FPGA Implementation on Reversible Floating Point Multiplier

M.Jenath, V.Nagarajan

Abstract-Field programmable gate arrays (FPGA) are increasingly being used in the high performance and scientific computing community to implement floating-point based system. The reversible single precision floating point multiplier (RSPFPM) requires the design of reversible integer multiplier (24×24) based on operand decomposition approach. Reversible logic is used to reduce the power dissipation than classical logic and do not loss the information bit which finds application in low power computing, quantum computing, optical computing, and other emerging computing technologies. Among the reversible logic gates, Peres gate is utilized to design the multiplier since it has lower quantum cost. Operands of the multiplier is decomposed into three partitions of 8 bits each using operand decomposition method. Thus the 24×24 bit reversible multiplication is performed through nine reversible 8x8 bit multipliers and output is summed to yield an efficient multiplier optimized in terms of quantum cost, delay, and garbage outputs. This proposed work is designed and developed in the VHSIC hardware description language (VHDL) code and simulation is done using Xilinx 9.1simulation tool.

Key words: Reversible logic gates, reversible logic circuits, reversible multiplier circuits, quantum computing, Nanotechnology based systems.

I. INTRODUCTION

In VLSI circuit design, reduction of power dissipation is the one of the major goal. As demonstrated by R.Landauer in the early 1960s, irreversible hardware computation, regardless of its realization techniques, results in energy dissipation due to the information loss [1]. It is proved that the loss of each one bit of information dissipates at least KTln2 joules of energy(heat), where K=1.3806505x$10^{-23}$$m^2kg^{-2}K^{-1}$ (joules Kelvin-1) is the Boltzmann's constant and T is the absolute temperature[1].

Reversible logic circuits have theoretically zero internal power dissipation because they do not lose information. Hence, In 1973, Bennett showed that in order to avoid KTln2 joules of energy dissipation in a circuit, it must be built using reversible logic gates [2].A circuit is said to be reversible if the input vector can be uniquely recovered from the output vector and there is a one-to-one correspondence between its input and output assignments, i.e. not only the outputs can be uniquely determined from the inputs, but also the inputs can be recovered from the outputs .

Manuscript Received January 09, 2012.

M.Jenath, Department of Electronics and Communication Engineering, Anna university of Technology. Adhiparasakthi Engineering College, Melmaruvathur, Chennai, India (**jenath.mohamad@gmail.com**)

***Dr.V.Nagarajan**, Department of Electronics and Communication Engineering, Anna university of Technology. Adhiparasakthi Engineering College, Melmaruvathur, Chennai, India **(nagarajanece31@rediffmail.com)**

Thus the number of inputs and outputs in reversible logic gates or circuits are equal. Reversible logic has received great attention in the recent years due to their ability to reduce the power dissipation which is the main requirement in low power Very large scale integration (VLSI) design. Quantum computers are constructed using reversible logic circuits. It has applications in various research areas such as low power CMOS design, optical computing, DNA computing, quantum computing, nanotechnology bioinformatic and thermodynamic technology. It is not possible to construct quantum circuits without reversible logic gates. Synthesis of reversible logic circuits is significantly more complicated than traditional irreversible logic circuits because in a reversible logic circuit, fan-out and feedback is not allowed. Thapliyal and Srinivas proposed the TSG gate, in reversible circuit designing. The construction of such new gates does not make any significance to reduce quantum cost nor the gate complexity[3]. Mohammadi et al (2009) has proposed a new reversible (4×4) multiplier circuit using "HNG" gate which has minimised the quantum cost with previous design. Michael Nachtigale et al (2010) proposed the single precision floating point multiplier (32 ×32) multiplier by using both Toffoli and Peres gate. This design uses the reversible 4:2 compressor and reversible Wallace tree multiplier.This reversible design of the 8x8 bit Wallace tree multiplier has been optimized in terms of quantum cost, delay, and number of garbage output [7].

Multiplication is a heavily used arithmetic operation in many computational units. It is necessary for the processors to have high speed multipliers with less hardware complexity. Floating point numbers are one possible way of representing real numbers in binary format. The IEEE 754 standard presents two different floating point formats, Binary interchange format and Decimal interchange format. Multiplying floating point numbers is a critical requirement for DSP applications involving large dynamic range. This paper focuses only on single precision binary interchange format. The proposed system describes a reversible single precision floating point multiplier (SPFP) using only the Peres gate. The efficient 24x24 bit reversible significand multiplication is performed through nine reversible 8x8 bit multipliers .Since Peres gate has lowest quantum cost compared to other reversible logic gates. The optimized values of quantum cost, gate delay, and garbage output is obtained compared to the existing design and reduces the hardware complexity of the system. Section 2 gives the different types of the reversible logic gates required for the present work. Section 3 describes the design of reversible multiplier circuit.Section 4 discuss the design of reversible exponent

addition.Section 5 deals with the final results and conclusion.

II. REVERSIBLE LOGIC GATES

Power dissipation is important factors in VLSI design.The recent computers erases a bit of information every time and dissipate the power while they perform a logical operation. Such logical operations are called "irreversible logic". An example of information being lost can be seen in an ordinary AND gate. An AND gate has two inputs and only one output, which means that in the process of moving from the input to the output of the gate, we loss one bit of information. The main advantages of reversible logic is low power consumption, reduce the information loss in the bit, minimise the power dissipation. Reversible logic gate is an n-input, n-output logic device with one-to-one mapping. This is used to determine the outputs from the inputs and also the inputs can be uniquely recovered from the outputs.

Fig 1 and 2 shows the classical gate (irreversible gate) and general N×N reversible gate.

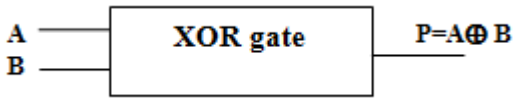


Fig 1. Classical (Irreversible) gate

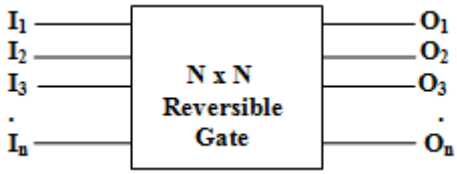


Fig 2. N x N Reversible gate

In the reversible XOR gate there is no loss of information bit signals. Since it maps the input vector with output vector which gives the equal number of inputs and output and it is shown in Fig 3.

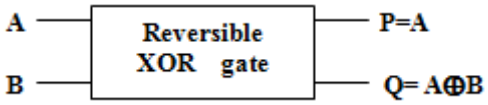


Fig 3. Reversible XOR gate

Peres gate is represent as 3×3 vector in Fig 4.In the proposed design, Peres gate is used because of its lowest quantum cost. The input vector is I (A, B, C) and the output vector is O (P, Q, R). The output is defined by P = A, Q = A⊕B and R=AB⊕C. Quantum cost of a Peres gate is 4.

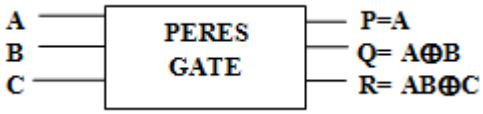


Fig 4 . 3×3 Peres gate.

The input vector of Feynman gate is I (A, B) and the output vector is O (P, Q). The outputs are defined by P=A, Q=A ⊕ B and it is shown in Fig 5. Quantum cost of a Feynman gate is 1.

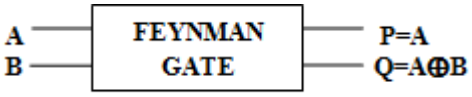


Fig 5 . 2×2 Feynman gate.

Fig 6 shows a 3×3 Fredkin gate. The input vector is I (A, B, C) and the output vector is O (P, Q, R). The output is defined by P=A, Q=A′B⊕AC and R=A′C⊕AB. Quantum cost of a Fredkin gate is 5.

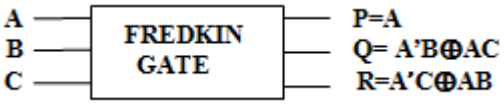


Fig 6. 3×3 Fredkin gate.

The 3×3 Toffoli gate is represent in Fig 7. The input vector is I(A, B, C) and the output vector is O(P,Q,R).The outputs are defined by P=A, Q=B, R=AB⊕C. Quantum cost of a Toffoli gate is

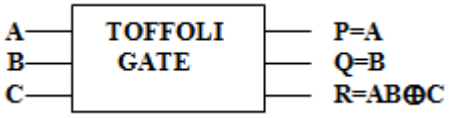


Fig 7. 3×3 Toffoli gate.

III. DESIGN OF REVERSIBLE SPFP MULTIPLIER

Block diagram which is shown in Fig 8 represents the design of reversible single precision floating point multiplier (RSPFPM)

The sign magnitude of multiplied product is the XOR function of X and Y. For the design of reversible 24 x 24 bit multiplier, the operands are decomposed into three partition of 8 bits each. The proposed reversible design of the 24x24 reversible multiplier gives the optimized values for quantum cost, delay, and number of garbage outputs. The optimization is done at each stage by using reversible full adders and half adders. The exponents are added using the reversible ripple carry adder and the bias value -127 is subtracted from exponents using ripple borrow subtractor.The design is developed in the VHDL code and simulation result is obtained in Xilinx Software tool .

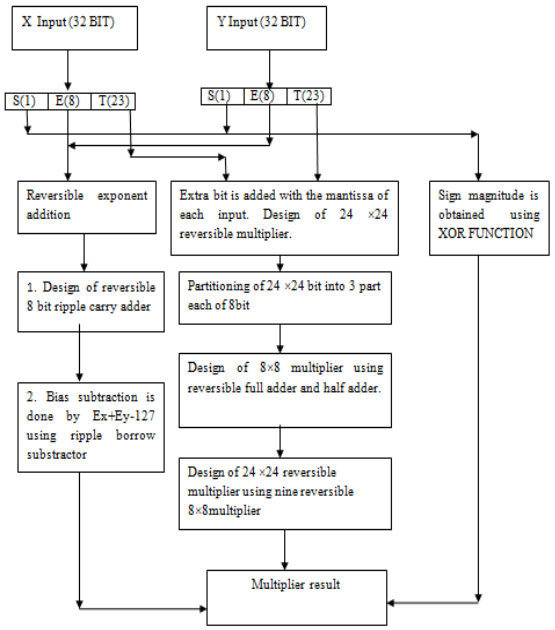


Fig 8.Block diagram for Reversible Single Precision FLP Multiplier.

A. Single Precision Floating Point Multiplier

IEEE754 standard defines the format for a single precision floating point number stored in 32 bits. The standard three fields for a single precision floating point numbers are a single sign bit S, an 8 bit biased exponent E, and a 23 bit trailing significand T. The three fields are illustrated in Fig 9.The XOR function, provided by a single Peres gate, suffices to calculate the sign of the product, as the product will be negative precisely when the signs of the two floating point factors differ. The field E is regarded as an unsigned integer that represents the signed exponent of the floating point number with a bias of 127 applied.

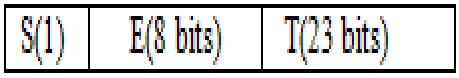


Fig 9.Single precision floating point multiplier format

Each significand is a 24 bit fixed point number, so we can use a 24x24 bit reversible multiplier to calculate the product of the significands occupies 48 bits.The product of the significands to fit it to the floating point format, the sum of the unbiased exponents of *x* and *y* will need to be biased. Because removing the bias from both exponents requires us to subtract 254, but adding in the bias for the difference requires us to add 127, the new biased exponent of the product of *x* and *y* will be E_{xy}= (E_x-127) + (E_y-127) +127=E_x+E_y-127.This exponent of the product can be computed reversibly by four 2bit reversible carry propagate adders (RCPAs) in series.

B.Reversible partitioning of the operand

Partitioned multipliers use a form of a technique called operand decomposition. The circuit treats the input numbers as concatenated inputs of smaller sizes, which are manipulated using a specific technique to calculate the output corresponding to the original inputs. The operand decomposition (OD) techniques exist that are motivated by power savings, speed and area. Reducing quantum cost, garbage outputs, or delay motivates the design of a reversible partitioned multiplier. It reduces switching activity in binary multipliers.

In the design of 24x24 bit numbers, *A* and *B*, to be multiplied are logically partitioned into 3 vectors of 8 bits each. *A* is divided into AH, AM, and AL and *B* is similarly divided into BH, BM and BL. Thus the 24x24 bit partial product generation is carried out using nine of optimized 8x8 bit reversible multiplier. The OD process decreases the number of bits with the value of '1' in the multiplicands and reduces the amount of approximation. This partioning of two 24 bit operand is given in [7].

C.Design of Reversible Multiplier

The design of the proposed reversible multiplier is done using two steps.

Partial Product Generation (PPG)
Conventional addition (CA)

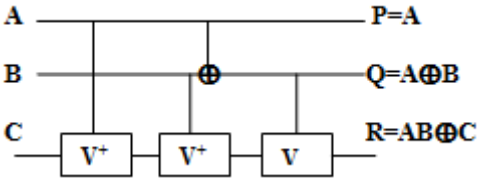
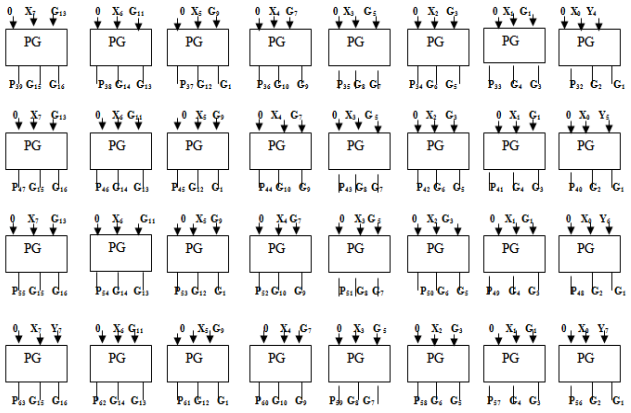


Fig 10. Peres gate Quantum cost 4

An 8x8 bit unsigned multiplication is performed in a reversible manner by utilizing only the Peres gate for the design to generate the 64 one bit partial products. The Fig 10 shows the quantum cost of Peres gate.The 64 partial products are obtained for 8x8 bit reversible multiplication X × Y=([*x7, x6,…. x0*] × [*y7, y6,…. y0*]) and is shown in Fig 11. In the summation stage of multiplier, this design realizes a lower quantum cost and fewer garbage outputs by virtue of proposed design of reversible full adder and half adder. Peres gate provide the necessary AND operator when their input C is hardwired to 0.Peres gates can generate the 64 partial products. The connection in series reduces garbage outputs and the substitution of Peres gates reduces the quantum cost. A cascade of two Peres gate can generate the full adder operation.



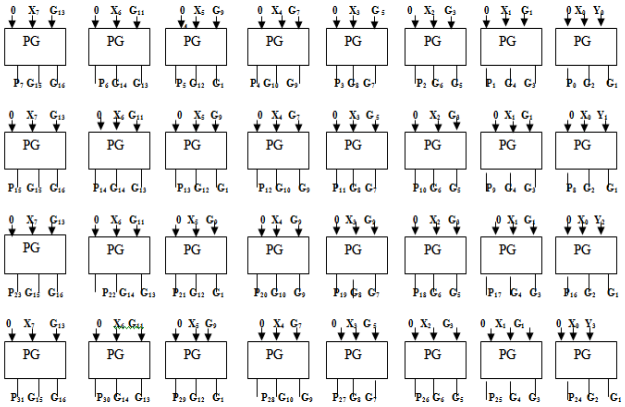


Fig 11 .Generation of 64 partial products

D.Design Using Peres Gate in Proposed System

Peres gate is used to realize the different logical functions. For the design of single precision floating point multiplier, reversible half adder (RHA) is obtained from Peres gate with the hardwired control of c=0. The expression became as Q=A⊕B, R =AB which is equal to the sum and carry out of half adder. This is represented in Fig 12.

Fig 12. Reversible Half adder

Reversible full adder (RFA) circuit is obtained by cascading the two Peres gate as shown in Fig 13.The sum and carry output of reversible full adder is given by the Boolean expression in equation 1&2.

$$S = A \oplus B \oplus C_{in} \quad (1)$$

$$Cout = (A \oplus B)C_{in} \oplus AB \quad (2)$$

Peres gate is equal with the transformation produced by a Toffoli Gate followed by a Feynman Gate. The final design of the reversible 8x8 bit reversible multiplier involves the use of reversible full adders, reversible half adders to summed the partial products for obtaining 16 bit output result for 8×8 multiplier which is given in Fig 14.

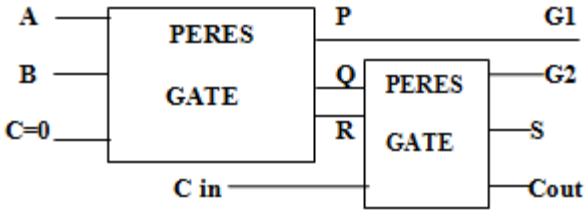


Fig 13. Reversible Full adder

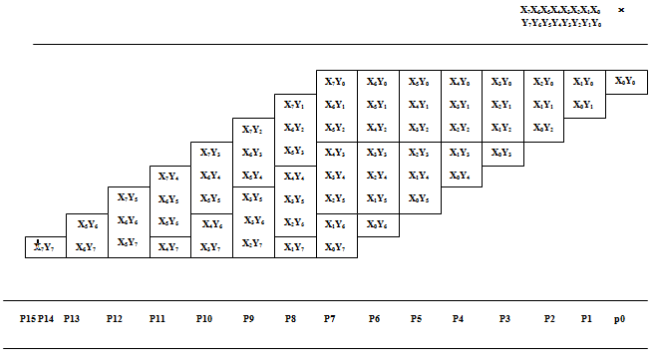


Fig 14.16 bit output in the reversible 8x8 multiplier

IV. DESIGN OF REVERSIBLE EXPONENT ADDITION

The full adder is the basic building block in the ripple carry adder, and most other adder circuits. The full adder computes the sum bit S_i and carry output bit C_{i+1} based on its addend input bits X_i and Y_i, and its carry input bit C_i. The sum bit and carry output bits are given in equation below.

$$S_i = X_i \oplus Y_i \oplus C_i \quad (3)$$

$$C_{i+1} = X_iY_i + X_iC_i + Y_iC_i \quad (4)$$

A. Reversible ripple carry adder

An 8-bit reversible ripple carry adder is used to add the two input exponents. As shown in Fig 15, a ripple carry adder is a chain of cascaded full adders and one half adder.Each full adder has three inputs (A, B, Ci) and two outputs (S, C_o). The carry out (C_o) of each adder is fed to the next full adder.

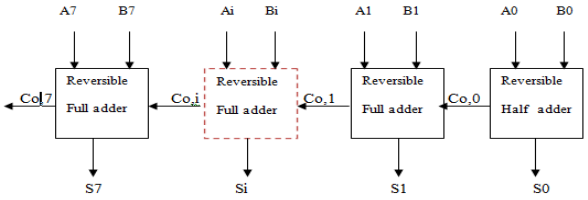


Fig15 -Reversible Ripple Carry Adder

The addition process produces an 8 bit sum (S7 to S0) and a carry bit (Co,7). These bits are concatenated to form a 9 bit addition result (S8 to S0) from which the Bias is subtracted. The Bias is subtracted using an array of ripple borrow subtractors.

B.Ripple borrow subtractor

Generally subtractor has three inputs (minuend (S), subtrahend (T), Borrow in (Bi)) and two outputs (Difference (R), Borrow out (Bo)). The subtractor logic can be optimized if one of its inputs is a constant value which is our case, where the Bias is constant $(127)_{10} = (001111111)_2$.

Block diagram for 1 bit subtractor is shown in Fig 16. Table 2 represent The

truth table for a 1-bit subtractor with the input T equal to 1 is called as "one subtractor (OS)".

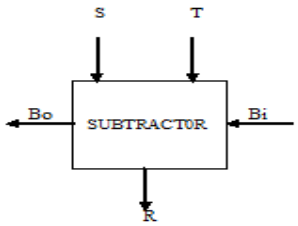


Fig 16.1 Bit subtractor

Table 2 - 1 Bit subtractor with the input T=1

S	T	Bi	Difference(R)	Bo
0	1	0	0	0
1	1	0	1	0
0	1	1	1	1
1	1	1	1	1

The Boolean equations to represent this subtractor is given as

$$\text{Difference}=(S \oplus B_i)' \quad (5)$$
$$\text{Borrow} \quad =S' + B_i \quad (6)$$

Table 3 shows the truth table for a 1-bit subtractor with the input T equal to 0 is called as "zero subtractor (ZS)".

Table 3 - 1 Bit subtractor with the input T=0

S	T	Bi	Difference(R)	Bo
0	0	0	0	0
1	0	0	1	0
0	0	1	1	1
1	0	1	0	0

The Boolean equations to represent this subtractor is given as

$$\text{Difference}=(S \oplus B_i) \quad (7)$$
$$\text{Borrow}=S'.\ B_i \quad (8)$$

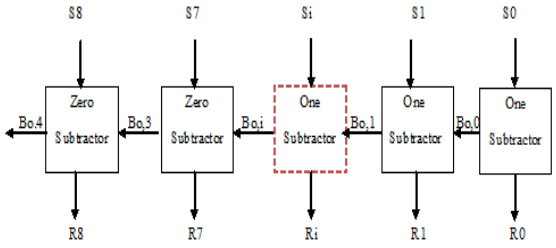


Fig 17.Ripple borrow subtractor

Fig 17 shows the Bias subtractor which is a chain of 7 one subtractors (OS) followed by 2 zero subtractors (ZS).The borrow output of each subtractor is fed to the next subtractor.

The XOR function is provided by a single Peres gate to calculate the sign of the product and will be negative precisely when the signs of the two floating point factors differ.

V. RESULT ANALYSIS & SIMULATION OUTPUT

In this paper, the cost metric parameters such as quantum cost, delay, and number of garbage outputs for each of the units designed is verified with the existing system.[7]

Table 4 gives the cost of the reversible partial product generation unit (RPPG) for the proposed system with the existing system[7].

Table 4- Cost of proposed 8×8 RPPG unit

	Quantum Cost	Garbage Outputs	Quantum Delay
Proposed Design Using Gate Peres	64	128	256
Existing Work Both Toffoli &Peres Gate	320	64	1280

The cost of the reversible 8×8 reversible multiplier unit

for the proposed system with the existing system[7] is shown in Table 5

Table 5-Cost of proposed 8×8 reversible multiplier

	Quantum Cost	Garbage Outputs	Quantum Delay
Proposed Design Using Gate Peres	171	179	684
Existing Work Both Toffoli &Peres Gate	1113	186	632

The table 6 gives the cost of proposed reversible single precision floating point multiplier with the existing system[7].

Table 6- Cost of reversible single precision floating point multiplier

	Existing system SPFP multiplier using Toffoli &Peres gate			Proposed system Single precision floating point multiplier using Peres gate		
	Quantum Cost	Garbage Outputs	Quantum Delay	Quantum Cost	Garbage Outputs	Quantum Delay
Sign	1	1	1	1	2	4
Exponent	82	31	492	15	15	60
24×24 Multiplier	6591	1387	39446	6536	1706	26144
TOTAL	6674	1419	39939	6552	1723	26208

Thus the result statistics shows that the design of this work gives the efficient optimized values for the quantum cost, garbage outputs and quantum delay.

The simulation output is obtained by using Xilinx simulation tool is as follows.

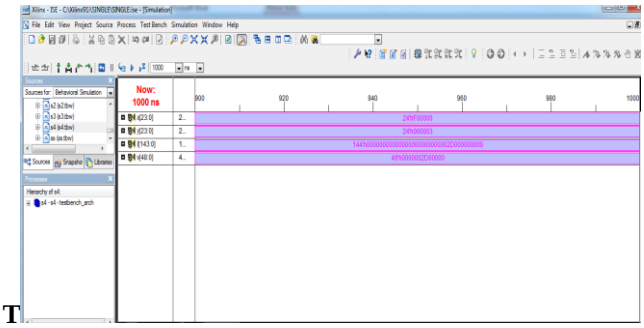


Fig 18. Reversible 24×24 Multiplier

The design of reversible 24×24 multiplier output which is obtained by Xilinx is shown in Fig 18. The output of this multiplier has product of 48 bits.

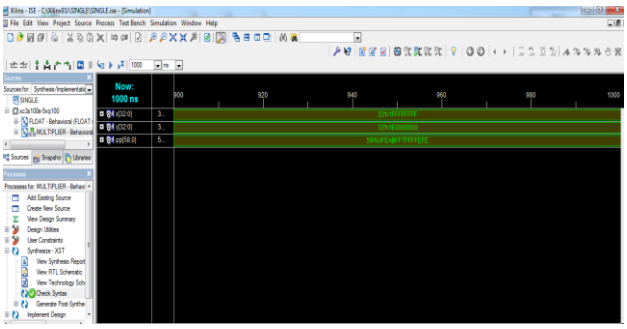


Fig 19- Reversible SPFP Multiplier

The simulation output for the design of single precision floating point multiplier is obtained and is shown in Fig 19.This is the final stage of the design and the product output is represented as pp in the waveform.

VI. CONCLUSION

Reversible single precision floating point multiplier utilizes the reversible 8×8 multiplier, reversible full adder and half adder to impose an efficient optimized multiplier in measurement of quantum cost, gate delay and garbage outputs .This proposed system can be used in the design of complex systems in nanotechnology.

The design of reversible single precision floating point multiplier reduces the quantum cost(6552) and quantum delay (26205) compared with the existing system[7] with increased garbage outputs(1723).It is also proposed to design a reversible exponent calculator with high precision values.

REFERENCES

1. R.Landauer, "Irreversibility and heat generation in the Computational process,"IBM J. Research and development, 5, pp. 183-191, 1961.
2. C. H. Bennett, "Logical reversibility of computation," IBM J. Research and Development, pp. 525-532, November,1973.
3. H.Thapliyal and M.B.Srinivas, "Novel reversible 'TSG' gate and its application for designing components of primitive/reversible quantum ALU," Proc. the 5th IEEE International Conference on Information, Communications and Signal Processing, Bangkok, Thailand, December 6-9, 2005.
4. M.Haghparast, M.Mohammadi, K.Navi and M.Eshghi,"Optimized reversible multiplier circuit, "Journal of Circuits, Systems, and Computers, vol.18, no. 2, February 2009.
5. IEEE 754, IEEE Standard for Floating-Point Arithmetic, 2008.
6. H.Thapliyal and N.Ranganathan,"Design of reversible sequential circuits optimizing quantum cost, delay and garbage outputs," ACM Journal of Emerging Technologies in Computing, September-2010.
7. Michael Nachtigal, Nagarajan Ranganathan,"Design of single precision floating point multiplier based on operand decomposition,''10th IEEE international conference on Nanotechnology Joint symposium with Nano Korea, August 2010.
8. Mohamed Al-Ashrafy, Ashraf Salem and Wagdy Anis, "An efficient implementation of Floating Point Multiplier, 978- 1-4577-0069-9, IEEE-2011.
9. D.Maslov,Reversible logic benchmarks.http://webhome.cs.uvic.ca/dmaslov/, 2009.
10. D. Maslov and D. M. Miller. Comparison of the cost metrics for reversible and quantum logic synthesis,http://arxiv.org/abs/quant-ph/0511008, 2006.