ISSN: 2348-1358

A Novel Carry Skip BCD Adder Implementation Using Reversible Logic for Nanotechnology-Based Systems

B. Vijaya Nirmala, Assistant Professor, Department of Physics, Dharmavant College of Science and Commerce, Yakuthpura, Hyderabad

ABSTRACT

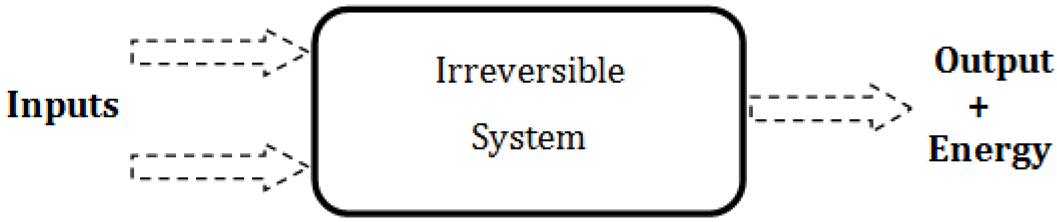
In recent years, reversible logic has emerged as a promising technology with applications spanning low-power CMOS, quantum computing, nanotechnology, and optical computing. Quantum computing, in particular, relies on reversible logic for its realization. The construction of low-power and low-loss computational structures is crucial for developing arithmetic circuits used in quantum computation, nanotechnology, and other low-power digital circuits. This work presents optimized one-digit BCD adders and carries skip BCD adders using new reversible logic gates. The proposed approach aims to minimize quantum cost and optimize performance compared to existing circuits, focusing on reducing the number of reversible gates used and minimizing garbage outputs produced. The objective is to design faster BCD adders and provide a foundation for building the decimal Arithmetic Logic Unit (ALU) of a Quantum CPU. The work is implemented using Hardware Description Language (HDL), with Simulation & Synthesis conducted on Xilinx ISE. The design targets the Xilinx Spartan 3E FPGA device. This research contributes to the advancement of efficient and high-performance arithmetic circuits suitable for emerging technologies such as quantum computing and nanotechnology.

Key Words: CMOS, quantum computing, nano technology, BCD adder etc.

I. INTRODUCTION

The development of computing machines found great success in the last decades. But the ongoing miniaturization of integrated circuits will reach its limits in the near future. Shrinking transistor sizes and power dissipation are the major barriers in the development of smaller and more powerful circuits. Reversible logic provides an alternative that may overcome many of these problems in the future. For low-power design, reversible logic offers significant advantages since zero power dissipation will only be possible if computation is reversible. Furthermore, quantum computation profits from enhancements in this area, because every quantum circuit is inherently reversible and thus requires reversible descriptions.

In most computing tasks, the number of output bits is relatively small compared to the number of input bits. For example, in a decision problem, the output is only one bit (yes or no) and the input can be as large as desired. However, computational tasks in digital signal processing, communication, computer graphics, and cryptography require that all of the information encoded in the input be preserved in the output. Some of those tasks are important enough to justify adding new microprocessor instructions to the HP PA-RISC (MAX and MAX-2), Sun SPARC (VIS), and PowerPC (AltiVec) instruction sets. In particular and new bit-permutation instructions were shown to vastly improve performance of several standard algorithms, including matrix transposition and DES, as well as two recent cryptographic algorithms Twofish and Serpent. Bit permutations are a special case of reversible functions, that is, functions that permute the set of possible input values. It is a key element of Fast Fourier Transform algorithms and has been used in application-specific Xtensa processors from Tensilica. One might expect to get further speed-ups by adding instructions to allow computation of an arbitrary reversible function. The problem of chaining such instructions together provides one motivation for studying reversible computation and reversible logic circuits, that is, logic circuits composed of gates computing reversible functions.



ISSN: 2348-1358

Fig1. An Irreversible System

Currently, energy losses due to irreversibility are dwarfed by the overall power dissipation, but this may change if power dissipation improves. In particular, reversibility is important for nanotechnologies where switching devices with gain are difficult to build.

II. LITERATURE SURVEY

Reversible Logic for Low-Power Design

Power dissipation and therewith heat generation is a serious problem for today's computer chips. A significant part of energy dissipation is due to the non-ideal behaviors of transistors and materials. Here, higher levels of integration and new fabrication processes reduced the heat generation in the last decade. However, a more fundamental reason for power dissipation arises from the observations made by Landauer in 1961. Landauer proved that using traditional (irreversible) logic; gates always lead to energy dissipation regardless of the underlying technology. More precisely, exactly k * T * log 2 Joule of energy is dissipated for each "lost" bit of information during the irreversible operation (where k is the Boltzmann constant and T is the temperature). While this amount of power currently does not sound significant, it may become crucial additionally considering that (1) today millions of operations are performed in some seconds (i.e. increasing processor frequency multiplies this amount) and (2) more and more operations are performed with smaller and smaller transistor sizes (i.e. in a smaller area).

In contrast, Bennett showed that energy dissipation is reduced or even eliminated if computation becomes information-lossless. This holds for reversible logic, since data is bijectively transformed without losing any of the original information. Bennett proved that circuits with zero power dissipation are only possible if they are built from reversible gates. In 2002, first reversible circuits have been built that exploit this observation. In fact, these circuits were powered by their input signals only (i.e. without additional power supplies). In the future, such circuits may be an alternative that can cope with the heat generation problem of traditional chips. Furthermore, since reversible circuits already work with low power, applications are also possible in domains where power is a limited resource (e.g. for mobile computation).

- **Reversible Logic as a Basis for Quantum Computation**

Quantum circuits offer a new kind of computation. Instead of logic signals 0 and 1, quantum circuits make use of qubits. A qubit is a two level quantum system, described by a two dimensional complex Hilbert space. The resulting tools can be obtained under www.revkit.org. This allows representing not only 0 and 1 but also a superposition of both. As a result, qubits may represent multiple states at the same time enabling enormous speed-ups in computations. For example, it has been shown that using a quantum circuit it is possible to solve the factorization problem in polynomial time, while for traditional circuits only exponential methods exist.

But, research in the area of quantum circuits is still at the beginning. Nevertheless, first promising results exist: At the University of Innsbruck one of the first quantum circuits consisting of 8 qubits was built in 2005. This has been further improved so that today circuits with dozens of qubits exist—with upward trend. Even first commercial realizations of quantum circuits (e.g. a random number generator) are available. Reversible logic is important in this area because every quantum operation is inherently reversible. Thus, progress in the domain of reversible logic can be directly applied to quantum logic.

III. REVERSIBLE GATES AND CIRCUITS

In conventional (irreversible) circuit synthesis, one typically starts with a universal gate library and some specification of a Boolean function. The goal is to find a logic circuit that implements the Boolean function and minimizes a given cost metric, e.g., the number of gates or the circuit depth. At a high level, reversible circuit synthesis is just a special case in which no fan-out is allowed and all gates must be reversible.

Reversible functions can be realized by reversible logic. Due to its special properties, reversible logic found large interest in several domains like low-power design or quantum computation. As a result, synthesis of reversible functions has become an intensively studied topic in the last years. Therefore, new kinds of circuits have been proposed that are introduced and compared to traditional circuits.

Reversible circuits

A circuit realizes a Boolean function. Usually, a circuit is composed of signal lines and a set of basic gates (called gate library). For traditional circuits, often the gate library depicted in Fig.2 is used. This includes gates for the operations AND, OR, and NOT, based on which any Boolean function can be realized. Furthermore, fan-outs are applied to use signal values more than once.

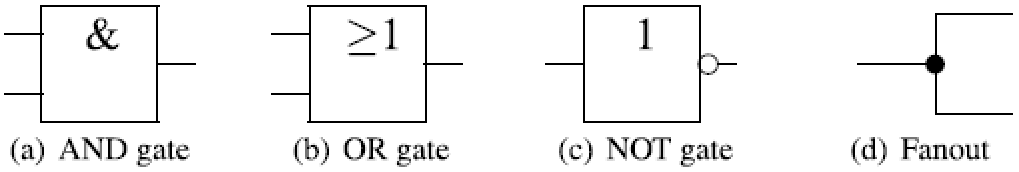


Fig 2. Traditional circuit elements

In contrast, to realize reversible logic some restrictions must be considered: fan-outs and feedback are not directly allowed, since they would destroy the reversibility of the computation. Also, the gate library from above as well as the traditional design flow cannot be utilized. As a result, a cascade structure over reversible gates is the established model to realize reversible logic.

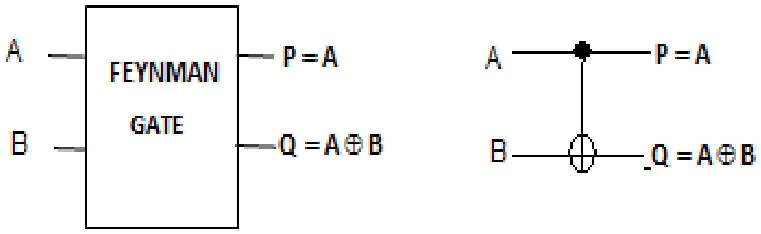


Fig 3. Feynman gate (FG) and Quantum implementation of FG

In above fig: 3 A ● is used to indicate a control line, while a ⊕ (×) is used for denoting the target line of a respective reversible gate.

The block diagram for 3*3 TG is shown in Fig.4. TG plays an important role in the reversible logic synthesis. It is also used in the design of any Boolean function and hence it can be considered as a universal reversible gate.

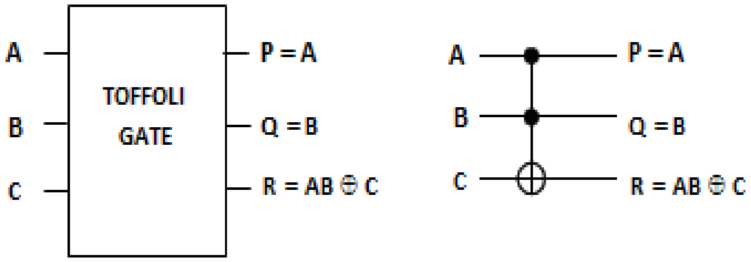


Fig .4 Toffoli gate (TG) and Quantum implementation of TG.

IV CONVENTIONAL CS BCD ADDERS

Conventional carry skip BCD adder

The proposed Carry Skip BCD Adder is being constructed in such a way that, the first full adder block consisting of 4 full adders can generate the output carry 'Cout' instantaneously, depending on the input signal and 'Cin', without waiting for the carry to be propagated in the ripple carry fashion. Fig.5 shows the proposed Carry Skip BCD adder.

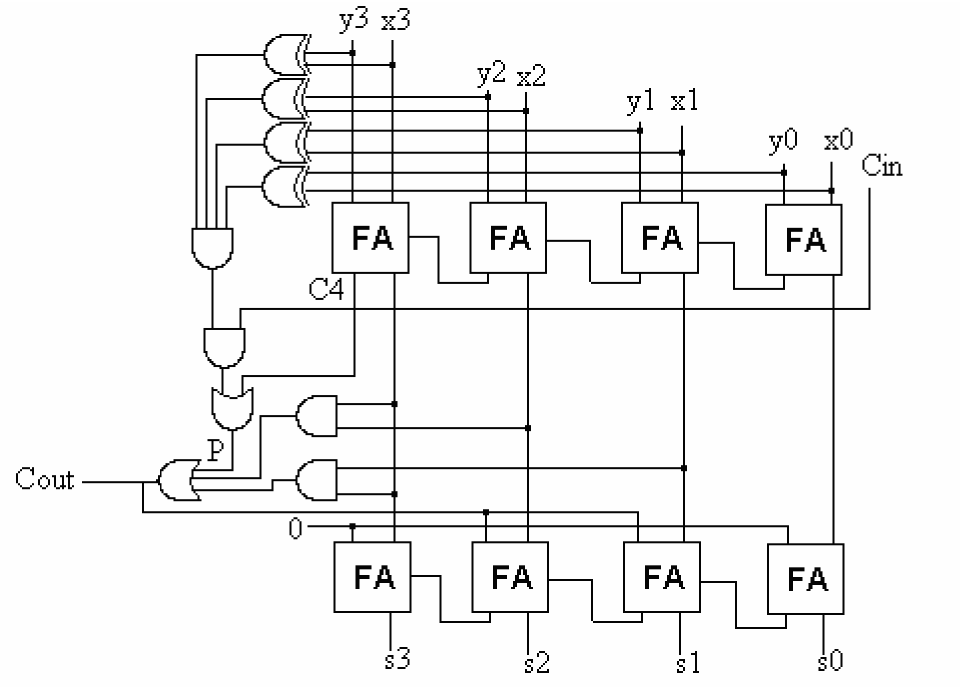


Fig 5. Conventional carry skip BCD adder

Proposed CS BCD Adder

For reversible BCD adder, one 4-bit parallel adder is used for binary addition of the numbers, a combinational circuit is used for detection of BCD overflow and another 4-bit parallel adder is used for error correction. For Reversible Carry Skip BCD adder, carry skip logic is incorporated for faster carry generation, which is used in the overflow detection logic. Other basic components are same as reversible BCD adder.

The circuit can be divided into three blocks.

1. Adder-1 and XOR-AND4 block which outputs C_{in}, C_4 , Z along with S_3, S_2, S_1 and Σ_0 shown in fig.6 and is represented in this paper as a single block shown in fig.5.12.
2. AND-OR and Correction logic circuit block which outputs $P= ZC_{in}+C_4$ and $C_{out} = P \oplus S_3 (S_2 + S_1)$ shown in fig.7.
3. Adder-2 block which adds 110 to S_3, S_2 and S_1 whenever Cout = 1 and outputs the corrected BCD $\Sigma_3\Sigma_2\Sigma_1\Sigma_0$ shown in fig.8.

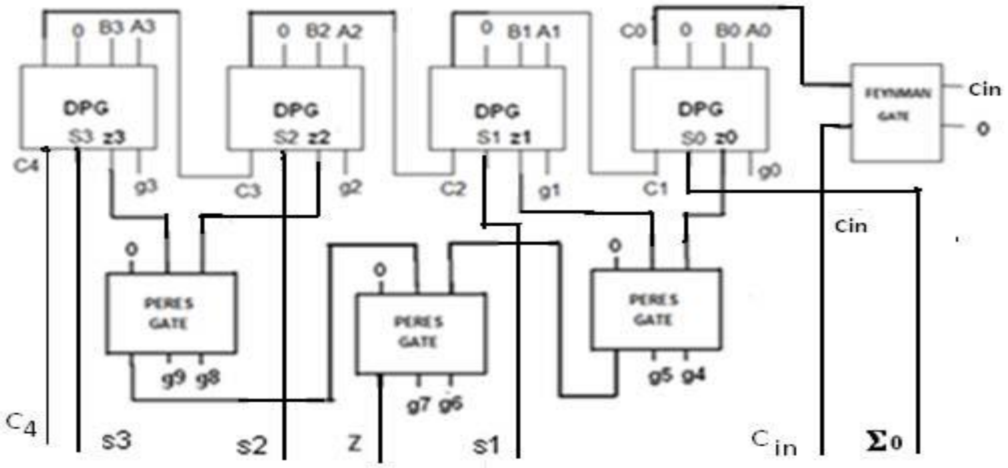


Fig.6 Design of Adder-1 and XOR-AND4 Block.

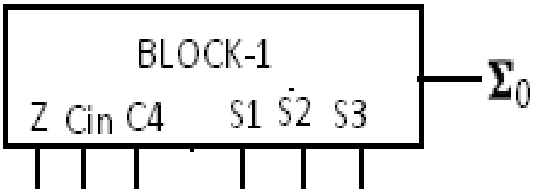


Fig.7. Design of Adder-1 and XOR-AND4 block.

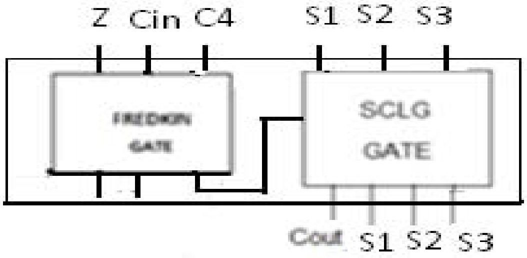


Fig.8. Design of AND-OR and Correction logic circuit block.

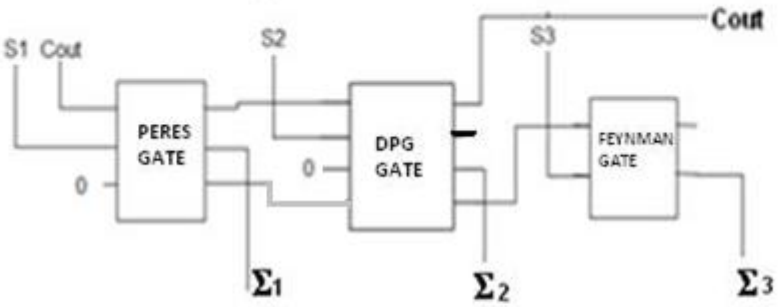


Fig.8. Design of Adder-2 block.

The proposed optimized circuit of a Carry skip BCD adder is as shown in fig.9.

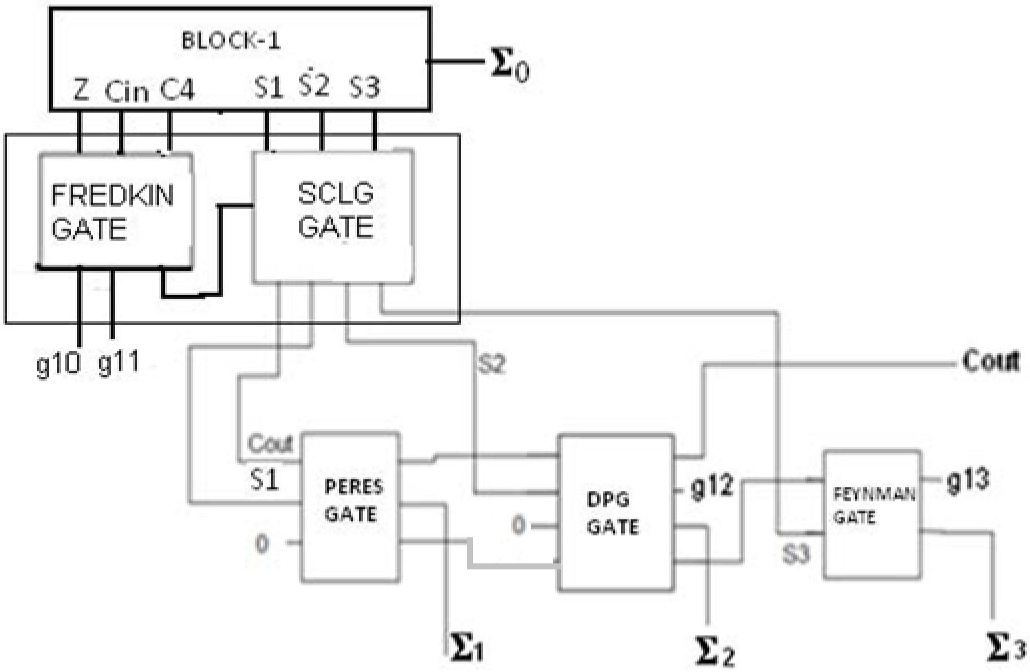


Fig.9. An Optimized Carry skip BCD adder circuit (Proposed design).

V. RESULTS AND DISCUSSION

BCD Adder

Several researchers have proposed the 4 bit parallel adder which is constructed using different reversible full adder gates [14-17].

In 3 New gates [NG] for the correction logic circuit and 8 TSG gates for the adders are used for the construction of reversible implementation of BCD adder. This reduces the number of gates but in this paper fan-out is not taken into account which when considered will increase the number of gates more than 11. This produces 22 garbage outputs with 11 constant inputs.

In [15] BCD adder is constructed using 23 reversible gates. One bit Full adder cell is realized using one NG and one PG gate which produce 2 garbage outputs. So a 4bit adder produces 8 garbage outputs and requires 8 reversible logic gates. The six correction logic is obtained using 3 TG with six garbage outputs. Also, it uses 4 FG for fan-out purpose. So the circuit of [15] uses a total of 23 reversible logic gates with 22 garbage outputs.

In [16] BCD adder is constructed using 23 reversible gates. One bit Full adder cell is realized using one NG and one NTG gate which produce 2 garbage outputs. So a 4bit adder produces 8 garbage outputs and requires 8 3*3 reversible logic gates. The six correction logic is obtained using three NG with six garbage outputs. Also it uses 4 FG far fan-out purpose. The circuit of [16] uses a total of 23 reversible logic gates with 22 garbage outputs and requires 17 constant inputs.

In [17] the BCD adder is realized using 8 HNGs along with one HNFG and one FG for fan-outs. It also uses 2 NGs, one TG and one FG for the implementation of the correction logic. This uses a total of 14 reversible gates and it produces 22 garbage outputs with 17 constant inputs in the complete circuit.

The implementation given in [18] uses two Fredkin gates and one Toffoli gate for the correction logic and also it uses a 4 bit parallel adder constructed using four TSGs. It also uses a combination of one FG, one PG and a TSG for the adder which adds the c_{out} to the sum in order to generate the final BCD sum. This implementation requires a total of 10 reversible logic gates and it produces a total number of 11 garbage outputs (including the garbage from the correction circuit) with 7 constant inputs.

The implementation given in [19] uses two Fredkin gates and one Toffoli gate for the correction logic and also it uses a 4bit parallel adder constructed using four MTSGs. It also uses a combination of one FG, one PG and a MTSG for the adder which adds the c_{out} to the sum in order to generate the final BCD sum. This implementation requires a total of 10 reversible logic gates and it produces a total number of 11 garbage outputs (including the garbage from the correction circuit) with 7 constant inputs. The advantage of using this design over that given in [18] is that the quantum cost of MTSG is less than the quantum cost of TSG [19].

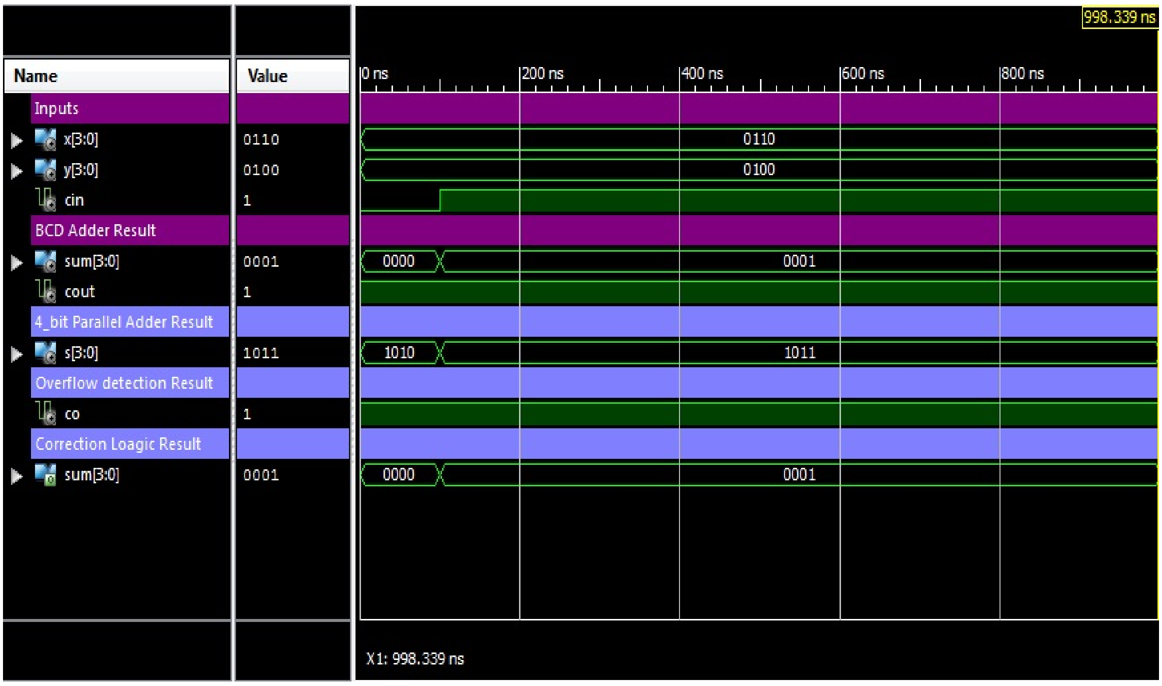
The proposed design uses a DPG gate whose quantum cost is same as MTSG. However, the numbers of reversible logic gates used in the proposed design are only 8 with number of garbage outputs 10 which is minimum as compared to the some of the designs. So, the proposed design is a much-optimized circuit. It uses a new gate specially designed for the correction logic circuit of a one-bit BCD adder [12].

The comparative studies of various designs of carry skip BCD adders are presented in Table..1.

BCD Adder	No. of gates	No. of garbages	Constant inputs	Delay
Paper (14) without fan-out	15	27	15	12
Paper (18)	15	14	11	10
Paper (19)	15	14	11	10
Proposed Design	**13**	**14**	**10**	**10**

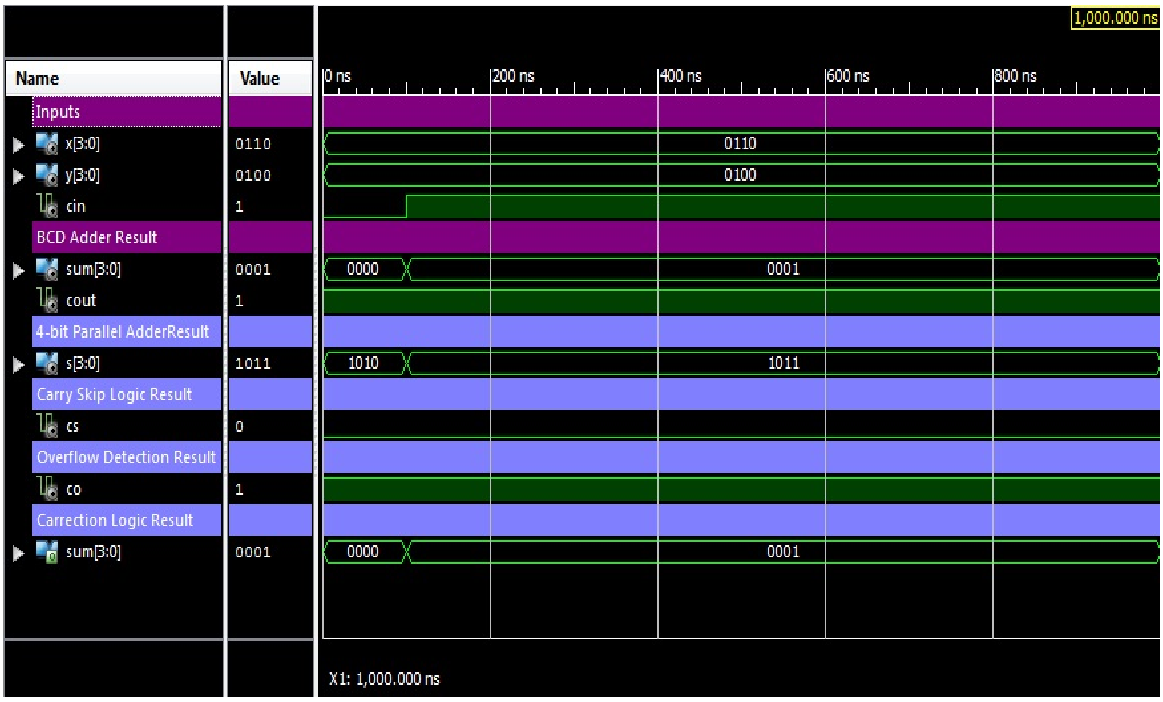
Table.1: Comparison of Carry skip BCD adders.

Simulation Result for BCD adder [14]



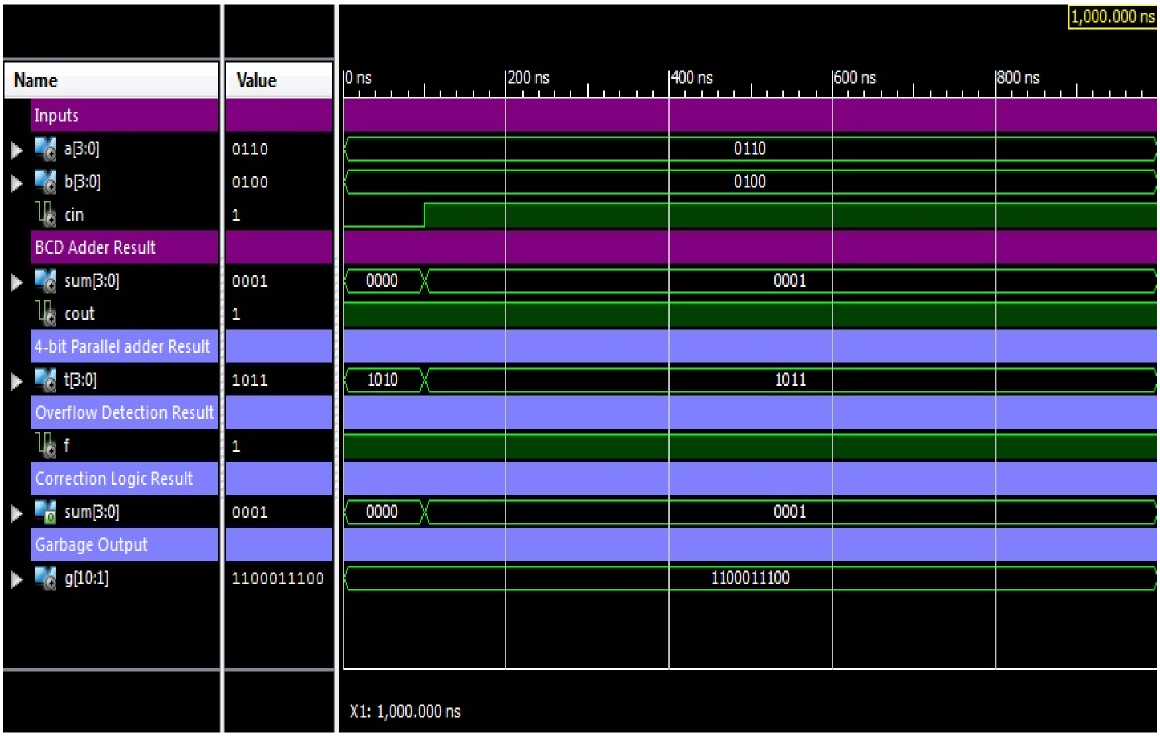
Name	Value
Inputs	
x[3:0]	0110
y[3:0]	0100
cin	1
BCD Adder Result	
sum[3:0]	0001
cout	1
4_bit Parallel Adder Result	
s[3:0]	1011
Overflow detection Result	
co	1
Correction Loagic Result	
sum[3:0]	0001

Simulation Result for CS BCD adder [14]



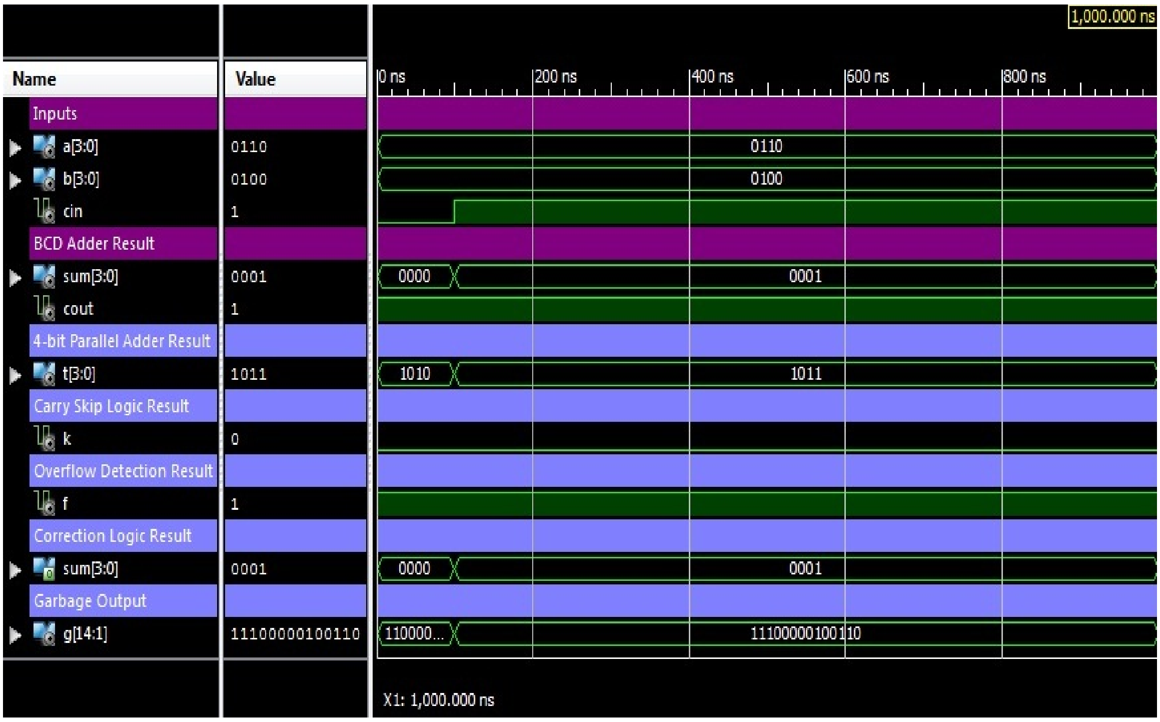
Name	Value
Inputs	
x[3:0]	0110
y[3:0]	0100
cin	1
BCD Adder Result	
sum[3:0]	0001
cout	1
4-bit Parallel AdderResult	
s[3:0]	1011
Carry Skip Logic Result	
cs	0
Overflow Detection Result	
co	1
Carrection Logic Result	
sum[3:0]	0001

Simulation Result for BCD adder [19]



Name	Value
Inputs	
a[3:0]	0110
b[3:0]	0100
cin	1
BCD Adder Result	
sum[3:0]	0001
cout	1
4-bit Parallel adder Result	
t[3:0]	1011
Overflow Detection Result	
f	1
Correction Logic Result	
sum[3:0]	0001
Garbage Output	
g[10:1]	1100011100

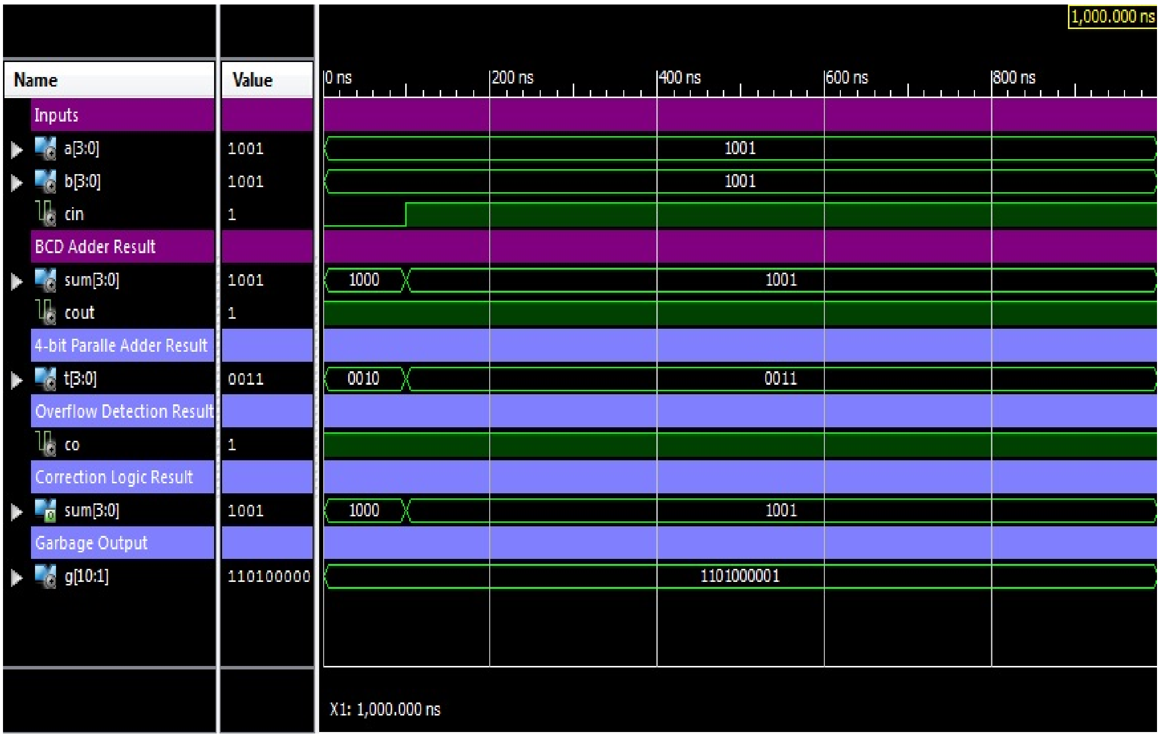
Simulation Result for CS BCD adder [19]



Name	Value
Inputs	
a[3:0]	0110
b[3:0]	0100
cin	1
BCD Adder Result	
sum[3:0]	0001
cout	1
4-bit Parallel Adder Result	
t[3:0]	1011
Carry Skip Logic Result	
k	0
Overflow Detection Result	
f	1
Correction Logic Result	
sum[3:0]	0001
Garbage Output	
g[14:1]	11100000100110

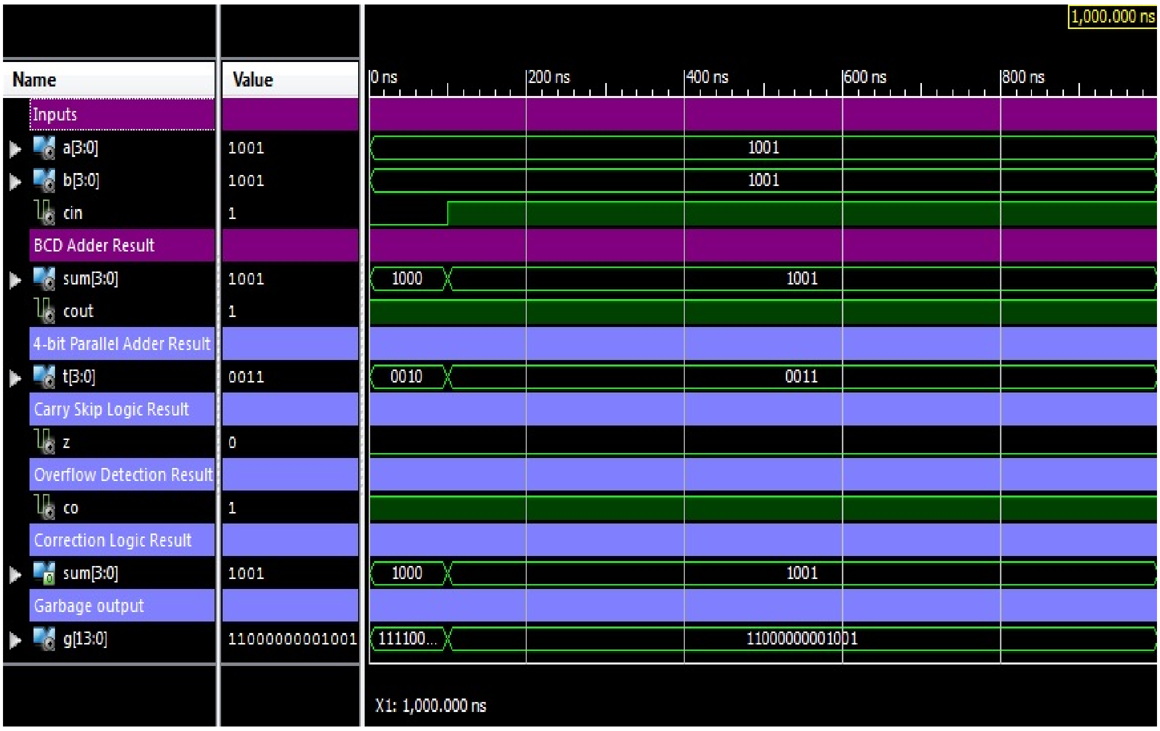
ISSN: 2348-1358

Simulation Result for BCD adder [Proposed design]



Name	Value
Inputs	
a[3:0]	1001
b[3:0]	1001
cin	1
BCD Adder Result	
sum[3:0]	1001
cout	1
4-bit Paralle Adder Result	
t[3:0]	0011
Overflow Detection Result	
co	1
Correction Logic Result	
sum[3:0]	1001
Garbage Output	
g[10:1]	110100000

Simulation Result for CS BCD adder [Proposed design]



Name	Value
Inputs	
a[3:0]	1001
b[3:0]	1001
cin	1
BCD Adder Result	
sum[3:0]	1001
cout	1
4-bit Parallel Adder Result	
t[3:0]	0011
Carry Skip Logic Result	
z	0
Overflow Detection Result	
co	1
Correction Logic Result	
sum[3:0]	1001
Garbage output	
g[13:0]	11000000001001

CONCLUSION

In this work, an optimized one digit BCD adder and an optimized one digit carry skip BCD adder are presented. The design is very useful for future computing techniques like ultra low power digital circuits and quantum computers. It is shown that the proposal is highly optimized in terms of the number of reversible logic gates, the number of garbage outputs and the delay involved.

This delay is used in calculating the delay involved in an N-digit BCD adder. The delays involved in various are calculated on the similar lines. This delay analysis results in only approximate values as there are different types of gates used in the circuits of various papers.

REFERENCES

[1] R. Landauer, "*Irreversibility and Heat Generation in the Computational Process*", IBM Journal of 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] R. Feynman, "*Quantum Mechanical Computers*," Optics News, Vol.11, pp. 11–20, 1985.

[4] T. Toffoli., "*Reversible Computing*", *Tech memo MIT/LCS/TM-151*, MIT Lab for Computer Science 1980.

[5] A. Peres, "*Reversible logic and quantum computers"*, Physical Review A 32 (6) (1985) 3266–3276.

[6] E. Fredkin, E. Toffoli, "*Conservative logic"*, Int. J. Theor. Phys. 21 (1983) 219–253.

[7] M. Mohammadi and M. Eshghi.On figures of merit in reversible and quantum logic designs."*Quantum Information Processing"*, 8(4):297–318, Aug. 2009.

[8] MajidMohammadi, Mohammad Eshghi, MajidHaghparast and Abbas Bahrololoom, (2008) "*Design and Optimization of Reversible BCD Adder/Subtractor Circuit for Quantum*

and Nanotechnology Based Systems", *World Applied Sciences Journal*, vol. 4, no. 6, pp. 787-792.

[9] W. N. N. Hung, X. Song, G. Yang, J. Yang, and M.Perkowski, "*Optimal Synthesis of Multiple Output Boolean Functions Using a Set of Quantum Gates by Symbolic Reachability Analysis*", IEEE Trans.Computer-Aided Design, Vol. 25, No. 9, pp.1652-1663, Sep 2006

[10] J. A. Smolin, D. P. Di Vincenzo, "*Five Two-Bit Quantum Gates are Sufficient to Implement the Quantum Fredkin Gate*," Physical Review A, 53, 1996, pp. 2855-2856.

[11] M.S. Islam et al., "*Low cost quantum realization of reversible multiplier circuit*", Information technology journal, 8 (2009) 208.

[12] HimanshuThapliyal and M.B Srinivas, "*A Novel Reversible TSG Gate and Its Application for Designing Reversible Carry Look-Ahead and Other Adder Architectures*", Accepted in Tenth Asia-Pacific Computer Systems Architecture Conference (ACSAC05), Singapore, October 24 - 26, 2005.