reversible 4:2 Compressor (R4:2C) is the cascaded combination of two HNG/RFAs having the value of quantum cost 12 and garbage outputs 4 is shown in figure 1(e).

III. SPFP AND DPFP MULTIPLIER

SPFP format has a single bit as sign bit, 8 bits for exponent representation and 23 bits mantissa. DPFP representation also has three fields same as SPFP with 1 bit for sign, 11 bit for exponent and 52 bit for mantissa representation in figure 2.

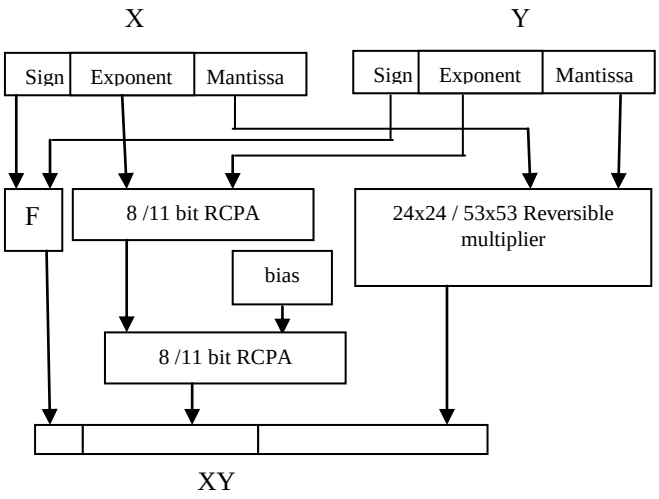

Figure 2: Reversible SPFP and DPFP Multiplier

For multiplying two SPFP or DPFP numbers, sign bit needs to be xoring together. Exponent bits need to be added together then bias (127 for SPFP and 1023 for DPFP) should be subtracted from it. Hidden '1' appended in the mantissa field then the mantissa will be of 24 bits in SPFP and 53 bits in DPFP. Now mantissa bits are multiplied together. So here we need a 24X24 multiplier for SPFP ultiplication and 53X53 multiplier for DPFP multiplication. Here we design reversible circuits for all the calculations by reversible gets to do the desired multiplication. Their corresponding quantum cost and garbage outputs are also calculated [4]. Reversible SPFP and DPFP multiplication algorithm has been shown in figure [2].

IV. PROPOSED METHODOLOGY

We propose the designs of our multipliers in two stages namely Reversible partial products generation stage (RPPGS) and partial products addition stage (RPPAS). Finally we add the products from different multipliers by carefully shifting the terms according to which part of the first operand is multiplied to the part of the second operand. In each multiplier we design two types of RPPGS, first by using all Peres gates and the second by using the combination of Toffoli and Peres gates. As we compare both the ways of partial products generation, we have found that the RPPGS with all peres gates exhibits less Quantum cost but more garbage outputs over the RPPGS with the Toffoli gates cascaded with Peres gates is shown in figure 3. In RPPGS, partial products of single bit of one operand with the single bit of second operand are generated. For this work, Toffoli gate and Peres gate are the suitable candidate with their third input as '0'. Toffoli gate and Peres gate, both gives the product (as a AND gate) of the first two inputs at their third output when their third input is zero. In RPPAS, the addition of the various partial product terms is done with the carefully chosen RHAs, RFAs and R4:2Cs so as to get the minimized value of Quantum cost and garbage outputs. We have been used four 12X12 multipliers for single precision floating point mantissa multiplier. Sixteen 12X12 multipliers, eight 5X12 multipliers and one 5X5 multiplier for single precision floating point mantissa multiplier is DPFP. 12X12 multiplier needs 144 partial products to be generated. For this purpose, one way is to use 144 Peres gates each one for each product term. Another way is to use Toffoli gates cascaded with peres gates which gives advantage of less garbage output as their two garbage outputs will be used in the preceding gates. This multiplier has used 28 RHAs, 28 RFAs and 35 4:2 Cs in its RPPAS in our design. 5X12 multiplier needs 60 partial products to be generated. In RPPGS of this multiplier, we have been calculating the Quantum cost and garbage output of both the ways i.e., by using all Peres gates and using Toffoli with Peres gates. In RPPAS, it uses 5 RHAs, 25 RFAs and 9 4:2 Cs. In our design of 5X5 multiplier 25 partial product terms are generated by two ways as in case of above two multipliers. In RPPAS, this multiplier uses 5 RHAs, 11 RFAs and 2 4:2 Cs. Comparative statistics of various Reversible multipliers

used in this paper have been shown in table 1 and table 2 in terms of Quantum cost and garbage outputs.

Table 1: Various multipliers showing their Quantum cost

Multipliers	Quantum cost		
	RPPGS		RPPAS
	With all Peres gates	With Toffoli and Peres gates	
12X12	576	697	701
5X12	240	284	278
5X5	100	116	110

Table 2: Various multipliers showing their Garbage output

Multipliers	Garbage outputs		
	RPPGS		RPPAS
	With all Peres gates	With Toffoli and Peres gates	
12X12	167	46	217
5X12	65	21	136
5X5	30	14	35

Michael et al has shown the SPFP 24X24 mantissa multiplier with nine 8X8 multipliers. We have proposed this multiplication by using four 12X12 multipliers with the improved value of Quantum cost and garbage outputs as shown in the following table 3. We have concluded that as the size of the part of the decomposed operand is increased, Quantum cost and Garbage outputs are reduced. But larger size of the parts of the decomposed operand increases the complexity of the circuits and also the overall calculation. In this paper, same type of RPPGS of Toffoli gates cascaded with Peres gates is used as used in Michael et al paper [1].

Table 3: Cost of SPFP multiplier

		SPFP multiplier	DPFP multiplier
Sign bit calculation	QC	1	1
	GO	1	1
Exponent bits calculation	QC	82	118
	GO	28	40

A. REVERSIBLE SIGN BIT AND EXPONENT BITS CALCULATION

Sign bit calculation of the multiplication of SPFP and DPFP numbers required a common method i.e., XORing the single sign bits of both the operands together. The Feynman gate serves the purpose of Reversible XOR (RXOR) gate very well with 1 Quantum cost and 1 garbage output. Exponent bits of the product of two SPFP or DPFP number has been calculated by adding exponent bits (8 bits for SPFP and 11 bits for DPFP) together and then subtract bias (127 for SPFP and 1023 for DPFP) from it. This procedure required two carry propagate adders. Here we use two 8 bit RCPA (Reversible Carry Propagate Adder) for exponent calculation of SPFP multiplier and two 11 bit RCPA for exponent calculation of DPFP multiplier. One 8 bit RCPA require 1 RH A, 6 RFA and 1 RXOR. One 11 bit RCPA needs 1 RHA, 9 RFA and 1RXOR. Quantum cost (QC) and garbage outputs (GO) of the sign bit and exponent bit calculation are shown in the table 4.

Table 4: Cost of sign bit and exponent bits calculation

	Quantum Cost	Garbage outputs
Previous method	6591	1387
New design	5911	1156
Improvement	10.31%	16.65%

B. MANTISSA BITS CALCULATION

In this paper, for mantissa bits calculation mantissas of both the operands are multiplied by using operand decomposition method. In SPFP multiplier, 24X24 multiplication has been done by four 12X12 Reversible multipliers. Finally these product terms are added up together by arranging them in proper positions. In DPFP, 53X53 multiplication has been done by breaking the each operand into 5 groups. Four groups of 12 bits each and one group of 5 bits. Here we get 25 product terms by 25 Reversible multipliers i.e., sixteen 12X12 Reversible multipliers results in product term each of 24 bits, eight 5X12 Reversible multipliers results in product term each of 17 bits and one 5X5 reversible multiplier.

term each of 17 bits and one 5X5 Reversible multiplier results in product term each of 10 bits. These 25 product terms are needed to be added very carefully. This Reversible circuit for addition has become very complex. Complexity increases the chances of mistakes in calculation. So complexity can be reduced by adding these terms in twosteps. In first step, we sort out the product terms according to their positions and the number of bits. We add the terms having the same positions keeping in mind the number bits they have. In second step, we arrange the results of addition side by side according to their positions. Now we propagate the carry of one over to another and get the final result of 53X53 multiplications in this final stage of Reversible shifted Addition Stage (RSAS) is shown in table 5.

Table 5: QC and GO of DPFP mantissa multiplier

	QC		GO	
	All Peres in RPPGS	Toffoli and Peres in RPPGS	All Peres in RPPGS	Toffoli and Peres in RPPGS
12X12 multipliers	20432	22352	4608	3156
5X12 multipliers	4144	4496	1608	1248
5X5 multiplier	210	226	65	49
RSAS	2790		916	

V. RESULTS AND STATISTICS

All the units presented in this paper are designed in reversible manner including RHA, RFA, and R4:2C, RXOR, 12X12, 5X5, 12X12 Reversible multipliers. All these units are functionally verified, also the QC and GO has been calculated for optimized results at each and every step of calculation. Table 6 presents the QC and GO of the overall Reversible circuit of binary 32 SPFP and 64DPFP multiplier.

Table 6: Cost of SPFP and DPFP multiplier

		SPFP multiplier	DPFP multiplier
All Peres in RPPGS	QC	5510	27695
	GO	1669	7238
Toffoli and Peres in RPPGS	QC	5994	29983
	GO	1185	5401

VI. CONCLUSION

We have concluded that Reversible logic circuits are the emerging field of research as they preserve information and thereby save energy. A traditional circuit dissipates kTIn2 joules of energy each time one bit of information is lost. On the other hand, Reversible circuits prove its utility in low power computing as they minimize this information loss. We have been presented new designs of Reversible DPFP multiplier which uses our designs of 12X12, 5X12 and 5X5 Reversible multipliers. We have also presented our new design of Reversible SPFP multiplier which has shown a significant improvement in terms of Quantum cost and garbage output over the existing design. We have also concluded that in the method of multiplication by operand decomposition as the size of the groups in the decomposed operand increased, value of Quantum cost and garbage outputs tends to reduce but at the same time it increases the complexity involved. We have proposed two generalized methods of partial products generation with the comparison between them in terms of two important attributes of Reversible logic. Partial product generation of each multiplier has been done by these two methods in this paper. The quantum cost and garbage outputs of the entire SPFP and DPFP multiplier has been presented in this paper.

REFERENCES

[1] Mioara Jolde and Bogdan Pasca, “Efficient Floating-Point Implementation of the Probit Function on FPGAs”, 31st International Conference on Application-specific Systems, Architectures and Processors (ASAP), IEEE 2020.

[2] Michael Nachtigal , “Design of a Reversible single Precision Floating Point M multiplier Based on Operand Decomposition “, 978-1-4244-7032-7/10.

[3] Raghava Garipelly, P.Madhu Kiran and A. Santhosh Kumar, “A Review on Reversible Logic Gates and their Implementation”, in International Journnal of Emerging Technology and Advanced Engineering, [March 2013, ISSN 2250-2459]

[4] Somayeh Babazadeh and Majid Haghparast, 2012.Design of a Nanometric Fault Tolerant Reversible Multiplier Circuit, Journal of Basic and Applied Scientific Thesis.

[5] L. Louca, T. A. Cook and W. H. Jhonson , “implementation of IEEE Single Prescision Floating Point Addition and Multiplication on FPGAs”, proceeding of 83rd IEEE Symposium on FPGAs for Custom Computing Machine (FCCM 96),(1996),pp. 107-116.

[6] Soumya Havaldar, K S Gurumurthy, “Design of Vedic IEEE 754 Floating Point Multiplier”, IEEE International Conference On Recent Trends In Electronics Information Communication Technology, May 20-21, 2016, India.

[7] Ragini Parte and Jitendra Jain, “Analysis of Effects of using Exponent Adders in IEEE- 754 Multiplier by VHDL”, 2015 International Conference on Circuit, Power and Computing Technologies [ICCPCT] 978-1-4799-7074-2/15/$31.00 ©2015 IEEE.

[8] Ross Thompson and James E. Stine, “An IEEE 754 Double-Precision Floating-Point Multiplier for Denormalized and Normalized Floating-Point Numbers”, International conference on IEEE 2015.

[9] M. K. Jaiswal and R. C. C. Cheung, “High Performance FPGA Implementation of Double Precision Floating Point Adder/Subtractor”, in International Journal of Hybrid Information Technology, vol. 4, no. 4, (2011) October.

[10] Fagin and C. Renard, "Field Programmable Gate Arrays and Floating Point Arithmetic," IEEE Transactions on VLS1, vol. 2, no. 3, pp. 365-367, 1994.

[11] N. Shirazi, A. Walters, and P. Athanas, "Quantitative Analysis of Floating Point Arithmetic on FPGA Based Custom Computing Machines," Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines (FCCM"95), pp.155-162, 1995.

[12] Malik and S. -B. Ko, “A Study on the Floating-Point Adder in FPGAs”, in Canadian Conference on Electrical and Computer Engineering (CCECE-06), (2006) May, pp. 86–89.

[13] Sangwan and M. K. Yadav, “Design and Implementation of Adder/Subtractor and Multiplication Units for Floating-Point Arithmetic”, in International Journal of Electronics Engineering, (2010), pp. 197-203.