

Full Length Research Paper

Novel nanometric reversible saturating adder

Majid Haghparast^{1*} and Ali Patiar²

¹Department of Computer Engineering, Shahre-Rey Branch, Islamic Azad University, Tehran, Iran.

²Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran.

Accepted 11 April, 2011

Reversible logic has found emerging attentions in nanotechnology, optical computing, quantum computing and low power CMOS design. In this paper a reversible saturating adder is presented for the first time. Some of digital signal processing (DSP) applications such as digital filters require that the result of arithmetic operations to be saturated. Saturation will happen when we have addition on two numbers with same signs. We have proposed a reversible 4-bit saturating adder for the first time with its essential units such as “overflow detection logic” and “saturation value generator” in the reversible fashion. All the important parameters in the design of reversible circuits like quantum cost, number of constant inputs and number of garbage outputs are reported. All the designs are in the nanometric scales.

Key words: Quantum computing, Reversible logic, saturation arithmetic.

INTRODUCTION

Irreversible hardware computation results in energy dissipation due to information loss. Landauer's research has proved that the amount of energy dissipated for every irreversible bit operation is at least $KT \ln 2$ joules, where $K = 1.3806505 \times 10^{-23} \text{ m}^2 \text{ kg}^{-2} \text{ K}^{-1}$ (Joule/Kelvin) is the Boltzmann constant and T is the temperature in degrees Kelvin at which operation is performed (Landauer, 1961; Parhami, 2006). In 1973, Bennett proved that $KT \ln 2$ energy would not dissipate from a system as long as the system allows the reproduction of the inputs from observed outputs (Bennett, 1973). Reversible logic circuits offer an alternative that allows computation with arbitrarily small energy dissipation (Haghparast and Navi, 2007; Hayes, 2006). Reversible logic supports the process of running system in both forward and backward directions. It means that reversible computations can generate inputs from its outputs and can stop and go back to any point in the computation history. Furthermore, reversible circuits are of major interest in optical computing, low power design, quantum computing and nanotechnology based systems. It is not possible to realize quantum computing without reversible logic and reversible computation in a system can be

performed only if the system is composed of reversible gates (Haghparast and Navi, 2008; Vasudevan et al., 2004; Haghparast et al., 2009). A characteristic of many DSP applications is that they are computationally intensive and execute millions of saturating arithmetic operations (Yadav et al., 1999) and energy dissipation in this type of processors is a point of interest. Signal processing algorithms and applications will be benefited from lower energy consumption and higher speed of quantum computing. One of the common applications in DSP world is filtering. In digital filtering, saturation arithmetic is used frequently and saturating adders as one of saturating arithmetic components, are special kind of adders that produce their saturated sum under overflow condition, instead of producing normal sum. In fact, Saturation is achieved when an “overflow condition” takes place and outcome of operation could not be represented on output; in this case, the result of the function will be set to a predefined “saturation value” (Balzola et al., 2001). For example, enhanced filtering coprocessor (EFCOP) is a sample for using of saturation arithmetic which can be found within Motorola's DSP56300 DSP families as one of its special-propose coprocessors (Venkataramani and Bhaskar, 2002).

Saturation value is the highest or lowest possible value for positive or negative numbers, respectively. Although we have chosen a 4-bit saturating adder to start, we

*Corresponding author. E-mail: haghparast@iausr.ac.ir.

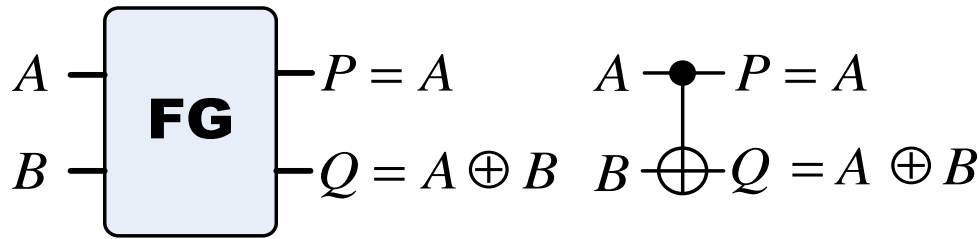


Figure 1. Different symbols for Feynman Gate.

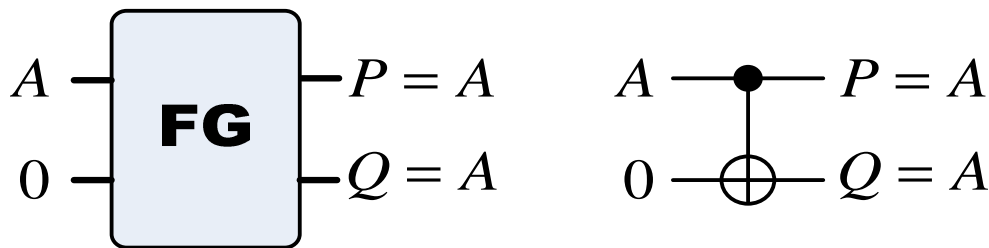


Figure 2. Different symbols for Feynman Gate as a copying output.

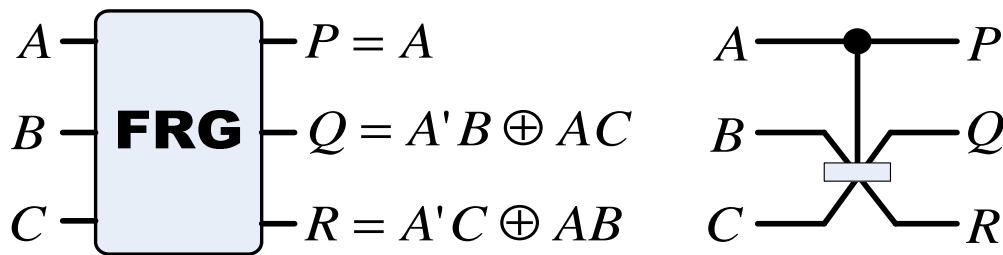


Figure 3. Different symbols for Fredkin Gate.

believe that it can be expanded to any n-bit system. The rest of the paper is organized as follow. Some reversible logic gates, which are used in this work, are going to be introduced. Our proposed reversible saturating adder will explain in details followed. And finally, we will sum up our work by giving a conclusion.

REVERSIBLE LOGIC GATES

There are some reversible logic gates such as Feynman Gate (FG) (Feynman, 1985) and Fredkin Gate (FRG) (Fredkin and Toffoli, 1982) which may have been used in current works. Feynman Gate, also known as controlled NOT (C-NOT), is a 2 x 2 reversible logic which is depicted in Figure 1. It implements the logic functions:

$$P = A \text{ and } Q = A \oplus B.$$

Feynman Gate is the most suitable gate for a single copy of a bit, since it is not producing any garbage output. A '0' on the second input will copy the first input in both outputs of the gate as is shown in Figure 2 (Babu and Chowdhury, 2005). A 3 x 3 Fredkin Gate is shown in the Figure 3. Here the input 'A' is passed as first output. Inputs 'B' and 'C' are swapped to get the second and third output which are controlled by input 'A'. If A = 0, then the outputs are simply duplicates of the inputs; otherwise if A = 1, then the two input lines (B and C) are swapped.

PROPOSED REVERSIBLE SATURATING ADDER

The letters used to symbolize the signals discussed as follows have the following meanings:

“a” and “b” are two 4-bit numbers input into the adder unit

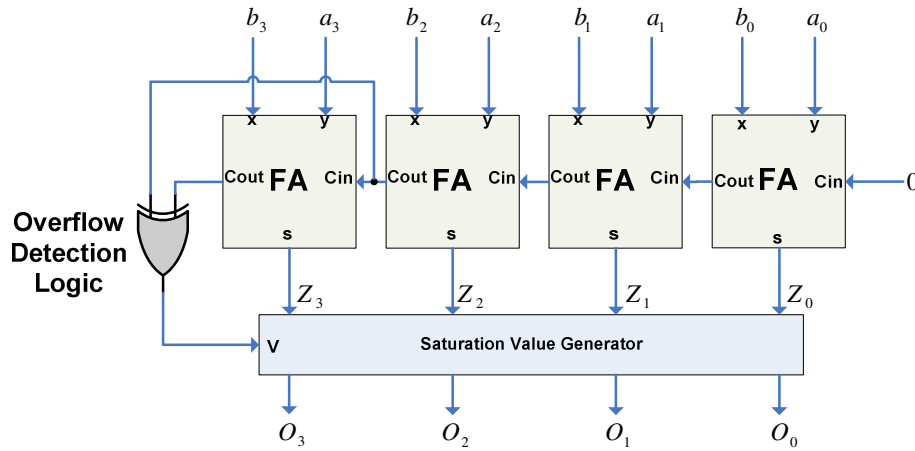


Figure 4. Overflow detection logic for a 4 bit signed number full adder.

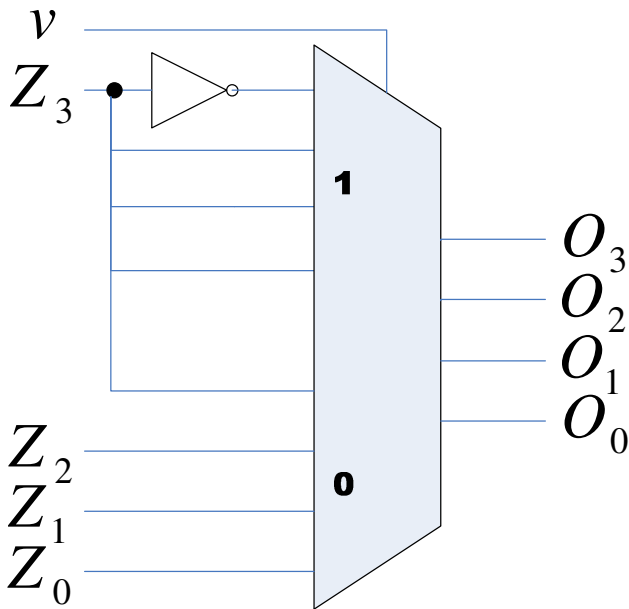


Figure 5. Schematic diagram of "saturation value generator".

as operands; "v" represents the output of the overflow detection unit (ODL); "Z" is a 4-bit number as an intermediary output of addition; and at last, "O" is a 4-bit number as final output of saturating adder unit.

Saturating adders rely on special logic that determines whether there is an overflow condition to saturate the result or not. For implementing a 4 bit saturating adder, four full adders are used to perform addition on input operands. Figure 4 shows a simple exclusive OR gate (XOR) which is used as "overflow detection logic". As overflow occurred, we need to know if the result has been saturated in positive or negative range in order to generate the maximum allowed value for the appropriate

sign. For this purpose, we use a simple equation based on inputs sign:

$$O_v = a.\bar{a}.\bar{a}.\bar{a} \text{ or } O_v = \bar{a}.a.a.a$$

This makes saturation value O_v equal to "1000" for negative saturation (minimum possible value in 2's complement format for 4-bit signed presentation) and "0111" for positive saturation condition. In fact, if operands A and B had got different signs, overflow will never take place and if they had got same signs, result of addition can be overflowed. So overflow is possible only when both of operands A and B have same signs. As Figure 4 demonstrates, for our 4-bit adder in 2's complement numbers system, a single exclusive OR gate can operate well as "overflow detection logic". Now, output of "overflow detection logic" can be used to drive "saturation value generator". Output of "overflow detection logic" and addition result will play role of inputs for this unit. "Saturation value generator" decides on final result of saturating adder. Schematic diagram of "saturation value generator" and its proposed reversible translation are depicted in Figures 5 and 6, respectively. A Feynman Gate (FG) is used as a NOT gate, three Feynman Gates (FG) are used for copying purpose, and four Fredkin Gates implement controlled selection logic. In Figure 6 we present entire reversible "saturation value generator". The adder adds two operands with considering an input carry, and saturates the result if it is necessary by replacing the desirable result at the output. Saturation happens when overflow occurs in addition of two operands and result cannot represent the actual value of addition. For two positive numbers whose sum cannot be represented in 4 bits, saturating results in maximum possible 4-bit number (0111). For two negative numbers whose sum cannot be represented in 4 bits, saturating results in minimum possible 4-bit number

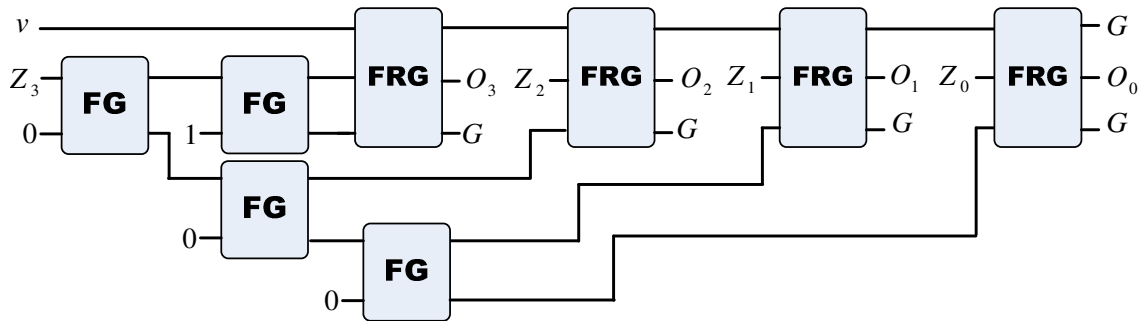


Figure 6. Proposed reversible "saturation value generator".

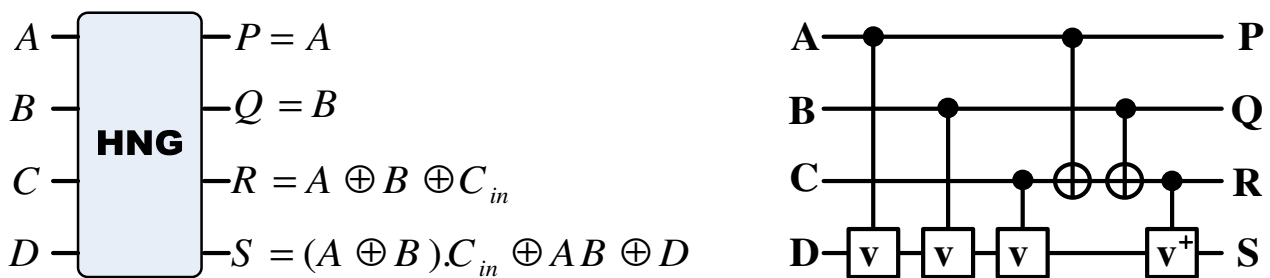


Figure 7. Different symbols for HNG gate.

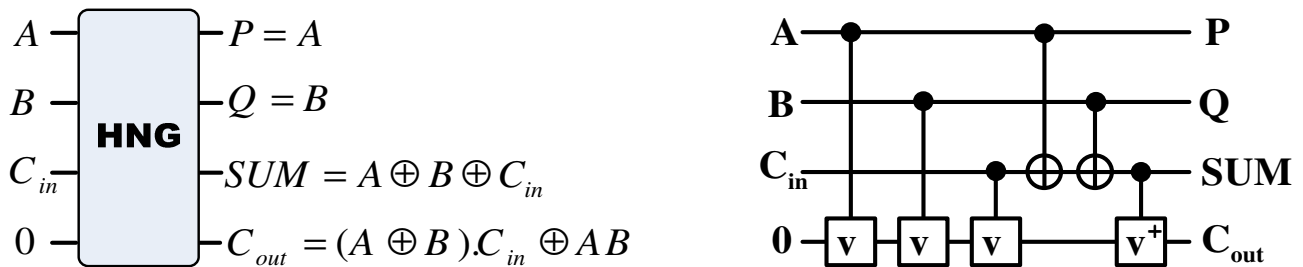


Figure 8. Different symbols for HNG gate as a full adder.

(1000).

We need an adder that can perform the basic function of a full adder (FA). An HNG (Haghparast and Navi, 2008; Haghparast et al., 2008) reversible gate is shown in the Figure 7. HNG reversible gate is used as a full adder in our design. One of the best characteristics of HNG gate is that it can individually operate as a full adder; as it is depicted in Figure 8 if $I_V = (A, B, C_{in}, 0)$, then output vector will become:

$$O_V = (P = A, Q = B, R = SUM, S = C_{out})$$

Complete design of proposed reversible saturating adder

is shown in Figure 9. Four HNG gates are used as full adders which every corresponding bit of both operands connected to A and B inputs of each HNGs. A single Feynman Gate operates as "overflow detection logic" and its output directly feeds "saturation value generator". Other Feynman Gates are used to copy the value of Z3. Table 1 shows constant inputs (CI), garbage outputs (GO/dc), and quantum cost (QC) of our proposed design for $n = 4, 8, 16, 24$ and 32 where n is the width of input operands in bits. Also we have following equations:

$$CI = (2 \times n) + 2 \tag{1}$$

$$GO = n + 2 \tag{2}$$

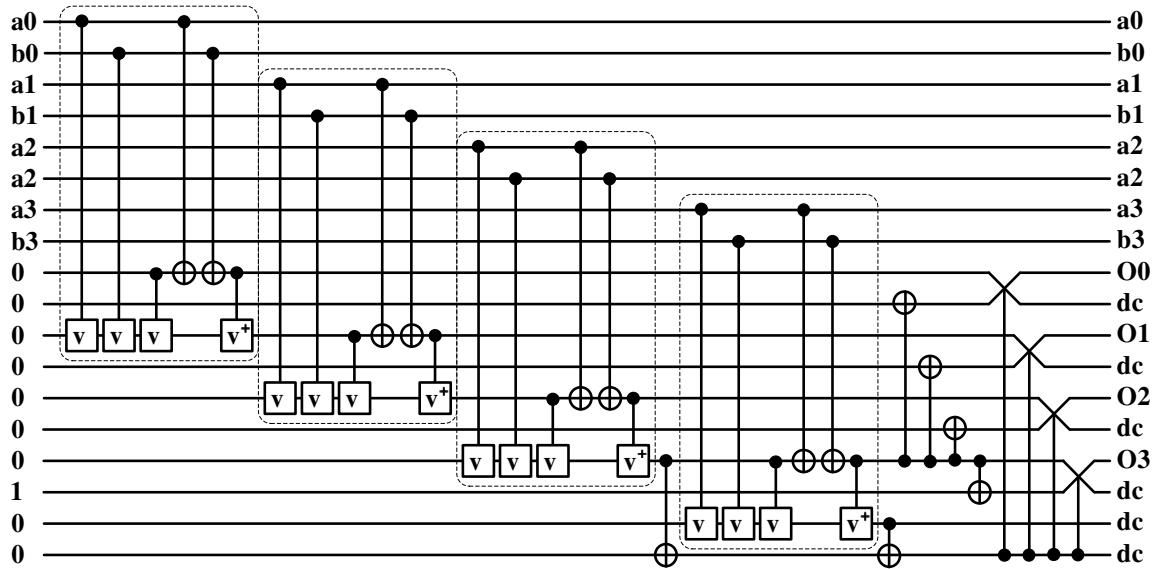


Figure 9. Our proposed reversible saturating adder for $n = 4$.

Table 1. Constant inputs, garbage outputs and quantum cost of the reversible saturating adder for different values of n .

n	Constant inputs	Garbage outputs	Quantum cost
(Width of input operands)	$2n+2$	$n+2$	$6(2n+1)$
$n = 4$	10	6	50
$n = 8$	18	10	98
$n = 16$	34	18	194
$n = 24$	50	26	290
$n = 32$	66	34	386

$$QC = n \times (QC_{HNG} + QC_{FG} + QC_{FRG}) + 2QC_{FG} = 2(6n+1) \quad (3)$$

Equations 1 and 2 are used to calculate constant inputs and garbage outputs of the circuit, respectively. Equation 3 is used to calculate quantum cost of the circuit, based on quantum cost for each type of reversible gates used in our design where:

$$QC_{HNG} = 6, QC_{FRG} = 5 \text{ and } QC_{FG} = 1.$$

CONCLUSION

A reversible saturating adder has been presented in this paper for the first time. In our design, we have used reversible HNG gate from one of our previous works as a full adder. For future works, it is possible to consider a complete range of arithmetic operations in reversible saturating mode. All the designs are in the nanometric scales.

REFERENCES

- Babu HMH, Chowdhury AR (2005). Design of a reversible binary coded decimal adder by using reversible 4-bit parallel adder. Proc. ICVD, pp. 255-260.
- Balzola PI, Schulte MJ, Ruan J, Glossner J, Hokenek E (2001). Design alternatives for parallel saturating multioperand adders. Proc. ICCD, pp. 172-177.
- Bennett CH (1973). Logical reversibility of computation. IBM J. Res. Dev., 17: 525-532.
- Feynman R (1985). Quantum mechanical computers. Optics News, 11: 11-20.
- Fredkin E, Toffoli T (1982). Conservative logic. Int. J. Theor. Phys., 21: 219-253.
- Haghparast M, Jassbi SJ, Navi K, Hashemipour O (2008). Design of a novel reversible multiplier circuit using HNG gate in nanotechnology. World Appl. Sci. J., 3(6): 974-978.
- Haghparast M, Mohammadi M, Navi K, Eshghi M (2009). Optimized reversible multiplier circuit. J. Circuits Syst. Comp., 18(2): 311-323.
- Haghparast M, Navi K (2008). A novel reversible BCD adder for nanotechnology based systems. Am. J. Appl. Sci., 5(3): 282-288.
- Haghparast M, Navi K (2007). A novel reversible full adder circuit for nanotechnology based systems. J. Appl. Sci., 7(24): 3995-4000.
- Hayes B (2006). Reversible engineering. Am. Sci., 94: 107-111.
- Landauer R (1961). Irreversibility and heat generation in the computing process. IBM J. Res. Dev., 5: 183-191.

- Parhami B (2006). Fault Tolerant Reversible Circuits. Proc. ACSSC, pp. 1726-1729.
- Vasudevan DP, Lala PK, Parkerson JP (2004). Online testable reversible logic circuit design using NAND blocks. Proc. DFTVS, pp. 324-331.
- Venkataramani B, Bhaskar M (2002). Digital Signal Processors: Architecture, Programming and Applications. In "An Overview of Motorola DSP563xx Processors." McGraw-Hill, pp. 373-390.
- Yadav N, Schulte M, John G (1999). Parallel Saturating Fractional Arithmetic Units. Ninth Great Lakes Symposium on VLSI, pp. 214-217.