International Journal of
Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-2, Issue-5)

Research Article | **May 2013**

Realization using Reversible Logic with CMOS Header Switching for Power Gating

Dr.Y.Srinivasa Rao
UGC-PDF Research Fellow &
Senior Associate Professor & BOS Member
Instrument Technology Department
AU College of engineering (A)
Andhra university
Visakhapatanam, India.

Mr. Abraham C G
Assistant Professor
Dept. of Electronics &
Communication Engineering
St.Jospeh's College of
Engineering and Technology,
Palai, India.

Abstract—

This paper proposes a Parity Generator and Checker designed using reversible logic with a CMOS header switching for power gating. Reversible logic uses logic gates with zero heat dissipation. The main functional building blocks of computational units are the adder blocks. The computation process produces results based on the incoming inputs. If incoming inputs are same then the computation need not be done again, as the results are already computed and available for the same. So a large amount of power and memory can be saved. This is achieved by the CMOS header switching mechanism. The proposed circuit is based on time constraint and have been simulated for better power saving and memory usage

Keywords---

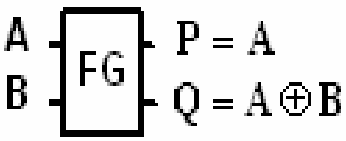
I. Introduction

The reversible logic design is a design where the logic gates used do not dissipate heat and so no information is lost. This logic enables to reproduce the inputs from the outputs. As said by Bennett "no energy would dissipate from a system if it would be able to return to its initial state from its final state regardless of what occurred in between". The logic gates to be reversible should have one-to-one mapping between the inputs and outputs. This produces unique outputs for each input. This logic has extremely low heat dissipation but the circuit occupies larger area which is its disadvantage. A Parity Generator & Checker designed using reversible logic is combined with a CMOS header switch. This provides better power grating to the circuit. Since the circuit occupies larger area, once if implemented in an IC with power gating circuit helps to be useful over the area constraint.

II. Basic Reversible Gates

1. Feynman Gate

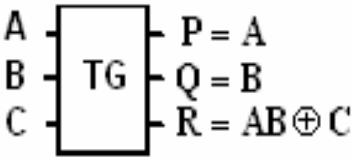
The well known (2, 2) reversible gate is the Feynman gate. The logical functions performed by a Feynman gate with input vector (A, B) and output vector (P,Q) are shown in Fig.



In Feynman gate, one of the input bits act as control signal (A). That is, if A= 0 then the output Q follows the input B. If A = 1 then the input B is flipped at the output Q. So it is called as controlled NOT (1-NOT) and also called as quantum XOR because of its popularity in the field of quantum computing. This gate is one-through gate which means that one input variable is also output. Feynman gate acts as copying gate when the second input is zero by duplicating the first input at the output.

2. Toffoli Gate

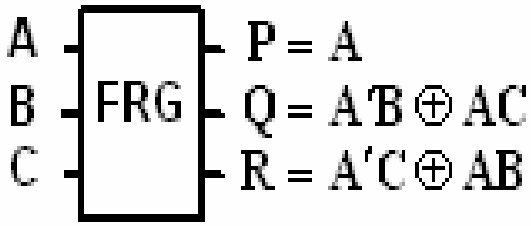
Toffoli gate is a (3, 3) reversible gates. Fig. shows the Toffoli gate.



This gate is two through gate because two of its outputs are identical with its inputs. Because of this, Toffoli gate is also known as two controlled NOT (2-CNOT). If the first two input bits are one, then the third output bit is the inverse of third input bit i.e., A = B = 1, then R = C. Toffoli gate performs basic AND operation when zero is given as its third input (C = 0; R = AB). As discussed earlier, any reversible gate has an inverse or dual. The dual of Toffoli gate is also a Toffoli gate and so it is self-reversible.

3. Fredkin Gate

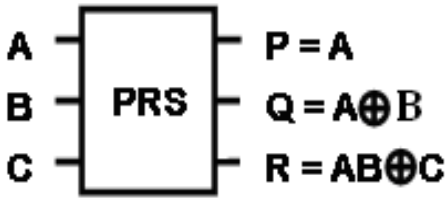
Fredkin gate is another (3, 3) reversible gate which realizes P=A, Q=A'B xor AC' and R=A'C xor AB where (A,B, C) is the input vector and (P, Q, R) is the output vector. Fredkin gate is also self-reversible as it is its own inverse. It is a conservative gate. It uses 'A' as its control input: if A = 0, then the outputs are simply duplicates of the inputs; otherwise if A = 1, then the two input lines (B and C) is interchanged at the output.



Fredkin gate is a universal gate, that is, we can construct the basic blocks such as AND, OR, NOT and other gates from this Fredkin gate by pre-setting some of its inputs.

4. Peres Gate

Peres gate is another important gate which has a low quantum cost as compared to other gates. It is shown in Fig. A single Peres gate can give generate and propagate outputs when the third input C = 0. Because of this, it is quite useful while designing carry look-ahead adders. Two Peres gates can be combined to form a full adder



5. TSG Gate

TSG gate is a (4,4) reversible gate. The main aspect of this gate is that it can work singly as a reversible full adder.This gate is better than its counterparts in terms of number of gates and garbage outputs. Minimising the number of reversible gates and garbage output is one of the major concerns of reversible logic which is achieved to an extent in TSG gate.

Parity Generator Concept:

Parity is a very useful tool in information processing in digital computers to indicate any presence of error in bit information. External noise and loss of signal strength cause loss of data bit information while transporting data from one device to other device, located inside the computer or externally. To indicate any occurrence of error, an extra bit is included with the message according to the total number of 1s in a set of data which is called parity.

If the extra bit is considered 0 if the total number of 1s is even and 1 for odd quantities of 1s in a set of data, then it is called even parity. On the other hand, if the extra bit is 1 for even quantities of 1s and 0 for an odd number of 1s, then it is called odd parity. A parity generator is a combination logic system to generate the parity bit at the transmitting side.

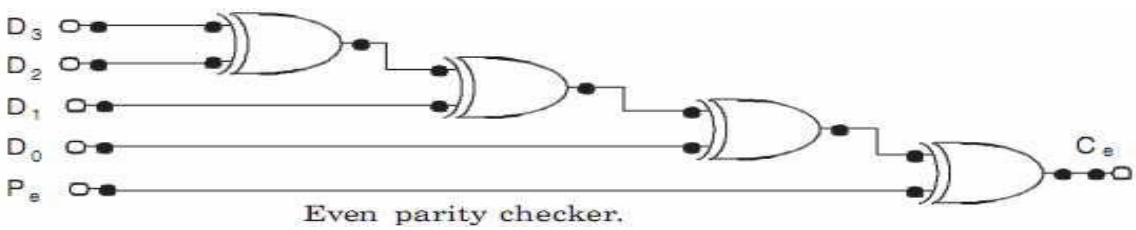
4-bit message D3D2D1D0	Even Parity (Pe)	Even Parity Checker (Ce)	4-bit message D3D2D1D0	Odd Parity (Po)	Odd Parity Checker(Co)
0000	0	0	0000	1	0
0001	1	0	0001	0	0
0010	1	0	0010	0	0
0011	0	0	0011	1	0
0100	1	0	0100	0	0
0101	0	0	0101	1	0
0110	0	0	0110	1	0
0111	1	0	0111	0	0
1000	1	0	1000	0	0
1001	0	0	1001	1	0
1010	0	0	1010	1	0
1011	1	0	1011	0	0
1100	0	0	1100	1	0
1101	1	0	1101	0	0
1110	1	0	1110	0	0
1111	0	0	1111	1	0

Four bit Message $D_3D_2D_1D_0$	*Even Parity* (P_e)	*Odd Parity* (P_o)
0000	0	1
0001	1	0
0010	1	0
0011	0	1
0100	1	0
0101	0	1
0110	0	1
0111	1	0
1000	1	0
1001	0	1
1010	0	1
1011	1	0
1100	0	1
1101	1	0
1110	1	0
1111	0	1

If the message bit combination is designated as D3D2D1D0, and Pe , Po are the even and odd parity respectively, then it is obvious from the table that the Boolean expressions of even parity and odd parity are
Pe = D3 D2 D1 D0 and Po = (D3 D2 D1 D0)
This can be implemented by either Feynman Gate or TSG gate depending upon the number of inputs. If the numbers of inputs are large then these reversible gates can be cascaded to obtain the required result.

Parity Checker Concept:
The message bits with the parity bit are transmitted to their destination, where they are applied to a parity checker circuit. The circuit that checks the parity at the receiver side is called the parity checker. The parity checker circuit produces a check bit and is very similar to the parity generator circuit. If the check bit is 1, then it is assumed that the received data is incorrect. The check bit will be 0 if the received data is correct.

Here the entire check bit is 0 indicating the received data is correct. The parity checker can be shown as below



Even parity checker.

The parity checker can be implemented with the TSG gate or the Feynman gate in the reversible logic. The generated output is XOR'ed with the inputs to obtain the checked out put. This provides better performance with less memory requirement, delay and power dissipation compared to the irreversible logic.

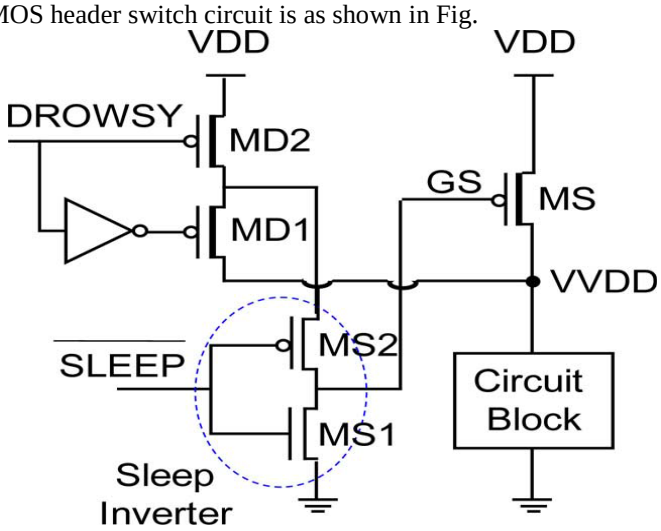
Ease of Use

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures

proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

III. Power Gating

A technique that uses high Vt sleep transistors which cut off VDD from a circuit block when the block is not switching. The sleep transistor sizing is an important design parameter. The technique is also known as MTCMOS. Power gating uses low-leakage PMOS transistors as header switches to shut off power supplies to parts of a design in standby or sleep mode. The CMOS header switch circuit is as shown in Fig.



Header Switch Circuit

Operation of CMOS Header Switch:

- **Header cell:**
  - When SLEEP = 0, MS1 : ON
    Voltage level at GS :0
    Independent of the value of the DROWSY input, circuit is in active mode.
  - When SLEEP =1, if DROWSY =0,

MS2 , MD2 :ON, MS: OFF,

The switch cell will operate in sleep mode. Sub-threshold leakage of the circuit block is limited by OFF HVT devices, MS and MD1.Exists a sneak path from VDD to the ground through MS1, MS2, and MD2.

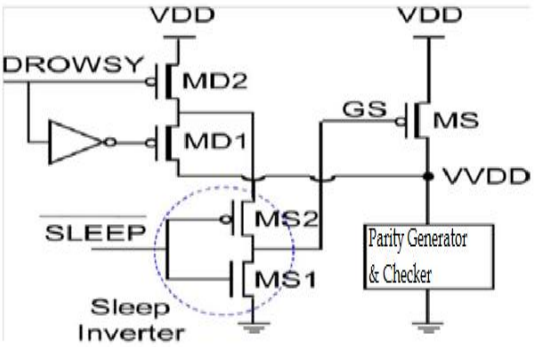
Transistor requirement: (only 4)

- MD1, MD2
- two transistors inside the inverter that feeds into gate terminal of MD2.

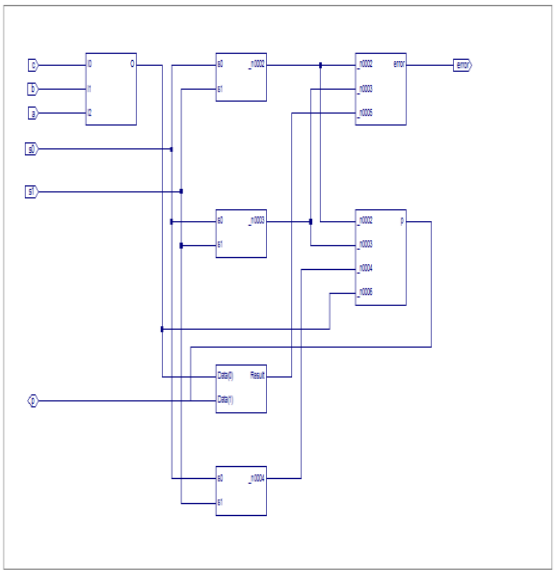
IV. REALIZED PARITY GENERATOR AND CHECKER WITH CMOS HEADER CIRCUIT

Presents a power-gated Parity Generator and Checker using reversible logic, used to reduce dynamic power consumption in the active mode by implementing voltage scaling.

The proposed circuit of Parity Generator & Checker using reversible logic with CMOS header switch is as shown



The RTL schematic of the Parity Generator and Checker using Reversible logic is as shown



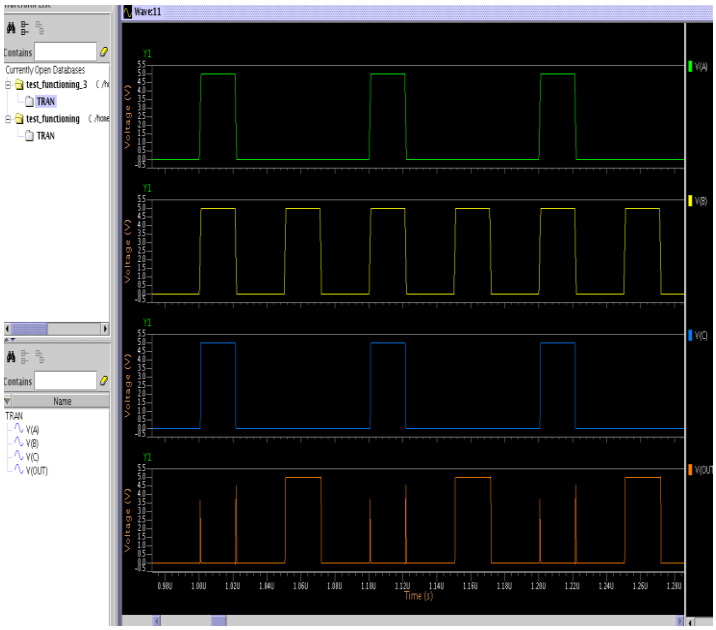
REVERSIBLE PARITY GENERATOR & CHECKER WITH CMOS HEADER CIRCUIT

Working:

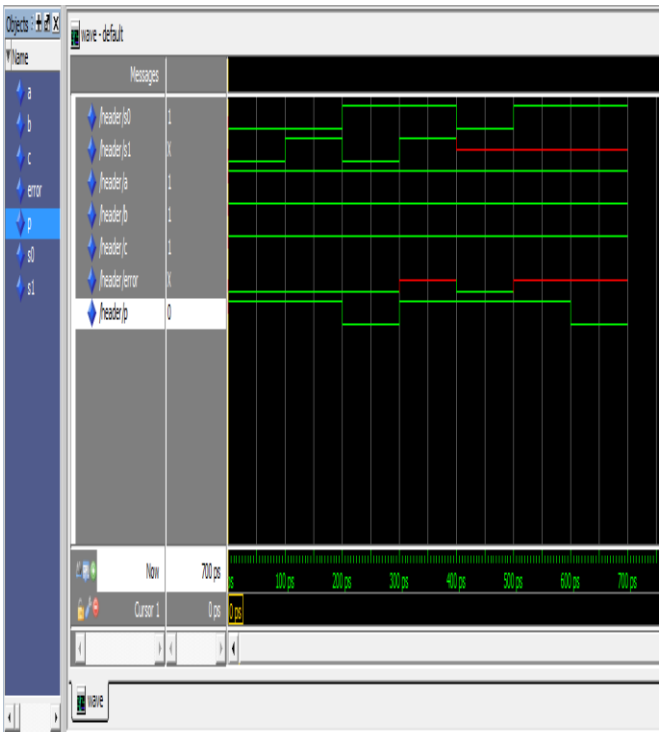
CMOS HEADER INPUT(s0,s1)	PARITY GENERATOR STATUS	PARITY CHECKER STATUS
00	Active	Active
01	Active	Active
10	Sleep	Sleep
11	Active	Drowsy

From the circuit we can understand that when ever s0 is '0' then the circuit will be in the Active state and so the generate and checking process will happen. But when the input s0 is '1' then depending on the input of s1 then circuit works in sleep mode and drowsy mode. Thus the power can be saved. This CMOS Header circuit helps to save the power by maintaing the circuit in sleep mode or drowsy mode when not in use.

V. OUTPUT WAVEFORM



Output wave form of CMOS Header with IC Station(Mentor Graphics)



Output Waveform Of Reversible Parity Generator & Checker With Cmos Header Circuit with Model Sim(Mentor Graphics)

VI. Conclusion

The Parity Generator and Checker designed using reversible logic with CMOS header switch enables the circuit in providing better performance with less memory usage and low power consumption with time constraint. This circuit saves the power during computation by moving from active mode to sleep mode when the input does not change. During this period the output can be reproduced from the buffer where it is stored. So by using the reversible logic to design the Parity Generator and Checker reduces the memory usage and heat dissipation and by adding the header switch enable reduce power consumption of the circuit. This is carried out in time constraints.

Reference

[1] R.Landauer, "Irreversibility and Heat Generation in the Computational Process", IBM Journal of Research and Development, 5,pp. 183-191,1961.
[2] Dmitri Maslov, "Reversible logic synthesis", University of New Brunswick, September 2003.
[3] C.H.Bennett, "Logical Reversibility of Computation", IBM J.Research and Development, pp.525-532, November1973.
[4] Feynman R., 1985. Quantum mechanical computers, Optics News, 11:11-20.
[5] Toffoli T., 1980. Reversible computing, Tech Memo MIT/LCS/TM-151, MIT Lab for Computer Science.
[6] Fredkin E. and T. Toffoli, 1982. Conservative logic. Int'l J. Theoretical Physics, 21: 219-253.
[7] Azad Khan Md. M. H., 2002. Design of full adder with reversible gate. International Conference on Computer and Information Technology, Dhaka, Bangladesh, pp. 515-519.
[8] Peres, A. 1985. Reversible logic and quantum computers. Physical Review A, 32: 3266-3276.
[9] Stephan Brown, Zvonko Vranesic, " Fundamentals of digital logic design with VHDL", Second edition, 2004
[10] Reto Zimmermann, "Binary adder architectures for cell based VLSI and their synthesis", Swiss Federal Institute of Technology, Zurich, 1997.
[11] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.