THE INTERNATIONAL JOURNAL OF SCIENCE & TECHNOLEDGE

Design Methodology for Single Precision Floating Point Multiplier Using Reversible Logic

Madivalappa Talakal
PG Student [VLSI & ES], Department of ECE, SSIT, Tumkur, Karnataka, India
G. Jyothi
Assistant Professor, Department of ECE, SSIT, Tumkur, Karnataka, India
K. N. Muralidhara
HOD, Department of ECE, PESCE, Mandya, Karnataka, India
M. Z. Kurian
HOD, Department of ECE, SSIT, Tumkur, Karnataka, India

Abstract:
The concept describes an efficient implementation of an IEEE 754 single precision floating point multiplier using reversible logic targeted for Xilinx Virtex-4 FPGA. Reversible logic is used to reduce the power dissipation as compared to 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. Peres gate is one of the simple reversible logic gate and its quantum cost is 4. Because of its simplicity as well as lowest quantum cost, Peres gate is used to design entire single precision floating point multiplier circuit. Verilog is used to implement a technology-independent pipelined design. The proposed design can be made to handle overflow and underflow cases. Rounding is not implemented to give more precision when using the multiplier in a Multiply and Accumulate (MAC) unit. Rounding may also be implemented by truncation method for further reduction in power and area. The entire design is modeled using Verilog hardware description language .The coding is done on Xilinx ISE 12.2 and simulation is performed on Modelsim 6.3.

Keywords: *Floating point, Reversible logic gates, FPGA, Quantum computing, MAC, Truncation, MFLOPs*

1. Introduction

Since 1990 reversible logic has received great attention because of its ability to reduce the power dissipation. The low power dissipation is major requirement in low power VLSI design. Reversible logic has verity of applications in low power CMOS and in optical communication, DNA computing in medical field and other advanced technologies like quantum computation and nanotechnology. In irreversible (classical) logic computation energy dissipation is more due to the loss of information. So R.Landauer's research states that the amount of energy dissipated or heat generated for every irreversible bit operation is at least kTln2 joules, where k is the Boltzmann's constant and T is the temperature at which operation is performed. The dissipation of energy due to the one bit information loss is very small at room temperature but in case of high speed computational operations the information loss is more so the heat dissipated by such operation will be too large that it affects the entire performance and also reduces the lifetime of the components. In 1973, Bennett showed that kTln2 of energy dissipation can be reduced or eliminated from the system by making the system to reproduce the inputs from observed outputs. Due to the reproduction of inputs from outputs, reversible logic supports the process of running the system both forward and backward direction. A circuit is said to be reversible then its input vector can be recovered from its output vector and there exists a one-to-one correspondence (mapping) between input and output assignments. If computation becomes lossless then heat generation due to information loss can automatically reduces.

Reversible logic gate has equal number of inputs as well as outputs, then we call it as k X k reversible logic gate. To maintain the equality between inputs and outputs additional outputs are added but which are not used in the synthesis of a given Boolean function but to achieve reversibility they are mandatorily used such outputs are called garbage. The important design constraints are reversible logic gates do not allow fan-outs, they should maintain minimum quantum cost, the compaction or optimization of design can be achieved by making the design to produce minimum number of garbage outputs and must use minimum number of constant inputs.

Floating point numbers are one possible way of representing real numbers in binary format. The different floating point formats available in IEEE-754 standard are, Binary interchange format and Decimal interchange format. Floating point number

Multiplication is a critical requirement for complex application like DSP computations involving large dynamic range. The implementation focuses only on the single precision binary interchange format. Figure 1 shows the representation of IEEE 754 single precision binary format. It consists of a one sign bit(S), an eight bit of exponent part (E), and fraction of Twenty three bit (M or Mantissa).To form the significand 1 bit is added to the fraction. If the exponent is smaller than 255 and greater than 0, and there is 1 in the MSB of the significand then the number is said to be a normalized, in this case the real number is represented by (1)

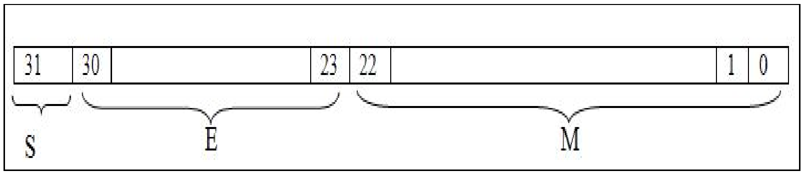


Figure 1: IEEE single precision floating point format

$Z = (-1^{s}) * 2^{(E - Bias)} * (1.M)$ (1)
Where $M = m_{22}\, 2^{-1} + m_{21}\, 2^{-2} + m_{20}\, 2^{-3} + \ldots + m_{1}\, 2^{-22} + m_{0}\, 2^{-23}$
Bias = 127.
Multiplication of two floating point number is done in three steps:

- Add exponents of both numbers then bias is subtracted from added result.
- Multiply the two significands.
- Finally sign calculation is accomplished by XORing the two numbers.

2. Related Work

To achieve reduced latency of floating –point operations large number of works has been reported in the literature. Systematic survey on different advanced techniques as well as reversible components is reported bellow.

Landauer [1] developed the formula kTln2; to calculate amount of energy needed to erase each bit of information. (Where k is the Boltzmann constant i.e. $3x10^{12}$ joule at room temperature). Therefore energy dissipation directly depends number of bits erased.

C. H. Bennett [2] in 1973 showed that the advantage of reversible logic model on lossless computations. So reversible logic concept plays a major role in low power technology, where very less power dissipation is needed. So researches on reversible concept started rapidly after 1973.

In 1994 Shor [3] proposed his research work on creating the algorithm using reversibility concept for factorizing larger number with maximum efficiency. So the pioneer work on reversible computation was started by more research fellows in different fields like low power CMOS technology and so on.

Edward Fredkin and Tommaso Toffoli [4, 5] developed new reversible gates known as Fredkin and Toffoli reversible logic gates. These gates are also called as universal gates; because of their universal nature they are used in reversible logic based circuits.

Peres [6] introduced a new gate known as pares gate. It has 3 inputs and 3 outputs (3X3 reversible logic gate). The quantum cost of Peres gate is only four. So peres gate is one of the most widely used reversible logic gate.

TR gate is also a type of reversible logic gate introduced by H Thapliyal and N Ranganathan [7]. The major objective behind this invention was to achieve fewer garbage output in a reversible logic based circuits.

Invention of IEEE-754 standard for Floating Point Arithmetic [8], intended for hardware implementation, although provisions were made for software implementation for several operations. Because of the standard format it is possible to write floating point arithmetic algorithms in such a way which could be executed on several different platforms and which would produce same results.

In 2000, the IEEE appointed a new committee to examine the IEEE-754 standard with the aim of including a decimal floating point arithmetic, including a good existing practice, providing the reproducible results, and clarifying the standard, while not invalidating and conforming the implementations of IEEE-754(1985) standard.

The concept "efficient implementation of an IEEE 754 single precision floating point multiplier" proposed by Mohamed Al-Ashrafy, Ashraf Salem, and Wagdy Anis [10]. The multiplier implementation can have an ability to handle the over/under flow cases. Rounding is not implemented to achieve more precision while using the multiplier in MAC unit. With latency of three clock cycles the design may achieves 301 MFLOPs.

Recently Himanshu Thapliyal and M.B Srinivas [11] proposed a 4 x 4 reversible logic gate called "TSG". The most significant aspect is that TSG can perform singly as a reversible full adder, after the invention of TSG gate reversible full adder can be implemented with a single gate only. The design proposed in this paper is NXN reversible multiplier using TSG gate and is based on two concepts. In this proposed design both partial products and delay will be generated in parallel and by designing reversible parallel adder from TSG logic gates addition can be reduced to log2N steps

In 2012 the concept of reversible single precision floating point multiplier (RSPFPM) proposed by M.Jenath, V.Nagarajan [13], the design includes reversible integer multiplier (24X24) based on operand decomposition approach. The objective of reversible logic is to reduce the power dissipation and also to reduce the information loss; because of low power dissipation feature which finds application in various fields like low power computation technique, quantum computation, optical computing field, and other new complex computing technologies. Among all other reversible logic gates, Peres gate is widely used to design the multiplier because of its low quantum cost.

In 2014 Madivalappa Talakal [14] proposed REA using peres gate. The most significant aspect is that REA using peres gate can gives lowest quantum cost as well as garbage output.

In 2014 April [15] he also made the brief survey on reversible logic technique as well as single precision floating point multiplier using reversible logic.

In 2014 May Madivalappa Talakal [16] again proposed 8X8 reversible multiplier using peres gates. The most significant aspect is that 8X8 reversible multiplier using peres gate can gives lowest quantum cost as well as garbage output.

3. Concept of Reversible Logic

A reversible logic gate/device which has k-input and k-output with one-to-one mapping in between inputs & outputs. So recovering the inputs from its outputs is very easy task. Direct fan-Outs are not allowed during the synthesis reversible circuits because one–to-many concept is not satisfy the reversibility property. The necessary fan outs in reversible circuits can be implemented by using additional gates. The reversible circuit utilizes very few reversible logic gates because all reversible logic gates can perform more than one operation of multiple operations. The complexity and performance of reversible logic circuit can be determined by bellow given parameters.

- Number of Reversible gates (N): The number of reversible gates needed to design circuit.
- Number of constant inputs (CI): The number of constant inputs(either 0 or 1) used to synthesize the given function.
- Number of garbage outputs (GO): These are the unused outputs present in the reversible logic circuit to achieve reversibility.
- Quantum cost calculation(QC): The quantum cost of reversible logic circuit determines the number of primitive gates present in that circuit.

Important design constraints of reversible logic circuits as given bellow.

- Reversible logic circuit does not allow direct fan-outs.
- It should maintain minimum quantum cost.
- It should maintain minimum number of garbage outputs; and it can be achieved by design optimization.
- Fewer constant inputs.
- The logic depth(less complexity)of reversible logic circuit should be very less.

3.1. Reversible Logic Gates

The power dissipation is the important factor in VLSI design. The computers in now a day to performing the logical operation dissipate the energy every time while erasing the bit of information. This happens in case of "irreversible (classical) logic". Low power dissipation is the main advantage of reversible logic because it does not allows power dissipation due to information loss. The classical gate (irreversible gate) and general kXk reversible logic gate is shown in figure 2(a) and (b) respectively. In reversible XOR gate there is no loss of information as compared to classical one. Since one-to-one mapping between input/output vectors will gives the same number of inputs and outputs.

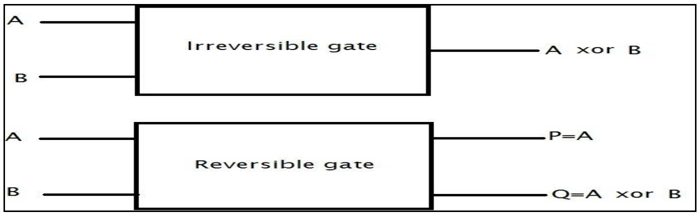


Figure 2: (a) Irreversible XOR Gate (b) Reversible XOR Gate

Not gate is the simplest reversible logic gate and it is also called as 1x1 gate because it has only 1-input and 1-output. An example for a 2X2 reversible logic gate is controlled Not gate. Every reversible logic gate has a cost associated with it and this depends on number of primitive gates used to design it called quantum cost. The quantum cost of 1X1 reversible logic gates is 0, and quantum cost of 2X2 reversible logic gates is 1. Any reversible logic gate is realized by using 1X1 Not gates and 2X2 reversible gates, such as V and V+ (V is square root and V+ is hermitian of Not gate) and Feynman gate which is also called as controlled Not gate (CNOT).

The V and V+ quantum gates have the property given bellow:

V X V = NOT

V X V+ = V+ X V = I

V+ X V+ = NOT

Except few cases the quantum cost of each reversible logic gate can be calculated by counting the number V, V+ and CNOT gates used in implementing it.

3.2. NOT Gate

This is also called 1X1 reversible gate. The output for Not gate is simply compliment of its input. The quantum cost of Not gate is 0.

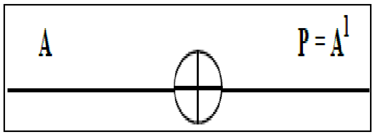


Figure 3: NOT Gate

3.3. FEYNMAN Gate

Feynman gate is also called as 2X2 reversible gate . The input and output vectors for Feynman gate is I (A, B) and O (P, Q) respectively. The outputs are defined as P=A, Q=A xor B. Quantum cost of a Feynman gate is one. It is most widely used in several circuits because it has low quantum cost.

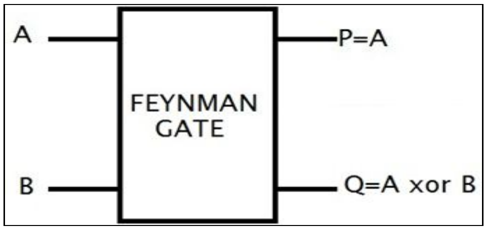


Figure 4: 2X2 FEYNMAN gate

3.4. PERES Gate

It is also called as 3X3 reversible gate. The input and out put vector for peres gate is I (A, B, C) and O (P, Q, R) respectively. The output is P = A, Q = A xor B and R=AB xor C. Quantum cost of Peres gate is 4. Because of its lowest quantum cost in many designs Peres gate is used. Single Peres gate is enough to design half adder.

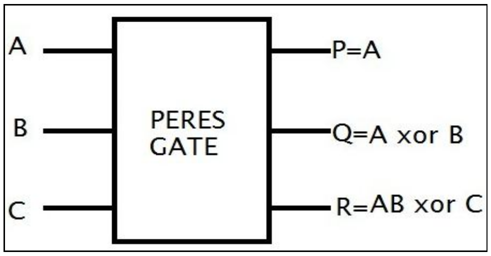


Figure 5: 3x3 PERES Gate

4. Proposed Methodology

The figure 6 represents the block diagram of reversible single precision floating point multiplier (RSPFPM).Peres gate is used to design entire structure is modelled using Verilog code. The coding is done on Xilinx ISE 12.2 for simulation purpose the Model sim 6.3 has been used. So the detailed explanation about the design methodology of RSPFPM is given in following sections.

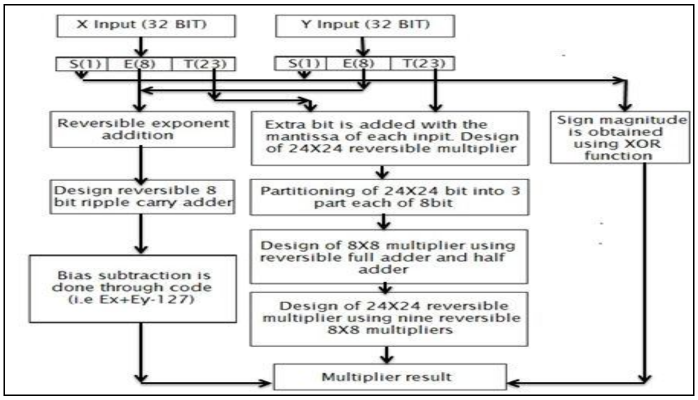

X Input (32 BIT)
Y Input (32 BIT)
S(1) E(8) T(23)
S(1) E(8) T(23)
Reversible exponent addition
Extra bit is added with the mantissa of each inpit. Design of 24X24 reversible multiplier
Sign magnitude is obtained using XOR function
Design reversible 8 bit ripple carry adder
Partitioning of 24X24 bit into 3 part each of 8bit
Design of 8X8 multiplier using reversible full adder and half adder
Bias subtraction is done through code (i.e Ex+Ey-127)
Design of 24X24 reversible multiplier using nine reversible 8X8 multipliers
Multiplier result

Figure 6: Block diagram of reversible single precision floating point multiplier

4.1. Sign Bit Calculation

The XOR function of X and Y gives the proper sign of multiplied product. The XOR function is implemented in the design by using single Peres gate.

4.2. Exponent Adder

The exponents are added by using 8X8 reversible ripple carry adder and the bias value -127 is subtracted from exponents result. The basic building block in the ripple carry adder is full adder, and most other adder circuits. So in this proposed paper single peres gate is used to design half adder and using 2 half adders the necessary full adder is designed. So peres gate is basic building block of proposed design and it has an ability to perform all necessary operations. The full adder has 3 inputs Xi,Yi and Ci and computes the sum bit Si and carry bit Ci+1. The equations for sum and carry output bits are given bellow.

$$\text{Sum (i)} = Xi \oplus Yi \oplus Ci \qquad (2)$$

$$Ci+1= XiYi + XiCi + YiCi \qquad (3)$$

Reversible ripple carry adder An 8-bit reversible ripple carry adder is used to add the two input exponents. As shown in Figure 7, a ripple carry adder is a chain of cascaded full adders and one half adder. Each full adder has 3 inputs (A, B, Ci) and 2 outputs (S, Co). The carry out (Co) of each adder is fed to the next full adder(ripple the carry from previous stage to next).

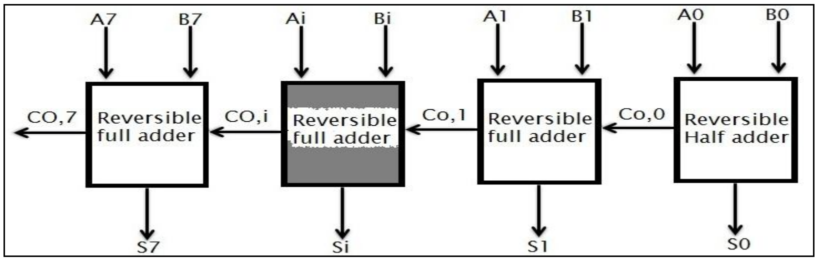
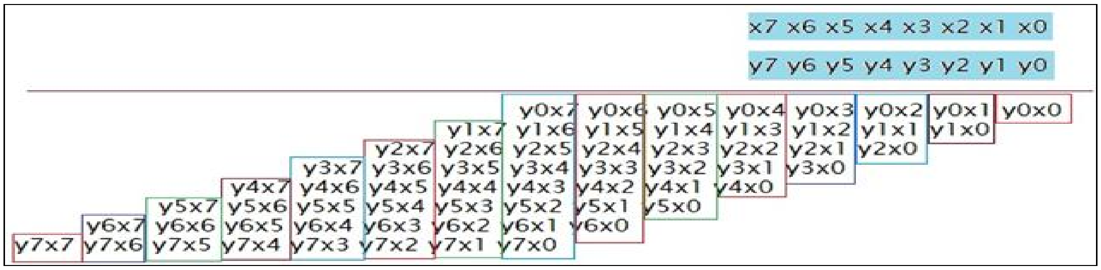


Figure 7: Reversible Ripple Carry adder

The addition process produces an 8 bit sum (S7 to S0) and a carry bit (Co, 7). Bias is subtracted from 9 bit addition result (S8 to S0). To minimize the circuit complexity as well as area required the operation of Bias subtraction is directly obtained from the Verilog code; this eliminates the use of subtractor. So in this way total efficiency of final system will increases in terms of amount of power required as well as number of reversible logic gates (peres) used.

4.3. Reversible Multiplier

The multiplier portion in the above figure 6 is a reversible 24X24 multiplier and to reduce the complexity initially it is divided into 3-8X8 reversible multipliers. 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. Simple 8X8 multiplier operation is shown in figure 8. The Figure 5 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 figure 9. Peres gate provide the necessary AND operator when their input C is hardwired to 0 and also Peres gates reduces the quantum cost. In the summation stage of multiplier Peres gate is used because a cascade of two Peres gate can generate the full adder operation. This design realizes a lower quantum cost and fewer garbage outputs by virtue of proposed design of reversible full adder and half adder. So the entire multiplier part designed and implemented using Verilog code. The coding is done on Xilinx ISE 12.2 for simulation purpose the Model sim 6.3 has been used.



x7 x6 x5 x4 x3 x2 x1 x0
y7 y6 y5 y4 y3 y2 y1 y0
y0x7 y0x6 y0x5 y0x4 y0x3 y0x2 y0x1 y0x0
y1x7 y1x6 y1x5 y1x4 y1x3 y1x2 y1x1 y1x0
y2x7 y2x6 y2x5 y2x4 y2x3 y2x2 y2x1 y2x0
y3x7 y3x6 y3x5 y3x4 y3x3 y3x2 y3x1 y3x0
y4x7 y4x6 y4x5 y4x4 y4x3 y4x2 y4x1 y4x0
y5x7 y5x6 y5x5 y5x4 y5x3 y5x2 y5x1 y5x0
y6x7 y6x6 y6x5 y6x4 y6x3 y6x2 y6x1 y6x0
y7x7 y7x6 y7x5 y7x4 y7x3 y7x2 y7x1 y7x0

Figure 8: 16 bit output in the reversible 8X8 multiplier

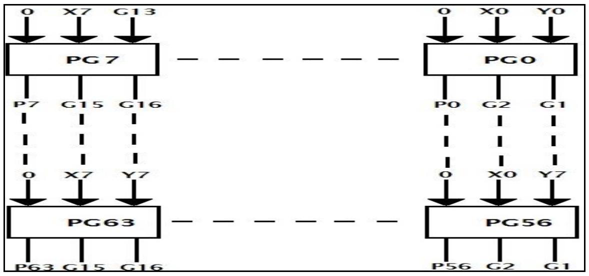


Figure 9: Generation of 64 partial products

5. Simulation and Results

The entire design of single precision floating point multiplier using reversible logic is modelled using (Verilog) HDL .The Xilinx ISE 12.2 is used for coding. The Modelsim 6.3is used for simulation purpose. The simulation result for proposed multiplier is shown in Figure 11. Finally the proposed multiplier will be efficient in terms of number of reversible gates and garbage output. Reduction in number of gates reduces the complexity of circuit. Table.1device utilization summary confirms that the proposed design uses very few slices (165), LUTs (288) and bonded IOBs (128).

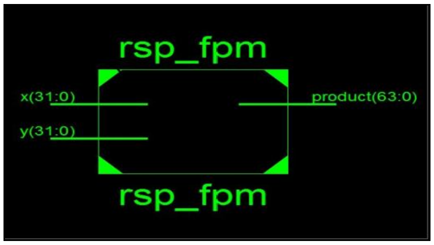
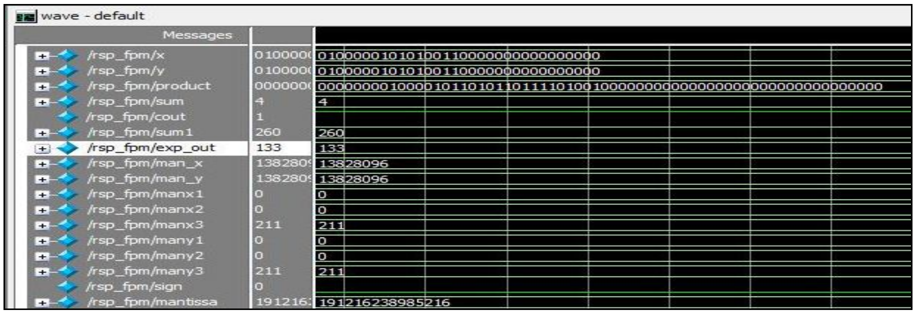


Figure 10: RTL schematic view of RSPFPM



```
/rsp_fpm/x          0100000  01000001010100110000000000000000
/rsp_fpm/y          0100000  01000001010100110000000000000000
/rsp_fpm/product    0000000  0000000010000101101011011110100100000000000000000000000000000000
/rsp_fpm/sum        4        4
/rsp_fpm/cout       1
/rsp_fpm/sum1       260      260
/rsp_fpm/exp_out    133      133
/rsp_fpm/man_x      138280   13828096
/rsp_fpm/man_y      138280   13828096
/rsp_fpm/manx1      0        0
/rsp_fpm/manx2      0        0
/rsp_fpm/manx3      211      211
/rsp_fpm/many1      0        0
/rsp_fpm/many2      0        0
/rsp_fpm/many3      211      211
/rsp_fpm/sign       0
/rsp_fpm/mantissa   191216   191216238985216
```

Figure 11: Simulation result of RSPFPM

Logic utilization	Used	Available	Utilization
Number of slices	165	5472	3%
Number of 4 input LUTs	288	10944	2%
Number of bonded IOBs	128	320	40%

Table 1: Device utilization summary

6. Applications

Because of low power dissipation reversible concept has wide range of applications some of them are given bellow:

- Low power CMOS VLSI design.
- Quantum computers.
- Nanotechnology and microelectronics.
- Optical computation.
- Design of efficient low power arithmetic for digital signal processing systems (DSP).

7. Conclusion

This paper presents a design methodology of a single precision floating point multiplier using reversible logic that supports the IEEE 754 binary interchange format. Basic reversible gates presented in this paper can be used in regular circuits realizing Boolean functions. The reversible exponent addition using reversible ripple carry adder, 8X8 reversible multiplier and sign bit generator are designed by using peres gate and modelled by Verilog hardware description language. The proposed multiplier uses such parts to accomplish its reversible multiplication operation and is efficient in terms of number of reversible gates and garbage output and also in terms of quantum cost. Reduction in number of gates to reduce complexity of circuit. Finally by observing table.1 it confirms that the proposed design uses very few slices (165), LUTs (288) and bonded IOBs (128). Reversible computing may have long-term benefit very well in those areas which require high energy efficiency, speed and performance. In future the proposed work may be implemented by using different reversible logic gates in order to improve the performance compared to proposed one.

8. Acknowledgment

Authors would like to thank Mrs. G Jyothi for her invaluable support and contribution.

9. References

1. Landauer .R, "Irreversibility and heat generation in the computing process", IBM J. Research and Development, pp. 183-191, 1961.
2. Bennett C.H., "Logical reversibility of Computation", IBM J.Research and Development, pp. 525-532, 1973.
3. P. Shor, Algorithms for quantum computation: discrete log and factoring, Proc. 35th Annual Symp. On Found. Of Computer Science (1994), IEEE Computer Society, Los Alamitos, 124-34.
4. T. Toffoli., Reversible Computing, Tech memo MIT/LCS/TM-151, MIT Lab for Computer Science (1980).
5. E. Fredkin and T. Toffoli, Conservative logic, Int. J. Theor.Phys., vol. 21, no. 3/4, pp.219 -253 1982.
6. Peres, Reversible Logic and Quantum Computers, Physical Review A, vol. 32, pp. 3266-3276, 1985.

7. H Thapliyal and N Ranganathan, Design of Efficient Reversible Binary Subtractors Based on a New Reversible Gate, IEEE Proceedings of the Computer Society Annual Symposium on VLSI, pp. 229-234 (2009).
8. Peter Markstein Woodside, CA 94062, USA "The New IEEE-754 Standard for Floating-Point Arithmetic."
9. IEEE 754-2008, IEEE Standard for Floating-Point Arithmetic, 2008.
10. Mohamed Al-Ashrafy, Ashraf Salem, WagdyAnis "An Efficient Implementation of Floating Point Multiplier."
11. H Thapliyal and M B Srinivas, 2006. Novel Reversible Multiplier Architecture using Reversible TSG gate. IEEE International Conference on Computer Systems and Applications, pp. 110-103.
12. Michael Nachtigal, Nagarajan,Ranganathan,"Design of a reversible single precision floating point multiplier based on operand decomposition,‵‶‵10thIEEE international conference on Nanotechnology Joint symposium with Nano Korea, August 2010.
13. M.Jenath, V.Nagarajan "FPGA Implementation On Reversible Floating Point Multiplier" International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.
14. Madivalappa Talakal, G Jyothi, K.N.Muralidhara, M.Z.Kurian "Design and Implementation of REA for Single Precision Floating Point Multiplier Using Reversible Logic" ISSN: 2278-8875, Volume-3, Issue-3, March 2014.
15. Madivalappa Talakal, G Jyothi, K.N.Muralidhara, M.Z.Kurian "A survey on Single Precision Floating Point Multiplier Using Reversible Logic technique" April 2014.
16. Madivalappa Talakal, G Jyothi, K.N.Muralidhara, M.Z.Kurian "Design and Implementation of 8X8 Multiplier for Single Precision Floating Point Multiplier Using Reversible Logic" May 2014