A New Look at Reversible Logic Implementation of Decimal Adder

Rekha K. James, Shahana T. K, K. Poulose Jacob
Cochin University of Science and Technology
Kochi, Kerala, India;
{rekhajames, shahanatk, kpj}@cusat.ac.in

Sreela Sasi
Gannon University
Erie, PA, USA
sasi001@gannon.edu

Abstract - **Reversibility plays a fundamental role when computations with minimal energy dissipation are considered. In recent years, reversible logic has emerged as one of the most important approaches for power optimization with its application in low power CMOS, quantum computing and nanotechnology. This research proposes a new implementation of Binary Coded Decimal (BCD) adder in reversible logic. The design reduces the number of gates and garbage outputs compared to the existing BCD adder reversible logic implementations. So, this design gives rise to an implementation with a reduced area and delay.**

Keywords: BCD adder, decimal arithmetic, reversible logic, garbage output

1. INTRODUCTION

Energy loss during computation is an important consideration in low power digital design. Landauer's principle states that a heat equivalent to *kT*ln2* is generated for every bit of information lost, where '*k*' is the Boltzmann's constant and '*T*' is the temperature [1]. At room temperature, though the amount of heat generated may be small it cannot be neglected for low power designs. The amount of energy dissipated in a system bears a direct relationship to the number of bits erased during computation. Bennett showed that energy dissipation would not occur if the computations were carried out using reversible circuits [2] since these circuits do not lose information. A reversible logic gate is an *n*-input, *n*-output (denoted as *n*n*) device that maps each possible input pattern to a unique output pattern. There is a significant difference in the synthesis of logic circuits using conventional gates and reversible gates [3]. While constructing reversible circuits with the help of reversible gates fan-out of each output must be 1 without feedback loops. As the number of inputs and outputs are made equal there may be a number of unutilized outputs called garbage in certain reversible implementations. This is the number of outputs added to make an n-input k-output function reversible. For example, a single output function of 'n' variables will require at least n-1 garbage outputs. Classical logic gates such as AND, OR, and XOR are not reversible. Hence, these gates dissipate heat and may reduce the life of the circuit. So, reversible logic is in demand in power aware circuits.

A reversible conventional BCD adder was proposed in [4] using conventional reversible gates. In [4], a full adder design using two types of reversible gates - NG (New Gate) and NTG (New Toffoli Gate) with 2 garbage outputs was implemented. The BCD adder was then designed using such full adders. Even though the implementation was improved in [5] using TSG reversible gates, this approach was not taking care of the fanout restriction of reversible circuits, and hence it was only a near-reversible implementation. An improved reversible implementation of decimal adder with reduced number of garbage outputs is proposed in [6]. The present work proposes a modified version of decimal addition using reversible gates which results in further reduction in number of gates and garbage outputs with a fanout of 1. The design is done using 3 types of reversible gates.

The organization of this paper is as follows: Initially, necessary background on reversible logic gates that are used for the design is given. Then the proposed BCD adder is implemented using reversible gates. Finally, the paper concludes with a comparison of the proposed design with different types of reversible BCD adders available in literature, in terms of delay, number of gates and garbage outputs.

II. REVERSIBLE LOGIC GATES

This section describes reversible gates that are used for the implementation of the proposed BCD adder.

Figure 1 shows a 3*3 New Gate [7]. New Gate can be used as a gate that generates an AND gate, an OR gate or an XOR gate. If B='0', then Q=C and R=(A'C')$\oplus$1=A+C. Similarly, when C='0', then Q=AB and R=A$\oplus$B which are the carry and sum outputs of a half adder. Figure 2 shows a Feynman Gate [8]. Feynman Gate (FG) can be used as a copying gate. Since a fanout greater than one is not allowed, this gate is useful for duplication of the required outputs.

1-4244-1368-0/07/$25.00 ©2007 IEEE

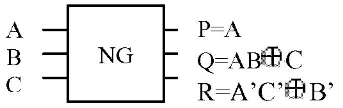

Figure 1: 3*3 New Gate (NG)

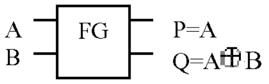

Figure 2: 2*2 Feynman Gate (FG)

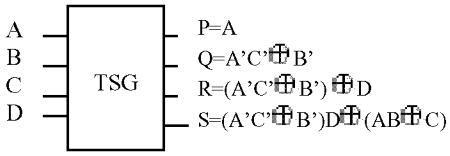

Figure 3: 4*4 TS Gate (TSG)

If B='0', then P=A and Q=A. Figure 3 shows a TS Gate (TSG) [9]. A full adder circuit can be realized by a TSG with C='0' and A, B and C_{in} at A, B, D inputs of TSG. Then Sum and C_{out} are realized at R and S outputs of TSG.

III. Reversible Logic Implementation Of Decimal Adder

The BCD adder shown in Figure 4 has three blocks – 4-bit binary adder, 6-correction circuit and a modified special adder. 4-bit full adder adds the BCD inputs and generates a binary sum, S (S_{3-0}).This output is checked for a value greater than '9' or for a carry out, by the 6-correction circuit which generates a '6-correction' bit, 'L' using Equation 1.

$$L= C_{out} + S_3\,(S_1+S_2) \qquad (1)$$

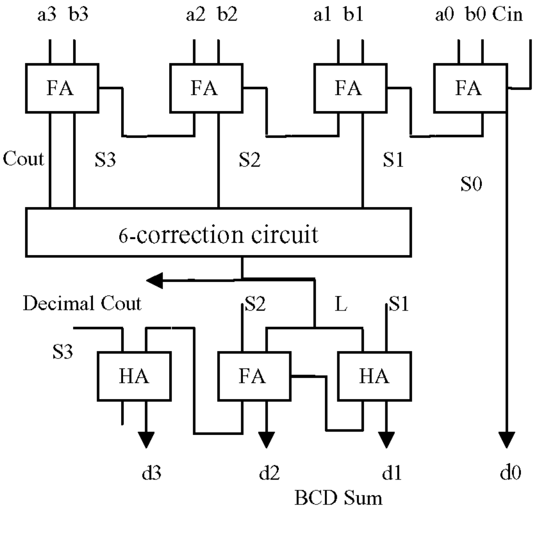

Figure 4: BCD Adder

The inputs to the second adder stage are S (S_{3-0}) and 4-bit number N (N_{3-0}) whose value is 6 (0110_2) or 0 (0000_2) depending on 'L' bit. So, N_0 and N_3 are always zero, and N_1 and N_2 is 'L' bit. To reduce the hardware and to increase the speed of the circuit, the final adder stage (special adder) is a modified version of the 4-bit binary adder with two half adders and one full adder.

Recently, reversible implementations of conventional BCD adders were proposed by Hafiz [4] Thapliyal [5] and James [6]. Implementation by Hafiz [4] makes use of 23 reversible gates and produces 22 garbage outputs whereas the implementation of Thapliyal [5] uses 11 reversible gates and produces 22 garbage outputs without taking care of fanout. There are 4 outputs in which the fanout is more than one in the implementation of [5], of which 3 are having fanout of 2 and the other having 3. If the fanout points were replaced by copying gates FG then the total number of gates would be increased to 16. Reversible implementation of BCD adder in [6] reduces the number of gates to 11 and garbage outputs to 13 with fanout restrictions. The proposed reversible implementation of the BCD adder done using the reversible gates such as TSG, FG and NG shown in Figure 5 takes care of the fanout restriction of reversible circuits and reduces the number of reversible gates.

A 4-bit binary adder is implemented using 4 full adders. A full adder has 3 inputs and 2 outputs. To make a full adder reversible some garbage outputs are to be added. A reversible full-adder circuit can be realized with at least two garbage outputs. In full-adder circuit, there are three input combinations (0, 0, 1), (0, 1, 0) and (1, 0, 0) for which the output is same (1, 0). So, at least two garbage bits are required to make a unique output combination for each input combination. A number of reversible full adders are available in literature [7, 9, 10, 11]. But the implementation of a full adder using TSG [5] takes least number of gates, and produces least number of garbage outputs. Since a full adder can be implemented using one TSG, a 4-bit binary reversible adder implementation requires 4 TSGs and produces 8 garbage outputs.

For reducing the number of gates, the 6-correction circuit output 'L' can be modified as in Equation (2).

$$L= Cout + S_3\,(S_1+S_2) = Cout \oplus S_3\,(S_1+S_2) \qquad (2)$$

It can be seen that the implementation requires 2 gates (2 NGs) to produce the 6-correction output, 'L', and the sum outputs (S_{3-1}) with only one garbage output. The S_1, S_2 and S_3 outputs produced without using any copying gate (FG) can be used as inputs for the next stage. This gives a reduction of 3 gates and 4 garbage outputs compared to the implementation in [4].

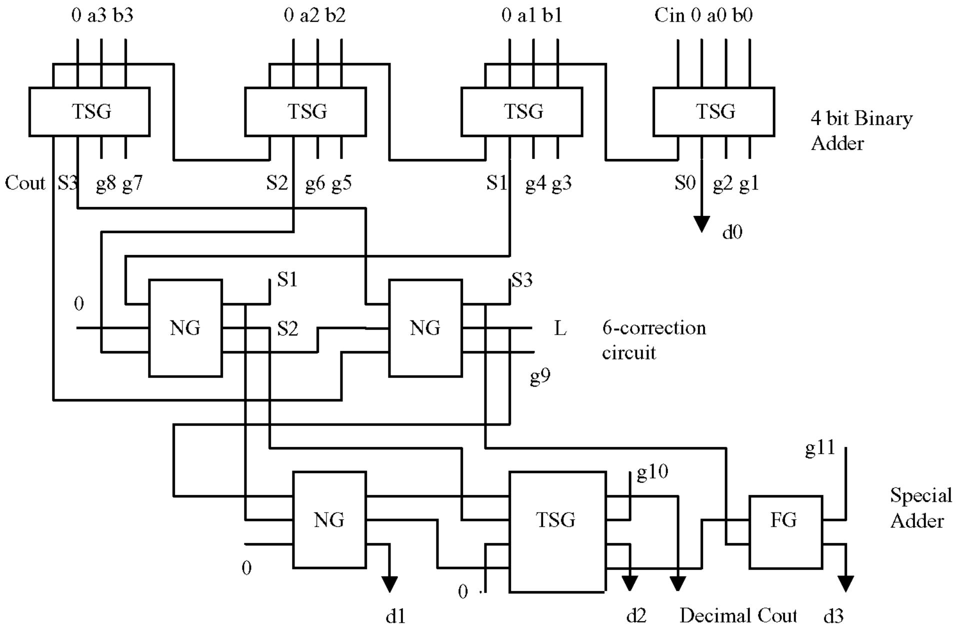

Figure 5: Reversible Implementation of proposed BCD Adder

Special adder is implemented using 3 gates (NG, TSG, FG). It is already seen that a 3*3 NG can implement a half adder, and a 4*4 TSG can implement a full adder. An FG replaces the final half adder in the special adder. This is because only the sum bit (d_3) is required as decimal sum output and the carry is discarded from the final addition. So, using an NG will give rise to 2 garbage outputs while an FG will produce only one garbage output. The BCD sum is indicated as d_{3-0} and carryout from the stage as 'Decimal C_{out}' in Figure 5. This implementation uses 9 reversible gates, and produces 11 garbage outputs. Further, it is noted that the implementation in [4] can be used only as a single digit BCD adder since a carryout (Decimal C_{out}) is not produced. This may be resolved by the addition of one more copying gate. The proposed design can be used for cascading BCD adders for multidigit BCD addition with the help of the carry out (Decimal C_{out}) produced.

An N-digit BCD adder will have a total (worst case) delay (T_{dsum}) equal to the sum of the 'carry delay' (T_{dcout}) through N digits and 'sum delay' through the last digit ($T_{sum-digit}$). This is given in equation (3).

$$T_{dsum} = NT_{dcout} + T_{special-adder} \tag{3}$$

where T_{dcout}= ($T_{bin-adder}$+ $T_{6-correction}$)

$T_{bin-adder}$ is the delay of 4-bit binary adder

$T_{6-correction}$ is the delay in generating Decimal C_{out} (d_{cout}) after $T_{bin-adder}$

$T_{special-adder}$ is the additional delay of the final adder stage (special adder) of the last digit after generating Decimal C_{out} (d_{cout})

For a reversible implementation this is given as

$$T_{rev-dsum} = 7N + 1 \tag{4}$$

where $T_{rev-dcout} = T_{rev-bin-adder} + T_{rev-6-correction}$

$T_{rev-bin-adder}$ = 4 gate delays (4TSG)

$T_{rev-6-correction}$ = 3 gate delays (2NG + 1TSG)

$T_{rev-special-adder}$ = 1 gate delay (1FG)

Similar analysis done on reversible implementations of BCD adders in [4], [5] and [6] are tabulated in Table 1. Even though this delay analysis will not give exact results because of the difference in complexity of the gates used, it gives a good estimate of the delay reduction attained by reversible implementation of proposed BCD adder. The Table also shows a comparison in terms of number of reversible gates and garbage outputs at different levels and for the complete circuit. It is clear that the proposed implementation uses least number of gates, produces least number of garbage outputs and gives least delay compared to all other implementations. The reduction in number of gates (area) required will lead to reduced power consumption.

IV. CONCLUSION AND FUTURE WORK

A modified reversible BCD adder implementation is presented. The architecture is specially designed to make it suitable for reversible logic implementation. It is demonstrated that the proposed design is highly optimized in terms of number of reversible gates and garbage outputs.

TABLE 1. COMPARATIVE ANALYSIS OF REVERSIBLE BCD ADDERS

Reversible BCD Adders	4 bit Adder		6-correction circuit		Correction/ Special Adder		Complete Circuit			
	No: of gates	No: of garbage o/p	No: of gates	No: of garbage o/p	No: of gates	No: of garbage o/p	No: of gates	No: of garbage o/p	Delay of Decimal Cout for N-digit BCD adder	Delay of BCD Sum for N-digit BCD adder
BCD Adder in [4]	NG-4 NTG-4 Total-8	8	FG-3 NG-3 Total-6	6	NG-4 NTG-4 Total-8	8	NG-11 NTG-8 FG-3 Total-22	22	12N	12N+7
BCD Adder using TSG [5] (fanout>1)	TSG-4	8	NG-3	6	TSG-4	8	TSG-8 NG-3 Total-11	22	7N	7N+3
BCD Adder using TSG [5] (fanout=1)	TSG-4	8	FG-3 NG-3 Total-6	6	TSG-4 FG-2 Total-6	8	TSG-8 NG-3 FG-5 Total-16	22	9N	9N+4
BCD Adder in [6] (fanout=1)	TSG-4	8	NG-2 FG-1 Total-3	2	FG-2 NG-1 TSG-1 Total-4	3	TSG-5 NG-3 FG-3 Total-11	13	7N	7N+3
Proposed BCD Adder (fanout=1)	TSG-**4**	**8**	NG-**2**	**1**	FG-1 NG-1 TSG-1 Total-**3**	**2**	TSG-5 NG-3 FG-1 Total-**9**	**11**	**7N**	**7N+1**

The design strategy is chosen in such a way to reduce the most important factor of the reversible circuit cost: the number of garbage outputs. This work forms an initial step in the building of complex reversible systems, which can execute more complicated operations. The reversible circuit proposed here forms the basis of a Decimal ALU for a reversible CPU. VLSI implementations using only one type of modular building block can decrease system design and manufacturing cost. Characterization of new families of ‘n-input’ – ‘n-output’ reversible gates that can be used for regular structures is an area which can be explored further.

In this research, a known traditional logic implementation for BCD adder was modified to get a delay reduction for multi-digit addition, and then each of the internal elements was replaced with reversible equivalents. Further investigation into determining alternate implementations can be done using logic synthesis methods [12].

REFERENCES

[1] R. Landauer, “Irreversibility and Heat Generation in the Computational Process”, IBM Journal of Research Development, 5, 1961, 183-191.

[2] Bennett, C., “Logical Reversibility of Computation,” IBM Journal of Research and Development, 17, 1973, 525-532.

[3] T. Toffoli., “Reversible Computing”, Tech memo MIT/LCS/TM-151, MIT Lab for Computer Science, 1980.

[4] Hafiz Md. Hasan Babu and A. R. Chowdhury, "Design of a Reversible Binary Coded Decimal Adder by Using Reversible 4-bit Parallel Adder”, VLSI Design 2005, pp-255-260, Kolkata, India, Jan 2005.

[5] Himanshu. Thapliyal, S. Kotiyal and M.B Srinivas, “Novel BCD Adders and their Reversible Logic Implementation for IEEE 754r Format”, VLSI Design 2006, Hyderabad, India, Jan 4-7, 2006, pp. 387-392.

[6] R. James, T. K. Shahana, K. P. Jacob and S. Sasi, “Improved Reversible Logic Implementation of Decimal Adder”, IEEE 11th VDAT Symposium Aug 8-11, 2007.

[7] Md. M. H. Azad Khan, “Design of Full-adder With Reversible Gates”, International Conference on Computer and Information Technology, Bangladesh, 2002, pp. 515-519.

[8] R. Feynman, “Quantum Mechanical Computers”, Optical News, 1985, pp. 11-20

[9] H. Thapliyal and M.B Srinivas, “A Novel Reversible TSG Gate and Its Application for Designing Reversible Carry Look-Ahead and Other Adder Architectures”, Tenth Asia-Pacific Computer Systems Architecture Conference, Singapore, Oct 24 - 26, 2005

[10] J.W.Bruce, M.A.Thornton, L.Shivakumariah, P.S.Kokate, X.Li, "Efficient Adder Circuits Based on a Conservative Logic Gate", Proceedings of the IEEE Computer Society Annual Symposium on VLSI, April 2002, PA, USA, pp 83-88.

[11] B. Parhami; “Fault Tolerant Reversible Circuits” Proc. 40th Asilomar Conf. Signals, Systems, and Computers, Pacific Grove, CA, October 2006.

[12] Dmitri Maslov, "Reversible Logic Synthesis”, PhD Dissertation, Computer Science Department, University of New Brunswick, Canada, October 2003.